

Package: fdatest (via r-universe)

October 12, 2026

Title Interval Wise Testing for Functional Data

Version 2.1.1

Description Implementation of local inferential (domain-selection) procedures for functional data in different frameworks (one- and two-sample tests, ANOVA, linear models). The current version of the package requires functional data evaluated on a common uniform grid or objects of class 'fd' as produced by the 'fda' package and provides the adjusted and unadjusted p-value functions based on nonparametric local tests. The package provides tools for visualization: (i) the p-value heatmap, the plot of the p-value functions and the plot of the functional data.

License GPL (>= 3)

Encoding UTF-8

LazyData true

Imports cli, fda, ggplot2, lifecycle, mirai, patchwork, rlang

LinkingTo cpp11

Suggests tinysnapshot, tinytest

RoxygenNote 7.3.3

Roxygen list(markdown = TRUE)

Depends R (>= 3.5.0)

URL <https://permaverse.github.io/fdatest/>,
<https://github.com/permaverse/fdatest/>

BugReports <https://github.com/permaverse/fdatest/issues>

Config/pak/sysreqs make

Repository <https://alessiapini.r-universe.dev>

Date/Publication 2026-04-15 06:15:52 UTC

RemoteUrl <https://github.com/alessiapini/fdatest>

RemoteRef HEAD

RemoteSha 67bb41c4d8051905a7dd5024e91f5ff2237e1b07

Contents

fdr2	2
functional_anova_test	5
functional_lm_test	8
functional_one_sample_test	11
functional_two_sample_test	13
global_aov	17
global2	19
Globallm	22
ITP1bspline	25
ITP2bspline	27
ITPaovbspline	31
ITPimage	33
ITPlmbspline	34
IWTaov	37
IWTimage	40
IWTlm	41
NASAtemp	44
pct2	44
plot.faov	48
plot.flm	50
plot.fos	52
plot.fts	55
summary.faov	57
summary.flm	58
tw2	60
TWTaov	62
TWTlm	65
Index	68

fdr2

Two population functional Benjamini-Hochberg procedure

Description

The function implements the functional Benjamini Hochberg (fBH) procedure for testing mean differences between two functional populations. Functional data are tested locally and unadjusted and adjusted p-value functions are provided. The unadjusted p-value function controls the point-wise error rate. The adjusted p-value function controls the family-wise error rate asymptotically.

Usage

```
fdr2(
  data1,
  data2,
  mu = 0,
  dx = NULL,
  n_perm = 1000L,
  paired = FALSE,
  alternative = c("two.sided", "less", "greater"),
  standardize = FALSE,
  verbose = FALSE
)

FDR2(
  data1,
  data2,
  mu = 0,
  dx = NULL,
  B = 1000L,
  paired = FALSE,
  alternative = c("two.sided", "less", "greater"),
  statistic = c("Integral", "Integral_std"),
  verbose = FALSE
)
```

Arguments

data1	Either a numeric matrix or an object of class <code>fda:fd</code> specifying the data in the first sample. If the data is provided within a matrix, it should be of shape $n_1 \times J$ and it should contain in each row one of the n_1 functions in the sample and in columns the evaluation of each function on a same uniform grid of size J .
data2	Either a numeric matrix or an object of class <code>fda:fd</code> specifying the data in the second sample. If the data is provided within a matrix, it should be of shape $n_2 \times J$ and it should contain in each row one of the n_2 functions in the sample and in columns the evaluation of each function on a same uniform grid of size J .
mu	Either a numeric value or a numeric vector or an object of class <code>fda:fd</code> specifying the functional mean difference under the null hypothesis. If mu is a constant, then a constant function is used. If mu is a numeric vector, it must correspond to evaluation of the mean difference function on the same grid that has been used to evaluate the data samples. Defaults to 0.
dx	A numeric value specifying the step of the uniform grid on which the data are evaluated. If NULL, the step is automatically inferred from the data. Defaults to NULL.
n_perm	An integer value specifying the number of permutations to use for the local testing procedure. Defaults to 1000L.

paired	A boolean value specifying whether a paired test should be performed. Defaults to FALSE.
alternative	A string specifying the type of alternative hypothesis. Choices are "two.sided", "less" or "greater". Defaults to "two.sided".
standardize	A boolean value specifying whether to standardize the test statistic. Defaults to FALSE.
verbose	A boolean value specifying whether to print the progress of the computation. Defaults to FALSE.
B	An integer value specifying the number of permutations to use for the local testing procedure. Defaults to 1000L.
statistic	A string specifying the test statistic to use. Possible values are: • "Integral": Integral of the squared sample mean difference. • "Max": Maximum of the squared sample mean difference. • "Integral_std": Integral of the squared t-test statistic. • "Max_std": Maximum of the squared t-test statistic. Defaults to "Integral".

Value

An object of class `fts` containing the following components:

- `data`: A numeric matrix of shape $n \times J$ containing the evaluation of the $n = n_1 + n_2$ functions on a **common** uniform grid of size p .
- `group_labels`: An integer vector of size $n = n_1 + n_2$ containing the group membership of each function.
- `mu`: A numeric vector of shape J containing the evaluation of the functional mean difference under the null hypothesis on the same uniform grid used to evaluate the functional samples.
- `unadjusted_pvalues`: A numeric vector of size J containing the evaluation of the unadjusted p-value function on the **same** uniform grid used to evaluate the functional samples.
- `adjusted_pvalues`: A numeric vector of size J containing the evaluation of the adjusted p-value function on the **same** uniform grid used to evaluate the functional samples.
- `correction_method`: A string containing the correction method used to compute the adjusted p-value function.

Optionally, the list may contain the following components:

- `global_pvalue`: A numeric value containing the global p-value. Only present if the correction argument is set to "Global".
- `pvalue_matrix`: A numeric matrix of shape $p \times p$ containing the p-values of the interval-wise tests. Element i, j contains the p-value of the test performed on the interval indexed by $j, j + 1, \dots, j + (p - i)$. Only present if the correction argument is set to "IWT".

References

- Lundtorp Olsen, Niels, Alessia Pini, and Simone Vantini. 2021. "False discovery rate for functional data." *TEST* 30, 784–809.

