

Multivariate analysis of variance for functional data using R

Tomasz Górecki and Łukasz Smaga

Faculty of Mathematics and Computer Science
Adam Mickiewicz University

European R Users Meeting
October 12-14, 2016, Poznań



- The multivariate analysis of variance problem for functional data can be formulated in the following way.
- Let

$$\mathbf{X}_{ij}(t) = (X_{ij1}(t), \dots, X_{ijp}(t))', \quad i = 1, \dots, l, \quad j = 1, \dots, n_i$$

denote l groups of vectors of random functions defined over a given finite interval $T = [a, b]$.

- Let $SP_p(\mathbf{m}, \mathbf{\Gamma})$ denote a p -dimensional stochastic process with mean vector $\mathbf{m}(t)$, $t \in T$ and covariance function $\mathbf{\Gamma}(s, t)$, $s, t \in T$.
- Assuming that the \mathbf{X}_{ij} are i.i.d. $SP_p(\mathbf{m}_i, \mathbf{\Gamma})$, $i = 1, \dots, l$, $j = 1, \dots, n_i$, it is interesting to test the equality of the l mean vectors

$$H_0 : \mathbf{m}_1(t) = \dots = \mathbf{m}_l(t), \quad t \in T. \quad (1)$$

Permutation tests based on a basis function representation

- Assume that $\mathbf{X}_{ij} \in L_2^p(T)$, $i = 1, \dots, l$, $j = 1, \dots, n_i$, where $L_2^p(T)$ is a Hilbert space of p -dimensional vectors of square integrable functions on the interval T , equipped with the following inner product: $\langle \mathbf{x}, \mathbf{y} \rangle = \int_T \mathbf{x}'(t)\mathbf{y}(t)dt$.
- Using the orthonormal basis in $L_2(T)$, we can approximate any function in this space arbitrarily well by taking a linear combination of a sufficiently large number of these functions (Ramsay and Silverman, 2005).
- Therefore we can assume that

$$X_{ijm}(t) = \sum_{r=0}^{K_m} \alpha_{ijmr} \varphi_r(t), \quad t \in T, \quad m = 1, \dots, p, \quad (2)$$

where α_{ijmr} , $r = 0, \dots, K_m$, are random variables with finite variance and $\{\varphi_r\}$ is orthonormal basis of $L_2(T)$.

Permutation tests based on a basis function representation

- Let $KM = \max\{K_1, \dots, K_p\}$, $\alpha_{ijm} = (\alpha_{ijm0}, \dots, \alpha_{ijmK_m}, 0, \dots, 0) \in \mathbb{R}^{KM+1}$ and $\varphi(t) = (\varphi_0(t), \dots, \varphi_{KM}(t))'$, $t \in T$. Then

$$\mathbf{X}_{ij}(t) = \begin{pmatrix} \alpha_{ij1} \\ \vdots \\ \alpha_{ijp} \end{pmatrix} \varphi(t) = \alpha_{ij} \varphi(t), \quad t \in T. \quad (3)$$

- Separately for each feature, the vectors $\alpha_{ijm}^* = (\alpha_{ijm0}, \dots, \alpha_{ijmK_m})'$, $i = 1, \dots, l$, $j = 1, \dots, n_i$, can be estimated by the least squares method, and the optimum value of K_m may be selected using the Bayesian information criterion (BIC), extended Bayesian information criterion (eBIC) or Akaike information criterion (AIC); then from the values of K_m corresponding to all processes a modal, minimum, maximum or mean value is selected as the common value for all $(X_{ijm}(t), t \in T)$, $i = 1, \dots, l$, $j = 1, \dots, n_i$ (see, for example, Górecki, Krzyśko and Waszak, 2014).