See Also

`global2()`, `tw2()`, `pct2()`, `iwt2()` for calling directly one of the other tests, `functional_two_sample_test()` for calling the interface test and `plot.fts()` for plotting the results.

Examples

```
# Performing the fBH for two populations

FDR_result <- fdr2(NASAtemp$paris, NASAtemp$milan)

# Plotting the results of the fBH
plot(
  FDR_result,
  xrange = c(0, 12),
  title = "FDR results for testing mean differences"
)

# Selecting the significant components at 5% level
which(FDR_result$adjusted_pvalues < 0.05)
```

functional_anova_test *Local testing procedures for the functional analysis of variance*

Description

Implements local testing procedures for comparing the mean functions of multiple functional populations in a one-way or multi-way functional analysis of variance (FANOVA) framework. Functional data are tested locally and unadjusted and adjusted p-value functions are provided. The unadjusted p-value function controls the point-wise error rate. The adjusted p-value function can be computed according to the following methods:

- global testing (controlling the FWER weakly)
- interval-wise testing (controlling the interval-wise error rate)
- threshold-wise testing (controlling the FWER asymptotically)

Usage

```
functional_anova_test(
  formula,
  correction = c("IWT", "TWT", "Global"),
  dx = NULL,
  B = 1000L,
  method = c("residuals", "responses"),
  recycle = TRUE,
  stat = c("Integral", "Max")
)
```

Arguments

formula	An object of class <code>stats::formula</code> (or one that can be coerced to that class) specifying the model to be fitted in a symbolic fashion. The response (left-hand side) can be either a matrix of dimension $n \times J$ containing the pointwise evaluations of n functions on the same grid of J points, or an object of class <code>fda::fd</code> .
correction	A string specifying the method used to calculate the adjusted p-value function. Choices are "Global" for global testing, "IWT" for interval-wise testing, or "TWT" for threshold-wise testing.
dx	A numeric value specifying the discretization step of the grid used to evaluate functional data when it is provided as objects of class <code>fda::fd</code> . Defaults to NULL, in which case a default value of 0.01 is used which corresponds to a grid of size 100L. Unused if functional data is provided in the form of matrices.
B	An integer value specifying the number of permutations used to evaluate the p-values of the permutation tests. Defaults to 1000L. Passed as <code>n_perm</code> in <code>iwt_aov()</code> , <code>twt_aov()</code> and <code>global_aov()</code> .
method	A string specifying the permutation scheme. "residuals" permutes residuals under the reduced model (Freedman-Lane scheme); "responses" permutes the responses (Manly scheme). Defaults to "residuals".
recycle	A boolean value specifying whether the recycled version of the interval-wise testing procedure should be used. See Pini and Vantini (2017) for details. Defaults to TRUE.
stat	A string specifying the test statistic used for the global test. "Integral" uses the integral of the F-statistic over the domain; "Max" uses the maximum. Defaults to "Integral".

Value

An object of class `faov` containing the following components:

- `call`: The matched call.
- `design_matrix`: The design matrix of the functional ANOVA model.
- `unadjusted_pval_F`: A numeric vector of length J containing the unadjusted p-value function of the global F-test evaluated on the grid.
- `adjusted_pval_F`: A numeric vector of length J containing the adjusted p-value function of the global F-test evaluated on the grid.
- `unadjusted_pval_factors`: A numeric matrix with one row per factor containing the unadjusted p-value functions of the per-factor F-tests.
- `adjusted_pval_factors`: A numeric matrix with one row per factor containing the adjusted p-value functions of the per-factor F-tests.
- `data_eval`: A numeric matrix containing the functional data evaluated on the grid.
- `coeff_regr_eval`: A numeric matrix containing the functional regression coefficients evaluated on the grid.
- `fitted_eval`: A numeric matrix containing the fitted values of the functional regression evaluated on the grid.

- `residuals_eval`: A numeric matrix containing the residuals of the functional regression evaluated on the grid.
- `R2_eval`: A numeric vector containing the functional R-squared evaluated on the grid.

Optionally, the list may contain the following components:

- `pval_matrix_F`: A matrix of dimensions $p \times p$ of p-values of the interval-wise F-tests. Element (i, j) contains the p-value of the test on the interval $(j, j + 1, \dots, j + (p - i))$. Present only if correction is "IWT".
- `pval_matrix_factors`: An array of dimensions $L \times p \times p$ of p-values of the per-factor interval-wise F-tests. Element (l, i, j) contains the p-value of the joint test on factor l and interval $(j, j + 1, \dots, j + (p - i))$. Present only if correction is "IWT".
- `global_pval_F`: Global p-value of the overall F-test. Present only if correction is "Global".
- `global_pval_factors`: A numeric vector of global p-values of the per-factor F-tests. Present only if correction is "Global".

References

- Abramowicz, K., Pini, A., Schelin, L., Stamm, A., & Vantini, S. (2022). "Domain selection and familywise error rate for functional data: A unified framework. *Biometrics* 79(2), 1119-1132.
- D. Freedman and D. Lane (1983). A Nonstochastic Interpretation of Reported Significance Levels. *Journal of Business & Economic Statistics* 1.4, 292-298.
- B. F. J. Manly (2006). *Randomization, Bootstrap and Monte Carlo Methods in Biology*. Vol. 70. CRC Press.
- Pini, A., & Vantini, S. (2017). Interval-wise testing for functional data. *Journal of Nonparametric Statistics*, 29(2), 407-424.
- Pini, A., Vantini, S., Colosimo, B. M., & Grasso, M. (2018). Domain-selective functional analysis of variance for supervised statistical profile monitoring of signal data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 67(1), 55-81.
- Abramowicz, K., Hager, C. K., Pini, A., Schelin, L., Sjøstedt de Luna, S., & Vantini, S. (2018). Nonparametric inference for functional-on-scalar linear models applied to knee kinematic hop data after injury of the anterior cruciate ligament. *Scandinavian Journal of Statistics* 45(4), 1036-1061.

See Also

[iwt_aov\(\)](#), [twf_aov\(\)](#) and [global_aov\(\)](#) for calling a specific correction directly. [plot.faov\(\)](#) for plotting the results and [summary.faov\(\)](#) for summarizing the results.