Permutation tests based on a basis function representation

- Similarly to the case of MANOVA tests for multivariate variables (see Anderson, 2003), the following matrices will be used in constructing test statistics for FMANOVA problem:

$$\mathbf{E} = \sum_{i=1}^l \sum_{j=1}^{n_i} \int_T \left(\mathbf{X}_{ij}(t) - \bar{\mathbf{X}}_i(t) \right) \left(\mathbf{X}_{ij}(t) - \bar{\mathbf{X}}_i(t) \right)' dt,$$

$$\mathbf{H} = \sum_{i=1}^l n_i \int_T \left(\bar{\mathbf{X}}_i(t) - \bar{\mathbf{X}}(t) \right) \left(\bar{\mathbf{X}}_i(t) - \bar{\mathbf{X}}(t) \right)' dt, \quad t \in T.$$

- The sample group mean functions and the sample grand mean function:

$$\bar{\mathbf{X}}_i(t) = \frac{1}{n_i} \sum_{j=1}^{n_i} \mathbf{X}_{ij}(t) = \frac{1}{n_i} \left(\sum_{j=1}^{n_i} \alpha_{ij} \right) \varphi(t), \quad \text{for } i = 1, \dots, l,$$

$$\bar{\mathbf{X}}(t) = \frac{1}{n} \sum_{i=1}^l \sum_{j=1}^{n_i} \mathbf{X}_{ij}(t) = \frac{1}{n} \left(\sum_{i=1}^l \sum_{j=1}^{n_i} \alpha_{ij} \right) \varphi(t), \quad t \in T.$$

Permutation tests based on a basis function representation

Theorem 1

If the components of stochastic processes $(\mathbf{X}_{ij}(t), t \in T)$, $i = 1, \dots, l$, $j = 1, \dots, n_i$, are represented by a finite number of orthonormal basis functions, i.e. the equation (3) holds, then $\mathbf{E} = \mathbf{A} - \mathbf{B}$ and $\mathbf{H} = \mathbf{B} - \mathbf{C}$, where

$$\mathbf{A} = \sum_{i=1}^l \sum_{j=1}^{n_i} \alpha_{ij} \alpha'_{ij}, \quad \mathbf{B} = \sum_{i=1}^l \frac{1}{n_i} \sum_{j=1}^{n_i} \sum_{m=1}^{n_i} \alpha_{ij} \alpha'_{im}, \quad \mathbf{C} = \frac{1}{n} \sum_{i=1}^l \sum_{j=1}^{n_i} \sum_{t=1}^l \sum_{u=1}^{n_t} \alpha_{ij} \alpha'_{tu}.$$

- By the above theorem, the matrices \mathbf{E} and \mathbf{H} can be designated only on the basis of the coefficient matrices α_{ij} , $i = 1, \dots, l$, $j = 1, \dots, n_i$, which is important for the practical implementation of the tests described below.

Permutation tests based on a basis function representation

- We consider the following test statistics, which are constructed based on test statistics appearing in the MANOVA problem for multivariate variables (see Anderson, 2003):
 - ① $W = \det(\mathbf{E}) / \det(\mathbf{E} + \mathbf{H})$ – the Wilks lambda test statistic;
 - ② $LH = \text{trace}(\mathbf{H}\mathbf{E}^{-1})$ – the Lawley-Hotelling trace test statistic;
 - ③ $P = \text{trace}(\mathbf{H}(\mathbf{H} + \mathbf{E})^{-1})$ – the Pillai trace test statistic;
 - ④ $R = \lambda_{\max}(\mathbf{H}\mathbf{E}^{-1})$ – Roy's maximum root test statistic ($\lambda_{\max}(\mathbf{M})$ denotes the maximum eigenvalue of a matrix \mathbf{M}).
- We reject the null hypothesis for small values of W , and for large values of LH , P and R .
- It seems to be difficult to find even the approximate distribution of the above test statistics under the null hypothesis. For this reason, we propose the permutation testing procedure.

Permutation tests based on a basis function representation

- The permutation test can sometimes be time-consuming. However, from the following remark, it follows that we do not have to compute all the elements of the test statistics W, LH, P and R B times. This observation results in quite fast implementation of the permutation tests.

Remark 1

*Any permutation of the stochastic processes $(\mathbf{X}_{ij}(t), t \in T)$, $i = 1, \dots, l$, $j = 1, \dots, n_i$, leaves the values of the sums **A** and **C** unchanged.*

- By Remark 1, for each random permutation of the data only the sum **B** has to be calculated.

Tests based on random projections

- We also propose tests based on random projections for FMANOVA problem, the idea for which is taken from Cuesta-Albertos and Febrero-Bande (2010).
- Assume that \mathcal{H} is a separable Hilbert space endowed with a scalar product $\langle \cdot, \cdot \rangle$, and μ is a Gaussian distribution on \mathcal{H} such that each of its one-dimensional projections is nondegenerate. Moreover, let $\mathbf{m}_i = (m_{i1}, \dots, m_{ip})'$, $m_{ij} \in \mathcal{H}$, $i = 1, \dots, l$, $j = 1, \dots, p$, and $v_m \in \mathcal{H}$ be a vector chosen at random employing the distribution μ for $m = 1, \dots, l$. It is easy to see that if the null hypothesis H_0 in (1) holds, then the null hypothesis

$$H_0^{\mathbf{V}} : \begin{pmatrix} \langle v_1, m_{11} \rangle \\ \vdots \\ \langle v_p, m_{1p} \rangle \end{pmatrix} = \dots = \begin{pmatrix} \langle v_1, m_{l1} \rangle \\ \vdots \\ \langle v_p, m_{lp} \rangle \end{pmatrix}$$

also holds for every $\mathbf{V} = (v_1, \dots, v_p)' \in \mathcal{H} \times \dots \times \mathcal{H}$.

- However, what happens to $H_0^{\mathbf{V}}$ when H_0 fails?

Theorem 2

Under the above assumptions about \mathcal{H} and μ , if $m_{ij} \in \mathcal{H}$, $i = 1, \dots, l$, $j = 1, \dots, p$, and there exist r_1, r_2, s such that $m_{r_1 s} \neq m_{r_2 s}$, then $(\mu \times \dots \times \mu)(\mathcal{A}) = 0$, where $\mu \times \dots \times \mu$ is a product measure on the space $\mathcal{H} \times \dots \times \mathcal{H}$ and the set \mathcal{A} consists of all vectors $\mathbf{V} = (v_1, \dots, v_p)' \in \mathcal{H} \times \dots \times \mathcal{H}$ for which $H_0^{\mathbf{V}}$ is true.

- Theorem 2 shows that if H_0 fails, then for $(\mu \times \dots \times \mu)$ -almost every $\mathbf{V} \in \mathcal{H} \times \dots \times \mathcal{H}$, the null hypothesis $H_0^{\mathbf{V}}$ also fails.
- Summarizing the above considerations, we conclude that a statistical test at level $\alpha \in (0, 1)$ to test $H_0^{\mathbf{V}}$ is also a statistical test at the same level to test H_0 . In other words, a test for the multivariate analysis of variance for random vectors can be used to test the multivariate analysis of variance problem for functional data by using it to test $H_0^{\mathbf{V}}$.