Examples

```
temperature <- rbind(NASAtemp$milan[, 1:100], NASAtemp$paris[, 1:100])
groups <- c(rep(0, 22), rep(1, 22))

# Performing the TWT for multiple populations
TWT_result <- functional_anova_test(
  temperature ~ groups,
  correction = "TWT",
  B = 10L
```

```

)

# Plotting the results of the TWT
plot(
  TWT_result,
  xrange = c(0, 12),
  main = "TWT results for testing mean differences"
)

# Selecting the significant components at 5% level
which(TWT_result$adjusted_pval_F < 0.05)

# Performing the IWT for multiple populations
IWT_result <- functional_anova_test(
  temperature ~ groups,
  correction = "IWT",
  B = 10L
)

# Plotting the results of the IWT
plot(
  IWT_result,
  xrange = c(0, 12),
  main = "IWT results for testing mean differences"
)

# Selecting the significant components at 5% level
which(IWT_result$adjusted_pval_F < 0.05)

```

functional_lm_test	<i>Local testing procedures for functional-on-scalar linear models</i>
--------------------	--

Description

Implements local testing procedures for testing the significance of the effects of scalar covariates on a functional response in a functional-on-scalar linear model. Functional data are tested locally and unadjusted and adjusted p-value functions are provided. The unadjusted p-value function controls the point-wise error rate. The adjusted p-value function can be computed according to the following methods:

- global testing (controlling the FWER weakly)
- interval-wise testing (controlling the interval-wise error rate)
- threshold-wise testing (controlling the FWER asymptotically)

Usage

```

functional_lm_test(
  formula,
  correction = c("IWT", "TWT", "Global"),

```

```

dx = NULL,
B = 1000L,
method = c("residuals", "responses"),
recycle = TRUE,
stat = c("Integral", "Max")
)

```

Arguments

formula	An object of class <code>stats::formula</code> (or one that can be coerced to that class) specifying the model to be fitted in a symbolic fashion. The response (left-hand side) can be either a matrix of dimension $n \times J$ containing the pointwise evaluations of n functions on the same grid of J points, or an object of class <code>fda::fd</code> .
correction	A string specifying the method used to calculate the adjusted p-value function. Choices are "Global" for global testing, "IWT" for interval-wise testing, or "TWT" for threshold-wise testing.
dx	A numeric value specifying the discretization step of the grid used to evaluate functional data when it is provided as objects of class <code>fda::fd</code> . Defaults to NULL, in which case a default value of 0.01 is used which corresponds to a grid of size 100L. Unused if functional data is provided in the form of matrices.
B	An integer value specifying the number of permutations used to evaluate the p-values of the permutation tests. Defaults to 1000L. Passed as <code>n_perm</code> in <code>iwt_aov()</code> , <code>twt_aov()</code> and <code>global_aov()</code> .
method	A string specifying the permutation scheme. "residuals" permutes residuals under the reduced model (Freedman-Lane scheme); "responses" permutes the responses (Manly scheme). Defaults to "residuals".
recycle	A boolean value specifying whether the recycled version of the interval-wise testing procedure should be used. See Pini and Vantini (2017) for details. Defaults to TRUE.
stat	A string specifying the test statistic used for the global test. "Integral" uses the integral of the F-statistic over the domain; "Max" uses the maximum. Defaults to "Integral".

Value

An object of class `flm` containing the following components:

- `call`: The matched call.
- `design_matrix`: The design matrix of the functional-on-scalar linear model.
- `unadjusted_pval_F`: A numeric vector of length J containing the unadjusted p-value function of the global F-test evaluated on the grid.
- `adjusted_pval_F`: A numeric vector of length J containing the adjusted p-value function of the global F-test evaluated on the grid.
- `unadjusted_pval_part`: A numeric matrix with one row per model term containing the unadjusted p-value functions of the per-term t-tests.

- `adjusted_pval_part`: A numeric matrix with one row per model term containing the adjusted p-value functions of the per-term t-tests.
- `data_eval`: A numeric matrix containing the functional response evaluated on the grid.
- `coeff_regr_eval`: A numeric matrix containing the functional regression coefficients evaluated on the grid.
- `fitted_eval`: A numeric matrix containing the fitted values of the functional regression evaluated on the grid.
- `residuals_eval`: A numeric matrix containing the residuals of the functional regression evaluated on the grid.
- `R2_eval`: A numeric vector containing the functional R-squared evaluated on the grid.

Optionally, the list may contain the following components:

- `pval_matrix_F`: A matrix of dimensions $p \times p$ of p-values of the interval-wise F-tests. Element (i, j) contains the p-value of the test on the interval $(j, j + 1, \dots, j + (p - i))$. Present only if correction is "IWT".
- `pval_matrix_part`: An array of dimensions $(L + 1) \times p \times p$ of p-values of the per-term interval-wise t-tests. Element (l, i, j) contains the p-value of the joint test on term l and interval $(j, j + 1, \dots, j + (p - i))$. Present only if correction is "IWT".
- `global_pval_F`: Global p-value of the overall F-test. Present only if correction is "Global".
- `global_pval_part`: A numeric vector of global p-values of the per-term t-tests. Present only if correction is "Global".

References

- Abramowicz, K., Pini, A., Schelin, L., Stamm, A., & Vantini, S. (2022). "Domain selection and familywise error rate for functional data: A unified framework. *Biometrics* 79(2), 1119-1132.
- D. Freedman and D. Lane (1983). A Nonstochastic Interpretation of Reported Significance Levels. *Journal of Business & Economic Statistics* 1.4, 292-298.
- B. F. J. Manly (2006). *Randomization, Bootstrap and Monte Carlo Methods in Biology*. Vol. 70. CRC Press.
- Pini, A., & Vantini, S. (2017). Interval-wise testing for functional data. *Journal of Nonparametric Statistics*, 29(2), 407-424.
- Pini, A., Vantini, S., Colosimo, B. M., & Grasso, M. (2018). Domain-selective functional analysis of variance for supervised statistical profile monitoring of signal data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 67(1), 55-81.
- Abramowicz, K., Hager, C. K., Pini, A., Schelin, L., Sjøstedt de Luna, S., & Vantini, S. (2018). Nonparametric inference for functional-on-scalar linear models applied to knee kinematic hop data after injury of the anterior cruciate ligament. *Scandinavian Journal of Statistics* 45(4), 1036-1061.