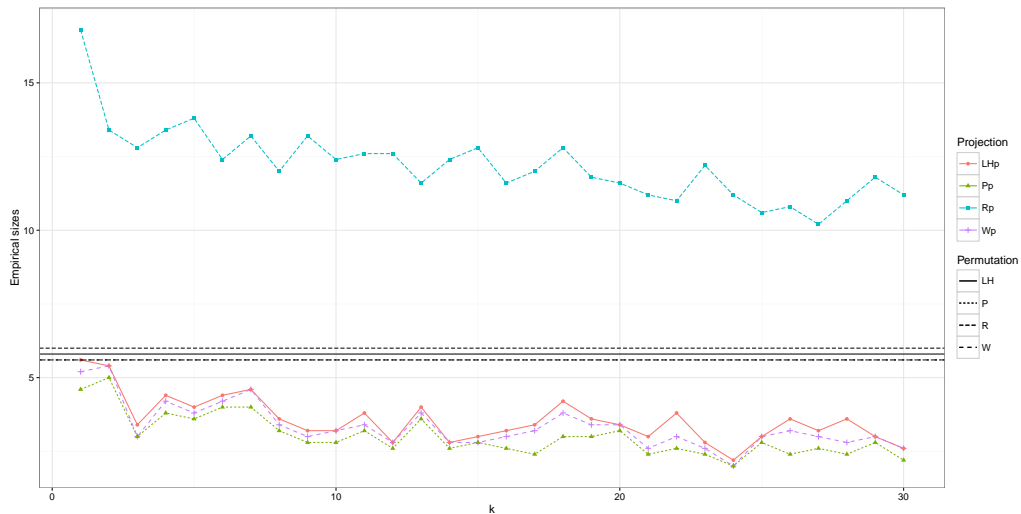
Tests based on random projections

- Assume that $\mathbf{X}_{ij} \in L_2^p(T)$, $i = 1, \dots, l$, $j = 1, \dots, n_i$. We propose the following testing procedure:
 - ① Choose, with Gaussian distribution, functions v_m , $m = 1, \dots, p$, in the space $L_2(T)$.
 - ② Compute the integrals $Y_{ijm} = \int_T X_{ijm}(t)v_m(t)dt$ for $i = 1, \dots, l$, $j = 1, \dots, n_i$, $m = 1, \dots, p$.
 - ③ Apply the appropriate MANOVA test for the l samples of vectors $\mathbf{Y}_{ij} = (Y_{ij1}, \dots, Y_{ijp})'$, $i = 1, \dots, l$, $j = 1, \dots, n_i$.
- To perform step 1, we use the distribution of a standard Brownian motion.
- In step 3 of the testing procedure, we have to choose an appropriate MANOVA test for random vectors. We applied the well-known MANOVA tests, namely Wilk's lambda test (the Wp test), the Lawley-Hotelling trace test (the LHp test), the Pillai trace test (the Pp test), and Roy's maximum root test (the Rp test), or their permutation versions.

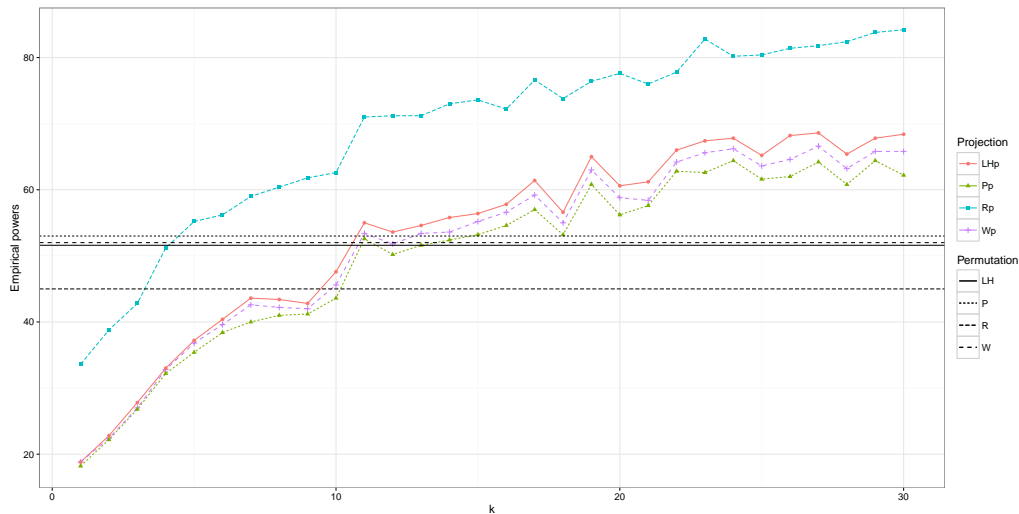
Tests based on random projections

- This procedure has two main drawbacks.
- Firstly, we are losing some information, since we are replacing a function by just one real number. This can be expected to bring some loss of power. Secondly, by employing a randomly chosen projection, we are accepting some random instability in the testing procedure, since it may happen that when we run the procedure twice, a null hypothesis could first be rejected and later accepted.
- To mitigate these problems, we propose to test the null hypothesis for each of $k > 1$ projections, and then correct the obtained p -values to control the False Discovery Rate (FDR). Cuesta-Albertos and Febrero-Bande (2010), using the results of Benjamini and Yekutieli (2001), proposed to take as a corrected p -value the quantity $\inf\{kp_{(i)}/i, i = 1, \dots, k\}$, where $p_{(1)} \leq \dots \leq p_{(k)}$ are the ordered p -values obtained in k tests for the same null hypothesis.
- By Cuesta-Albertos and Febrero-Bande (2010) and Górecki and Smaga (2016), we recommend to choose the number of projections k near 30, but when the results are unconvincing (e.g., the p -value is close to the nominal level), use a higher k .

Simulation study



Simulation study



fmanova package - preparation of the data

- In practice, functional samples are not continuously observed, i.e. each function is usually observed on a grid of design time points. In our implementations of tests in the R programming language (R Core Team, 2015), all functions are observed on a common grid of design time points equally spaced in T .
- In the case where the design time points are different for different individual functions or not equally spaced in T , we follow the methodology proposed by Zhang (2013).
- First we have to reconstruct the functional samples from the observed discrete functional samples using smoothing technique such as regression splines, smoothing splines, P-splines and local polynomial smoothing (see Zhang, 2013, Chapters 2-3). For this purpose, in R we can use the `smooth.spline` function from package `stats` (R Core Team, 2015) or functions given in the packages: `splines` (R Core Team, 2015), `bigsplines` (Helwig, 2016), `pspline` (Ripley, 2015) and `locpol` (Ojeda Cabrera, 2012). After that we discretize each individual function of the reconstructed functional samples on a common grid of time points equally spaced in T , and then the implementations of the tests can be applied to discretized samples.

fmanova package - Permutation Test based on a Basis Function Representation

```
fmanova.ptbfr(x, group.label, basis = c('Fourier', 'own'), own.basis ,  
             criterion = c('BIC', 'eBIC', 'AIC', 'NO'),  
             method = c('dominant', 'min', 'max', 'mean'),  
             minK = 3, maxK = 101, int, gamma.eBIC = 0.5, B = 1000, seed=101)
```

- **x** - the list of data frames or matrices of data, whose each row is a discretized version of a function. The m th element of this list contains the data of m th feature, $m = 1, \dots, p$.
- **group.label** - the vector containing group labels,
- **basis** - the choice of orthonormal basis of functions used in the basis function representation of the data,
- **own.basis** - if **basis** = 'own', the list of length p , whose elements are $n \times (K_m + 1)$ matrices ($m = 1, \dots, p$) with rows containing the coefficients of the basis function representation of the observations,
- **criterion** - the choice of information criterion for selecting the optimum value of K_m ; 'NO' means that K_m are equal to the argument **maxK** defined below,

fmanova package - Permutation Test based on a Basis Function Representation

```
fmanova.ptbfr(x, group.label, basis = c('Fourier', 'own'), own.basis,
             criterion = c('BIC', 'eBIC', 'AIC', 'NO'),
             method = c('dominant', 'min', 'max', 'mean'),
             minK = 3, maxK = 101, int, gamma.eBIC = 0.5, B = 1000, seed=101)
```

- `method` - the choice of method for selecting the common value for all observations from the values of K_m corresponding to all processes,
- `minK` (resp. `maxK`) - the minimal (resp. maximum) value of K_m ,
- `int` - the vector of two elements representing the interval $T = [a, b]$. When it is not specified, it is determined by the design time points.
- `gamma.eBIC` - the $\gamma \in [0, 1]$ parameter in the eBIC,
- `B` - the number of permutation replicates,
- `seed` - the seed given by `set.seed(seed)`.
- The `fmanova.ptbfr` function uses the `fda` package (Ramsay, Wickham, Graves and Hooker, 2014).

fmanova package - Permutation Test based on a Basis Function Representation

- The executions of selecting the optimum values of K_m by some information criterion and the permutation loop are the most time consuming steps. To reduce the computational cost of the procedure, they are parallelized. The parallel execution is handled by `doParallel` package (Revolution Analytics and Steve Weston, 2015).

```
fmanova.ptbfr.parallel(x, group.label, basis = c('Fourier', 'own'), own.basis,
  criterion = c('BIC', 'eBIC', 'AIC', 'NO'),
  method = c('dominant', 'min', 'max', 'mean'),
  minK = 3, maxK = 101, int, gamma.eBIC = 0.5, B = 1000,
  seed = 101, nnodes = 3, ncores = 2)
```

- `nnodes` - the number of nodes to be created in the cluster,
- `ncores` - the number of cores to use for parallel execution.
- The `fmanova.ptbfr` and `fmanova.ptbfr.parallel` functions return a list of the class `fmanovaptbfr` containing the values of the test statistics (`W`, `LH`, `P`, `R`), the p -values (`pvalueW`, `pvalueLH`, `pvalueP`, `pvalueR`) and some of the arguments.

```
fmanova.trp(x, group.label, k = 30, seed = 101, permutation = F, B = 1000)
```