See Also

`iwt_lm()`, `twt_lm()` and `global_lm()` for calling a specific correction directly. `plot.flm()` for plotting the results and `summary.flm()` for summarizing the results.

Examples

```
# Defining the covariates
temperature <- rbind(NASAtemp$milan, NASAtemp$paris)
groups <- c(rep(0, 22), rep(1, 22))

# Performing the TWT
TWT_result <- functional_lm_test(
  temperature ~ groups,
  correction = "TWT",
  B = 10L
)
# Summary of the TWT results
summary(TWT_result)
```

functional_one_sample_test

Local testing procedures for the functional one-sample test

Description

The function implements local testing procedures for testing the center of symmetry of a functional population. Functional data are tested locally and unadjusted and adjusted p-value functions are provided. The unadjusted p-value function controls the point-wise error rate. The adjusted p-value function can be computed according to the following methods:

- interval-wise testing (controlling the interval-wise error rate)

Usage

```
functional_one_sample_test(
  data,
  correction = c("IWT"),
  mu = 0,
  dx = NULL,
  n_perm = 1000L,
  verbose = FALSE,
  aggregation_strategy = c("integral", "max"),
  recycle = TRUE
)
```

Arguments

- | | |
|------------|--|
| data | Either a numeric matrix or an object of class <code>fda::fd</code> specifying the sample data. If the data is provided within a matrix, it should be of shape $n \times J$ and it should contain in each row one of the n functions in the sample and in columns the evaluation of each function on a same uniform grid of size J . |
| correction | A string specifying the correction method to perform the local functional testing procedure and adjust the p-value function. Currently only "IWT" is available. |

mu	Either a numeric value or a numeric vector or an object of class <code>fda::fd</code> specifying the functional center of symmetry under the null hypothesis. If mu is a constant, then a constant function is used. If mu is a numeric vector, it must correspond to evaluations of the center function on the same grid that has been used to evaluate the data sample. Defaults to $\emptyset$.
dx	A numeric value specifying the step of the uniform grid on which the data are evaluated. If NULL, the step is automatically inferred from the data. Defaults to NULL.
n_perm	An integer value specifying the number of permutations to use for the local testing procedure. Defaults to 1000L.
verbose	A boolean value specifying whether to print the progress of the computation. Defaults to FALSE.
aggregation_strategy	A string specifying the strategy to aggregate the point-wise test statistics for the correction procedure. Possible values are "integral" and "max". Defaults to "integral".
recycle	A boolean value specifying whether to recycle the test statistic values across permutations for the IWT procedure. Defaults to TRUE.

Value

An object of class `fos` containing the following components:

- `data`: A numeric matrix of shape $n \times J$ containing the evaluation of the n functions on a uniform grid of size J .
- `mu`: A numeric vector of shape J containing the evaluation of the functional center of symmetry under the null hypothesis on the same uniform grid used to evaluate the functional sample.
- `unadjusted_pvalues`: A numeric vector of size J containing the evaluation of the unadjusted p-value function on the same uniform grid used to evaluate the functional sample.
- `adjusted_pvalues`: A numeric vector of size J containing the evaluation of the adjusted p-value function on the same uniform grid used to evaluate the functional sample.
- `correction_method`: A string containing the correction method used to compute the adjusted p-value function.

Optionally, the list may contain the following component:

- `pvalue_matrix`: A numeric matrix of shape $p \times p$ containing the p-values of the interval-wise tests. Element i, j contains the p-value of the test performed on the interval indexed by $j, j + 1, \dots, j + (p - i)$. Only present if the `correction` argument is set to "IWT".

References

For the interval-wise testing procedure:

- Pini, Alessia, and Simone Vantini. 2016. "The interval testing procedure: a general framework for inference in functional data analysis." *Biometrics* 72 (3): 835–845.
- Pini, Alessia, and Simone Vantini. 2017. "Interval-Wise Testing for Functional Data." *Journal of Nonparametric Statistics* 29 (2): 407–24.

See Also

`iwt1()` for calling directly the IWT test and `plot.fos()` and `autoplot.fos()` for plotting the results.

Examples

```
# Performing the IWT for one population
IWT_result <- functional_one_sample_test(
  NASAtemp$paris, mu = 4, n_perm = 10L
)

# Plotting the results of the IWT
IWTimage(IWT_result, abscissa_range = c(0, 12))

# Selecting the significant components at 5% level
which(IWT_result$adjusted_pvalues < 0.05)
```

functional_two_sample_test

Local testing procedures for the functional two-sample test

Description

The function implements local testing procedures for testing mean differences between two functional populations. Functional data are tested locally and unadjusted and adjusted p-value functions are provided. The unadjusted p-value function controls the point-wise error rate. The adjusted p-value function can be computed according to the following methods:

- global testing (controlling the FWER weakly)
- interval-wise testing (controlling the interval-wise error rate)
- threshold-wise testing (controlling the FWER asymptotically)
- partition closed testing (controlling the FWER on a partition)
- functional Benjamini Hochberg (controlling the FDR)

Usage

```
functional_two_sample_test(
  data1,
  data2,
  correction = c("Global", "IWT", "TWT", "PCT", "FDR"),
  mu = 0,
  dx = NULL,
  n_perm = 1000L,
  paired = FALSE,
  alternative = c("two.sided", "less", "greater"),
  standardize = FALSE,
```

```

    verbose = FALSE,
    aggregation_strategy = c("integral", "max"),
    recycle = TRUE,
    partition = NULL
  )

```

Arguments

data1	Either a numeric matrix or an object of class <code>fda::fd</code> specifying the data in the first sample. If the data is provided within a matrix, it should be of shape $n_1 \times J$ and it should contain in each row one of the n_1 functions in the sample and in columns the evaluation of each function on a same uniform grid of size J .
data2	Either a numeric matrix or an object of class <code>fda::fd</code> specifying the data in the second sample. If the data is provided within a matrix, it should be of shape $n_2 \times J$ and it should contain in each row one of the n_2 functions in the sample and in columns the evaluation of each function on a same uniform grid of size J .
correction	A string specifying the correction method to perform the local functional testing procedure and adjust the p-value function. Choices are "Global", "IWT", "TWT", "PCT" or "FDR".
mu	Either a numeric value or a numeric vector or an object of class <code>fda::fd</code> specifying the functional mean difference under the null hypothesis. If mu is a constant, then a constant function is used. If mu is a numeric vector, it must correspond to evaluation of the mean difference function on the same grid that has been used to evaluate the data samples. Defaults to $\emptyset$.
dx	A numeric value specifying the step of the uniform grid on which the data are evaluated. If NULL, the step is automatically inferred from the data. Defaults to NULL.
n_perm	An integer value specifying the number of permutations to use for the local testing procedure. Defaults to 1000L.
paired	A boolean value specifying whether a paired test should be performed. Defaults to FALSE.
alternative	A string specifying the type of alternative hypothesis. Choices are "two.sided", "less" or "greater". Defaults to "two.sided".
standardize	A boolean value specifying whether to standardize the test statistic. Defaults to FALSE.
verbose	A boolean value specifying whether to print the progress of the computation. Defaults to FALSE.
aggregation_strategy	A string specifying the strategy to aggregate the point-wise test statistics for the correction procedure. Possible values are "integral" and "max". Defaults to "integral".
recycle	A boolean value specifying whether to recycle the test statistic values across permutations for the IWT procedure. Defaults to TRUE.
partition	An integer vector of length J specifying the membership of each point of the domain to an element of the partition. Only used and must be set if the correction argument is set to "PCT".

Value

An object of class `fts` containing the following components:

- `data`: A numeric matrix of shape $n \times J$ containing the evaluation of the $n = n_1 + n_2$ functions on a **common** uniform grid of size p .
- `group_labels`: An integer vector of size $n = n_1 + n_2$ containing the group membership of each function.
- `mu`: A numeric vector of shape J containing the evaluation of the functional mean difference under the null hypothesis on the same uniform grid used to evaluate the functional samples.
- `unadjusted_pvalues`: A numeric vector of size J containing the evaluation of the unadjusted p-value function on the **same** uniform grid used to evaluate the functional samples.
- `adjusted_pvalues`: A numeric vector of size J containing the evaluation of the adjusted p-value function on the **same** uniform grid used to evaluate the functional samples.
- `correction_method`: A string containing the correction method used to compute the adjusted p-value function.

Optionally, the list may contain the following components:

- `global_pvalue`: A numeric value containing the global p-value. Only present if the correction argument is set to "Global".
- `pvalue_matrix`: A numeric matrix of shape $p \times p$ containing the p-values of the interval-wise tests. Element i, j contains the p-value of the test performed on the interval indexed by $j, j + 1, \dots, j + (p - i)$. Only present if the correction argument is set to "IWT".

References

For the global testing procedure:

- Hall, Peter, and Nader Tajvidi. 2002. "Permutation Tests for Equality of Distributions in High-Dimensional Settings." *Biometrika* 89 (2): 359–74.
- Pini, Alessia, Aymeric Stamm, and Simone Vantini. 2018. "Hotelling's T2 in Separable Hilbert Spaces." *Journal of Multivariate Analysis* 167: 284–305.

For the partition closed testing procedure:

- Vsevolozhskaya, Olga A, Mark C Greenwood, GJ Bellante, Scott L Powell, Rick L Lawrence, and Kevin S Repasky. 2013. "Combining Functions and the Closure Principle for Performing Follow-up Tests in Functional Analysis of Variance." *Computational Statistics & Data Analysis* 67: 175–84.
- Vsevolozhskaya, Olga, Mark Greenwood, and Dmitri Holodov. 2014. "Pairwise comparison of treatment levels in functional analysis of variance with application to erythrocyte hemolysis." *The Annals of Applied Statistics* 8 (2): 905–25. <https://doi.org/10.1214/14-AOAS723>.

For the interval-wise testing procedure:

- Pini, Alessia, and Simone Vantini. 2016. "The interval testing procedure: a general framework for inference in functional data analysis." *Biometrics* 72 (3): 835–845.
- Pini, Alessia, and Simone Vantini. 2017. "Interval-Wise Testing for Functional Data." *Journal of Nonparametric Statistics* 29 (2): 407–24.

- Pini, Alessia, Simone Vantini, Bianca Maria Colosimo, and Marco Grasso. 2018. "Domain-Selective Functional Analysis of Variance for Supervised Statistical Profile Monitoring of Signal Data." *Journal of the Royal Statistical Society Series C: Applied Statistics* 67 (1): 55–81.
- Abramowicz, Konrad, Charlotte K Häger, Alessia Pini, Lina Schelin, Sara Sjöstedt de Luna, and Simone Vantini. 2018. "Nonparametric Inference for Functional-on-Scalar Linear Models Applied to Knee Kinematic Hop Data After Injury of the Anterior Cruciate Ligament." *Scandinavian Journal of Statistics* 45 (4): 1036–61.

For the threshold-wise testing procedure:

- Abramowicz, Konrad, Alessia Pini, Lina Schelin, Sara Sjöstedt de Luna, Aymeric Stamm, and Simone Vantini. 2023. "Domain Selection and Familywise Error Rate for Functional Data: A Unified Framework." *Biometrics* 79 (2): 1119–32.

For the functional Benjamini-Hochberg procedure:

- Lundtorp Olsen, Niels, Alessia Pini, and Simone Vantini. 2021. "False discovery rate for functional data." *TEST* 30, 784–809.

See Also

`global2()`, `iwt2()`, `tw2()`, `pct2()`, `fdr2()` for calling directly one specific test and `plot.fts()` for plotting the results.

Examples

```
# Performing the TWT for two populations
TWT_result <- functional_two_sample_test(
  NASAtemp$paris, NASAtemp$milan,
  correction = "TWT", n_perm = 10L
)

# Plotting the results of the TWT
plot(
  TWT_result,
  xrange = c(0, 12),
  title = "TWT results for testing mean differences"
)

# Selecting the significant components at 5% level
which(TWT_result$adjusted_pvalues < 0.05)

# Performing the IWT for two populations
IWT_result <- functional_two_sample_test(
  NASAtemp$paris, NASAtemp$milan,
  correction = "IWT", n_perm = 10L
)

# Plotting the results of the IWT
plot(
  IWT_result,
  xrange = c(0, 12),
```

```

    title = "IWT results for testing mean differences"
  )

  # Selecting the significant components at 5% level
  which(IWT_result$adjusted_pvalues < 0.05)

```

global_aov

*Global testing procedure for testing functional analysis of variance***Description**

Implements the Global Testing procedure for comparing the mean functions of several functional populations in a one-way or multi-way functional analysis of variance framework. Functional data are tested globally. The adjusted p-value function is constant and controls the family-wise error rate weakly.