- `x` - the list of data frames or matrices of data, whose each row is a discretized version of a function. The m th element of this list contains the data of m th feature, $m = 1, \dots, p$.
- `group.label` - the vector containing group labels,
- `k` - the vector of numbers of projections,
- `seed` - the seed given by `set.seed(seed)`,
- `permutation` - the logical indicating whether to compute p -values by permutation method,
- `B` - the number of permutation replicates.

fmanova package - Test based on Random Projections

- To reduce the computational cost of the projection loop, it is parallelized by `doParallel` package and the `manova.statistics.quick` function is used to compute the values of the test statistics in the permutation loop. This function is much faster than the `manova` function.

```
fmanova.trp.parallel(x, group.label, k = 30, seed = 101, permutation = F,  
                    B = 1000, nnodes = 3, ncores = 2)
```

- `nnodes` - the number of nodes to be created in the cluster,
- `ncores` - the number of cores to use for parallel execution.
- The `fmanova.trp` and `fmanova.trp.parallel` functions return a list of class `fmanovatr` containing the arguments and the following elements:
 - `data.projections` - the list of length $|k|$, whose elements are lists of $n \times p$ matrices of projections of the observations ($|k|$ - the length of vector k),
 - `pvalues` - the $4 \times |k|$ matrix of p -values of the tests.

fmanova package - auxiliary functions

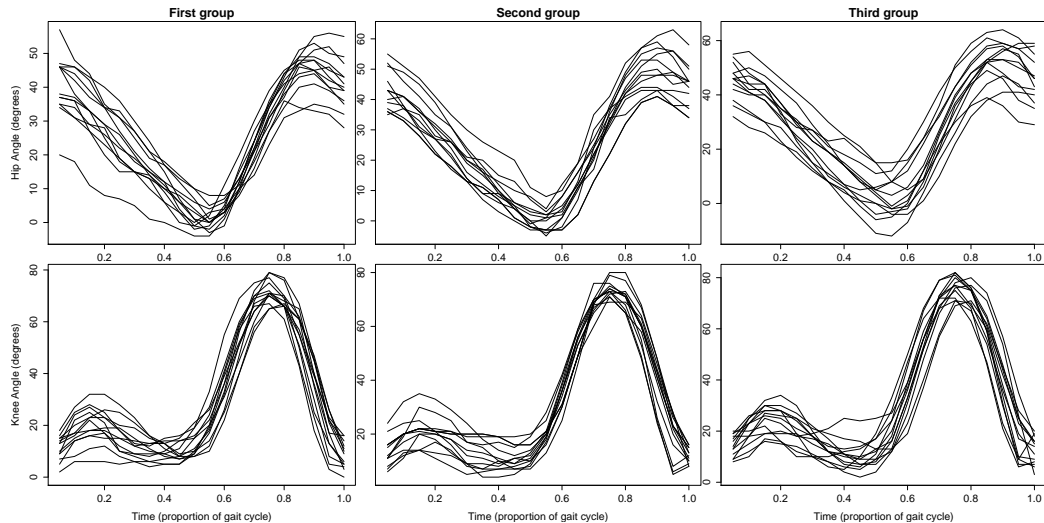
- The p -values of the Wp, LHp, Pp and Rp tests against the number of projections are visualized by the following function using `ggplot2` package (Wickham, 2009).

```
plot.fmanovatr(x.stand , x.perm , withoutRoy = F)
```

- `x.stand` - the result of the `fmanova.trp` or `fmanova.trp.parallel` functions for the standard tests,
- `x.perm` - the result of the `fmanova.trp` or `fmanova.trp.parallel` functions for the permutation tests. We can use only one of the arguments `x.stand` and `x.perm`, or both simultaneously.
- `withoutRoy = F` - the logical indicating whether to plot the p -values of the Rp test.
- `print` - The function prints out the p -values and values of the test statistics for the implemented testing procedures.

- The gait data consist of the angles formed by the hip and knee of each of 39 children over each child's gait cycle. This data set comes from [fda](#) package. The data functions are the simultaneous variation of the hip and knee angles for 39 children at 20 equally spaced time points in $[0.025, 0.975]$. Both sets of functions are periodic.
- For illustrative purposes, we create three groups, so that the first group consists of the observed curves for the first 13 children, the second group of the curves for the next 13 children, and the third group of the curves for the remaining children. The observed curves of these three groups are given in the following figure. The curves of both features seem to be similar in all groups. Hence it is possible that the mean gait curves are not different for the three groups of children. We check this statistically using the proposed tests for the MANOVA problem for three functional samples, each of size 13.

fmanova package - illustrative example



fmanova package - illustrative example

```
data.gait
```

```
[[1]]
```

	0.025	0.075	0.125	...	0.925	0.975
boy1.Hip Angle	37	36	33	...	40	35
boy2.Hip Angle	47	46	42	...	45	43
boy3.Hip Angle	46	44	39	...	50	49
...						
boy39.Hip Angle	48	50	47	...	61	55

```
[[2]]
```

	0.025	0.075	0.125	...	0.925	0.975
boy1.Knee Angle	10	15	18	...	22	11
boy2.Knee Angle	16	25	28	...	23	16
boy3.Knee Angle	18	27	32	...	22	14
...						
boy39.Knee Angle	14	25	32	...	25	15

fmanova package - illustrative example

```
fmanova.ptbfr(data.gait, group.label = rep(1:3, each = 13),  
              basis = 'Fourier', criterion = 'BIC', method = 'dominant',  
              int = c(0.025, 0.975), seed = 101)  
fmanova.ptbfr.parallel(data.gait, group.label = rep(1:3, each = 13),  
                       basis = 'Fourier', criterion = 'BIC', method = 'dominant',  
                       int = c(0.025, 0.975), seed = 101, nnodes = 3, ncores = 2)
```

FMANOVA - Permutation Tests based on a Basis Function Representation
(using B = 1000 permutations)

```
W  = 0.9061231    p-value = 0.496  
LH = 0.1023917    p-value = 0.491  
P  = 0.09497426   p-value = 0.502  
R  = 0.08874477   p-value = 0.368
```

fmanova package - illustrative example

```
xx = fmanova.trp(x = data.gait, group.label = rep(1:3, each = 13),  
                k = c(1, 5, 10, 15, 20), seed = 101)  
yy = fmanova.trp(x = data.gait, group.label = rep(1:3, each = 13),  
                k = c(1, 5, 10, 15, 20), seed = 101, permutation = T)  
xx = fmanova.trp.parallel(x = data.gait, group.label = rep(1:3, each = 13),  
                          k = c(1, 5, 10, 15, 20), seed = 101)  
yy = fmanova.trp.parallel(x = data.gait, group.label = rep(1:3, each = 13),  
                          k = c(1, 5, 10, 15, 20), seed = 101, permutation = T)  
  
plot(x.stand = xx, x.perm = yy)  
plot(x.stand = xx, x.perm = yy, withoutRoy = T)
```

fmanova package - illustrative example

FMANOVA - Tests based on $k = 1$ Random Projections (without permutation)

p-value Wilks = 0.2431773

p-value Lawley-Hotelling = 0.2429649

p-value Pillai = 0.2441983

p-value Roy = 0.07451204

...

FMANOVA - Tests based on $k = 20$ Random Projections (without permutation)

p-value Wilks = 0.3042285

p-value Lawley-Hotelling = 0.3065187

p-value Pillai = 0.3026675

p-value Roy = 0.09969583

fmanova package - illustrative example

FMANOVA - Tests based on $k = 1$ Random Projections (using $B = 1000$ permutations)

p-value Wilks = 0.241

p-value Lawley-Hotelling = 0.237

p-value Pillai = 0.239

p-value Roy = 0.222

...

FMANOVA - Tests based on $k = 20$ Random Projections (using $B = 1000$ permutations)