Usage

```

global_aov(
  formula,
  dx = NULL,
  n_perm = 1000L,
  method = c("residuals", "responses"),
  stat = c("Integral", "Max")
)

Globalaov(
  formula,
  dx = NULL,
  B = 1000L,
  method = c("residuals", "responses"),
  stat = c("Integral", "Max")
)

```

Arguments

formula	An object of class <code>stats::formula</code> (or one that can be coerced to that class) specifying the model to be fitted in a symbolic fashion. The response (left-hand side) can be either a matrix of dimension $n \times J$ containing the pointwise evaluations of n functions on the same grid of J points, or an object of class <code>fda::fd</code> .
dx	A numeric value specifying the discretization step of the grid used to evaluate functional data when it is provided as objects of class <code>fda::fd</code> . Defaults to <code>NULL</code> , in which case a default value of <code>0.01</code> is used which corresponds to a grid of size <code>100L</code> . Unused if functional data is provided in the form of matrices.
n_perm	An integer value specifying the number of permutations for the permutation tests. Defaults to <code>1000L</code> .

method	A string specifying the permutation scheme. "residuals" permutes residuals under the reduced model (Freedman-Lane scheme); "responses" permutes the responses (Manly scheme). Defaults to "residuals".
stat	A string specifying the test statistic used for the global test. "Integral" uses the integral of the F-statistic over the domain; "Max" uses the maximum. Defaults to "Integral".
B	An integer value specifying the number of permutations used to evaluate the p-values of the permutation tests. Defaults to 1000L. Passed as n_perm in <code>iwt_aov()</code> , <code>twt_aov()</code> and <code>global_aov()</code> .

Value

An object of class `faov` containing the following components:

- `call`: The matched call.
- `design_matrix`: The design matrix of the functional ANOVA model.
- `unadjusted_pval_F`: A numeric vector of length J containing the unadjusted p-value function of the global F-test evaluated on the grid.
- `adjusted_pval_F`: A numeric vector of length J containing the adjusted p-value function of the global F-test evaluated on the grid.
- `unadjusted_pval_factors`: A numeric matrix with one row per factor containing the unadjusted p-value functions of the per-factor F-tests.
- `adjusted_pval_factors`: A numeric matrix with one row per factor containing the adjusted p-value functions of the per-factor F-tests.
- `data_eval`: A numeric matrix containing the functional data evaluated on the grid.
- `coeff_regr_eval`: A numeric matrix containing the functional regression coefficients evaluated on the grid.
- `fitted_eval`: A numeric matrix containing the fitted values of the functional regression evaluated on the grid.
- `residuals_eval`: A numeric matrix containing the residuals of the functional regression evaluated on the grid.
- `R2_eval`: A numeric vector containing the functional R-squared evaluated on the grid.

Optionally, the list may contain the following components:

- `pval_matrix_F`: A matrix of dimensions $p \times p$ of p-values of the interval-wise F-tests. Element (i, j) contains the p-value of the test on the interval $(j, j + 1, \dots, j + (p - i))$. Present only if correction is "IWT".
- `pval_matrix_factors`: An array of dimensions $L \times p \times p$ of p-values of the per-factor interval-wise F-tests. Element (l, i, j) contains the p-value of the joint test on factor l and interval $(j, j + 1, \dots, j + (p - i))$. Present only if correction is "IWT".
- `global_pval_F`: Global p-value of the overall F-test. Present only if correction is "Global".
- `global_pval_factors`: A numeric vector of global p-values of the per-factor F-tests. Present only if correction is "Global".

References

- Abramowicz, K., Pini, A., Schelin, L., Stamm, A., & Vantini, S. (2022). "Domain selection and familywise error rate for functional data: A unified framework. *Biometrics* 79(2), 1119-1132.
- D. Freedman and D. Lane (1983). A Nonstochastic Interpretation of Reported Significance Levels. *Journal of Business & Economic Statistics* 1.4, 292-298.
- B. F. J. Manly (2006). Randomization, *Bootstrap and Monte Carlo Methods in Biology*. Vol. 70. CRC Press.

See Also

[iwt_aov\(\)](#), [twi_aov\(\)](#) and [global_aov\(\)](#) for calling a specific correction directly. [plot_faov\(\)](#) for plotting the results and [summary_faov\(\)](#) for summarizing the results.

Examples

```
temperature <- rbind(NASAtemp$milan, NASAtemp$paris)
groups <- c(rep(0, 22), rep(1, 22))

# Performing the test
Global_result <- global_aov(temperature ~ groups, n_perm = 1000L)

# Summary of the test results
summary(Global_result)

# Plot of the results
layout(1)
plot(Global_result)

# All graphics on the same device
layout(matrix(1:4, nrow = 2, byrow = FALSE))
plot(
  Global_result,
  main = "NASA data",
  plot_adjval = TRUE,
  xlab = "Day",
  xrange = c(1, 365)
)
```

Description

The function implements the Global Testing procedure for testing mean differences between two functional populations. Functional data are tested globally and a global p-value and a constant adjusted p-value function are provided. The adjusted p-value function controls the family-wise error rate weakly.

Usage

```

global2(
  data1,
  data2,
  mu = 0,
  dx = NULL,
  n_perm = 1000L,
  paired = FALSE,
  alternative = c("two.sided", "less", "greater"),
  standardize = FALSE,
  verbose = FALSE,
  aggregation_strategy = c("integral", "max")
)

Global2(
  data1,
  data2,
  mu = 0,
  dx = NULL,
  B = 1000L,
  paired = FALSE,
  alternative = c("two.sided", "less", "greater"),
  statistic = c("Integral", "Max", "Integral_std", "Max_std"),
  verbose = FALSE
)

```

Arguments

data1	Either a numeric matrix or an object of class <code>fda::fd</code> specifying the data in the first sample. If the data is provided within a matrix, it should be of shape $n_1 \times J$ and it should contain in each row one of the n_1 functions in the sample and in columns the evaluation of each function on a same uniform grid of size J .
data2	Either a numeric matrix or an object of class <code>fda::fd</code> specifying the data in the second sample. If the data is provided within a matrix, it should be of shape $n_2 \times J$ and it should contain in each row one of the n_2 functions in the sample and in columns the evaluation of each function on a same uniform grid of size J .
mu	Either a numeric value or a numeric vector or an object of class <code>fda::fd</code> specifying the functional mean difference under the null hypothesis. If mu is a constant, then a constant function is used. If mu is a numeric vector, it must correspond to evaluation of the mean difference function on the same grid that has been used to evaluate the data samples. Defaults to 0.
dx	A numeric value specifying the step of the uniform grid on which the data are evaluated. If NULL, the step is automatically inferred from the data. Defaults to NULL.
n_perm	An integer value specifying the number of permutations to use for the local testing procedure. Defaults to 1000L.

paired	A boolean value specifying whether a paired test should be performed. Defaults to FALSE.
alternative	A string specifying the type of alternative hypothesis. Choices are "two.sided", "less" or "greater". Defaults to "two.sided".
standardize	A boolean value specifying whether to standardize the test statistic. Defaults to FALSE.
verbose	A boolean value specifying whether to print the progress of the computation. Defaults to FALSE.
aggregation_strategy	A string specifying the strategy to aggregate the point-wise test statistics for the correction procedure. Possible values are "integral" and "max". Defaults to "integral".
B	An integer value specifying the number of permutations to use for the local testing procedure. Defaults to 1000L.
statistic	A string specifying the test statistic to use. Possible values are: • "Integral": Integral of the squared sample mean difference. • "Max": Maximum of the squared sample mean difference. • "Integral_std": Integral of the squared t-test statistic. • "Max_std": Maximum of the squared t-test statistic. Defaults to "Integral".

Value

An object of class `fts` containing the following components:

- `data`: A numeric matrix of shape $n \times J$ containing the evaluation of the $n = n_1 + n_2$ functions on a **common** uniform grid of size p .
- `group_labels`: An integer vector of size $n = n_1 + n_2$ containing the group membership of each function.
- `mu`: A numeric vector of shape J containing the evaluation of the functional mean difference under the null hypothesis on the same uniform grid used to evaluate the functional samples.
- `unadjusted_pvalues`: A numeric vector of size J containing the evaluation of the unadjusted p-value function on the **same** uniform grid used to evaluate the functional samples.
- `adjusted_pvalues`: A numeric vector of size J containing the evaluation of the adjusted p-value function on the **same** uniform grid used to evaluate the functional samples.
- `correction_method`: A string containing the correction method used to compute the adjusted p-value function.

Optionally, the list may contain the following components:

- `global_pvalue`: A numeric value containing the global p-value. Only present if the correction argument is set to "Global".
- `pvalue_matrix`: A numeric matrix of shape $p \times p$ containing the p-values of the interval-wise tests. Element i, j contains the p-value of the test performed on the interval indexed by $j, j + 1, \dots, j + (p - i)$. Only present if the correction argument is set to "IWT".

References

- Hall, Peter, and Nader Tajvidi. 2002. “Permutation Tests for Equality of Distributions in High-Dimensional Settings.” *Biometrika* 89 (2): 359–74.
- Pini, Alessia, Aymeric Stamm, and Simone Vantini. 2018. “Hotelling’s T2 in Separable Hilbert Spaces.” *Journal of Multivariate Analysis* 167: 284–305.

See Also

`iwt2()`, `tw2()`, `pct2()`, `fdr2()` for calling directly one of the other tests, `functional_two_sample_test()` for calling the interface test and `plot.fts()` for plotting the results.

Examples

```
# Performing the Global for two populations
Global_result <- global2(NASAtemp$paris, NASAtemp$milan)

# Plotting the results of the Global
plot(
  Global_result,
  xrange = c(0, 12),
  title = "Global results for testing mean differences"
)

# Selecting the significant components at 5% level
which(Global_result$adjusted_pvalues < 0.05)
```

Globallm

Global testing procedure for testing functional-on-scalar linear models

Description

The function is used to fit and test functional linear models. It can be used to carry out regression, and analysis of variance. It implements the global testing procedure for testing the significance of the effects of scalar covariates on a functional population.

Usage

```
Globallm(
  formula,
  dx = NULL,
  B = 1000L,
  method = c("residuals", "responses"),
  stat = c("Integral", "Max")
)

global_lm(
  formula,
```

```

dx = NULL,
n_perm = 1000L,
method = c("residuals", "responses"),
stat = c("Integral", "Max")
)

```

Arguments

formula	An object of class <code>stats::formula</code> (or one that can be coerced to that class) specifying the model to be fitted in a symbolic fashion. The response (left-hand side) can be either a matrix of dimension $n \times J$ containing the pointwise evaluations of n functions on the same grid of J points, or an object of class <code>fda::fd</code> .
dx	A numeric value specifying the discretization step of the grid used to evaluate functional data when it is provided as objects of class <code>fda::fd</code> . Defaults to <code>NULL</code> , in which case a default value of <code>0.01</code> is used which corresponds to a grid of size <code>100L</code> . Unused if functional data is provided in the form of matrices.
B	An integer value specifying the number of permutations used to evaluate the p-values of the permutation tests. Defaults to <code>1000L</code> . Passed as <code>n_perm</code> in <code>iwt_aov()</code> , <code>twt_aov()</code> and <code>global_aov()</code> .
method	A string specifying the permutation scheme. "residuals" permutes residuals under the reduced model (Freedman-Lane scheme); "responses" permutes the responses (Manly scheme). Defaults to "residuals".
stat	A string specifying the test statistic used for the global test. "Integral" uses the integral of the F-statistic over the domain; "Max" uses the maximum. Defaults to "Integral".
n_perm	An integer value specifying the number of permutations for the permutation tests. Defaults to <code>1000L</code> .