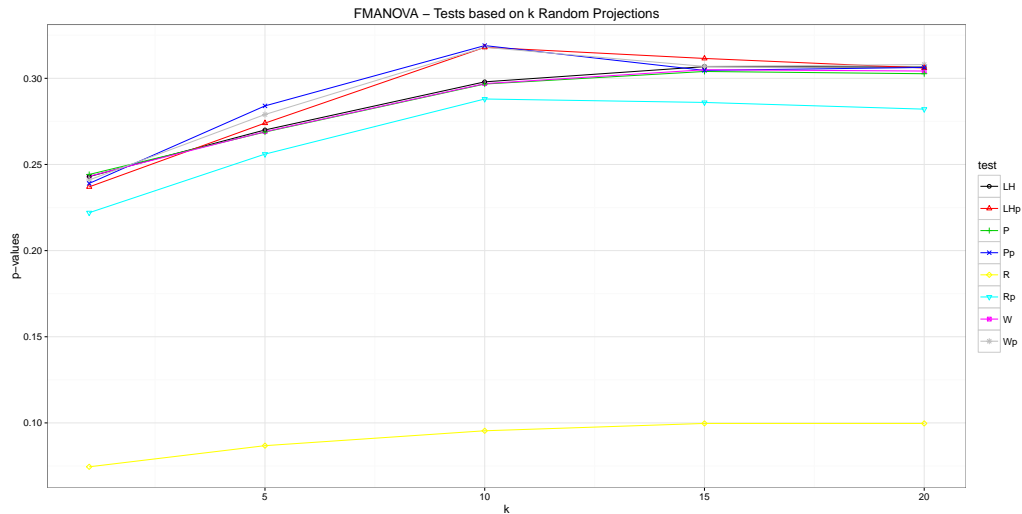
p-value Wilks = 0.308

p-value Lawley-Hotelling = 0.306

p-value Pillai = 0.3063158

p-value Roy = 0.2821053

fmanova package - illustrative example



References

- ① Anderson T.W. (2003) An introduction to multivariate statistical analysis, 3rd edition. Wiley, London.
- ② Benjamini Y., Yekutieli D. (2001) The control of the false discovery rate in multiple testing under dependency. *Ann. Statist.* 29, 1165–1188.
- ③ Cuesta-Albertos J.A., Febrero-Bande M. (2010) A simple multiway ANOVA for functional data. *Test* 19, 537–557.
- ④ Cuesta-Albertos J.A., Fraiman R., Ransford T. (2007) A sharp form of the Cramér-Wold theorem. *J. Theoret. Probab.* 20, 201–209.
- ⑤ Górecki T., Krzyśko M., Waszak Ł. (2014) Functional discriminant coordinates. *Comm. Statist. Theory Methods* 43, 1013–1025.
- ⑥ Górecki T., Smaga Ł. (2015) A comparison of tests for the one-way ANOVA problem for functional data. *Comput. Statist.* 30, 987–1010.
- ⑦ Górecki, T., Smaga, Ł. (2016) Multivariate analysis of variance for functional data. *J. Appl. Stat.* (accepted)

References

- ⑧ Helwig N.E. (2016) bigsplines: Smoothing Splines for Large Samples. R package version 1.0-9. <http://CRAN.R-project.org/package=bigsplines>
- ⑨ Horváth L., Kokoszka P. (2012) Inference for functional data with applications. Springer, New York.
- ⑩ Martínez-Camblor P., Corral N. (2011) Repeated measures analysis for functional data. Comput. Statist. Data Anal. 55, 3244–3256.
- ⑪ Ojeda Cabrera J.L. (2012) locpol: Kernel local polynomial regression. R package version 0.6-0. <http://CRAN.R-project.org/package=locpol>
- ⑫ Ramsay J.O., Hooker G., Graves S. (2009) Functional data analysis with R and MATLAB. Springer, Berlin.
- ⑬ Ramsay J.O., Silverman B.W. (2005) Functional data analysis, 2nd edition. Springer, New York.
- ⑭ Ramsay J.O., Wickham H., Graves S., Hooker G. (2014) fda: Functional data analysis. R package version 2.4.3. <http://CRAN.R-project.org/package=fda>
- ⑮ R Core Team (2015) R: a language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. <http://www.R-project.org/>

- 16 Revolution Analytics and Steve Weston (2015) doParallel: Foreach Parallel Adaptor for the 'parallel' Package. R package version 1.0.10. <http://CRAN.R-project.org/package=doParallel>
- 17 S original by Jim Ramsey. R port by Brian Ripley <ripley@stats.ox.ac.uk>. (2015) pspline: Penalized Smoothing Splines. R package version 1.0-17. <http://CRAN.R-project.org/package=pspline>
- 18 Wickham H. (2009) ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York.
- 19 Zhang J.T. (2013) Analysis of variance for functional data. Chapman & Hall, London.
- 20 Zhang J.T., Liang X. (2014) One-way ANOVA for functional data via globalizing the pointwise F-test. Scand. J. Stat. 41, 51–71.

Thanks for your attention!
Any questions?