Value

An object of class `flm` containing the following components:

- `call`: The matched call.
- `design_matrix`: The design matrix of the functional-on-scalar linear model.
- `unadjusted_pval_F`: A numeric vector of length J containing the unadjusted p-value function of the global F-test evaluated on the grid.
- `adjusted_pval_F`: A numeric vector of length J containing the adjusted p-value function of the global F-test evaluated on the grid.
- `unadjusted_pval_part`: A numeric matrix with one row per model term containing the unadjusted p-value functions of the per-term t-tests.
- `adjusted_pval_part`: A numeric matrix with one row per model term containing the adjusted p-value functions of the per-term t-tests.
- `data_eval`: A numeric matrix containing the functional response evaluated on the grid.
- `coeff_regr_eval`: A numeric matrix containing the functional regression coefficients evaluated on the grid.

- `fitted_eval`: A numeric matrix containing the fitted values of the functional regression evaluated on the grid.
- `residuals_eval`: A numeric matrix containing the residuals of the functional regression evaluated on the grid.
- `R2_eval`: A numeric vector containing the functional R-squared evaluated on the grid.

Optionally, the list may contain the following components:

- `pval_matrix_F`: A matrix of dimensions $p \times p$ of p-values of the interval-wise F-tests. Element (i, j) contains the p-value of the test on the interval $(j, j + 1, \dots, j + (p - i))$. Present only if correction is "IWT".
- `pval_matrix_part`: An array of dimensions $(L + 1) \times p \times p$ of p-values of the per-term interval-wise t-tests. Element (l, i, j) contains the p-value of the joint test on term l and interval $(j, j + 1, \dots, j + (p - i))$. Present only if correction is "IWT".
- `global_pval_F`: Global p-value of the overall F-test. Present only if correction is "Global".
- `global_pval_part`: A numeric vector of global p-values of the per-term t-tests. Present only if correction is "Global".

References

- Abramowicz, K., Pini, A., Schelin, L., Stamm, A., & Vantini, S. (2022). "Domain selection and familywise error rate for functional data: A unified framework. *Biometrics* 79(2), 1119-1132.
- D. Freedman and D. Lane (1983). A Nonstochastic Interpretation of Reported Significance Levels. *Journal of Business & Economic Statistics* 1(4), 292-298.
- B. F. J. Manly (2006). Randomization, *Bootstrap and Monte Carlo Methods in Biology*. Vol. 70. CRC Press.

See Also

`iwt_lm()`, `twt_lm()` and `global_lm()` for calling a specific correction directly. `plot.flm()` for plotting the results and `summary.flm()` for summarizing the results.

Examples

```
# Defining the covariates
temperature <- rbind(NASAtemp$milan, NASAtemp$paris)
groups <- c(rep(0, 22), rep(1, 22))

# Performing the Global test
Global_result <- Globallm(temperature ~ groups, B = 1000)
# Summary of the Global test results
summary(Global_result)

# Plot of the Global test results
plot(
  Global_result,
  main = "NASA data",
  plot_adjpv = TRUE,
  xlab = "Day",
```

```
xrange = c(1, 365)
)
```

ITP1bspline

One population Interval Wise Testing procedure

Description

The function implements the Interval Wise Testing procedure for testing the center of symmetry of a functional population. Functional data are tested locally and unadjusted and adjusted p-value functions are provided. The unadjusted p-value function controls the point-wise error rate. The adjusted p-value function controls the interval-wise error rate.

Usage

```
ITP1bspline(data, mu = 0, B = 1000, order = 2, nknots = dim(data)[2])
```

```
ITP1fourier(data, mu = 0, B = 1000, maxfrequency = floor(dim(data)[2]/2))
```

```
iwt1(
  data,
  mu = 0,
  n_perm = 1000L,
  dx = NULL,
  recycle = TRUE,
  verbose = FALSE,
  aggregation_strategy = c("integral", "max")
)
```

```
IWT1(data, mu = 0, B = 1000L, dx = NULL, recycle = TRUE)
```

Arguments

data	Either a numeric matrix or an object of class <code>fda::fd</code> specifying the sample data. If the data is provided within a matrix, it should be of shape $n \times J$ and it should contain in each row one of the n functions in the sample and in columns the evaluation of each function on a same uniform grid of size J .
mu	Either a numeric value or a numeric vector or an object of class <code>fda::fd</code> specifying the functional center of symmetry under the null hypothesis. If mu is a constant, then a constant function is used. If mu is a numeric vector, it must correspond to evaluations of the center function on the same grid that has been used to evaluate the data sample. Defaults to 0.
B	The number of iterations of the MC algorithm to evaluate the p-values of the permutation tests. Defaults to 1000L.
order	Order of the B-spline basis expansion. Defaults to 2L.
nknots	Number of knots of the B-spline basis expansion. Defaults to <code>dim(data)[2]</code> .

maxfrequency	The maximum frequency to be used in the Fourier basis expansion of data. Defaults to $\text{floor}(\text{dim}(\text{data})[2] / 2)$, leading to an interpolating expansion.
n_perm	An integer value specifying the number of permutations to use for the local testing procedure. Defaults to 1000L.
dx	A numeric value specifying the step of the uniform grid on which the data are evaluated. If NULL, the step is automatically inferred from the data. Defaults to NULL.
recycle	A boolean value specifying whether to recycle the test statistic values across permutations for the IWT procedure. Defaults to TRUE.
verbose	A boolean value specifying whether to print the progress of the computation. Defaults to FALSE.
aggregation_strategy	A string specifying the strategy to aggregate the point-wise test statistics for the correction procedure. Possible values are "integral" and "max". Defaults to "integral".

Value

An object of class `fos` containing the following components:

- `data`: A numeric matrix of shape $n \times J$ containing the evaluation of the n functions on a uniform grid of size J .
- `mu`: A numeric vector of shape J containing the evaluation of the functional center of symmetry under the null hypothesis on the same uniform grid used to evaluate the functional sample.
- `unadjusted_pvalues`: A numeric vector of size J containing the evaluation of the unadjusted p-value function on the same uniform grid used to evaluate the functional sample.
- `adjusted_pvalues`: A numeric vector of size J containing the evaluation of the adjusted p-value function on the same uniform grid used to evaluate the functional sample.
- `correction_method`: A string containing the correction method used to compute the adjusted p-value function.

Optionally, the list may contain the following component:

- `pvalue_matrix`: A numeric matrix of shape $p \times p$ containing the p-values of the interval-wise tests. Element i, j contains the p-value of the test performed on the interval indexed by $j, j + 1, \dots, j + (p - i)$. Only present if the `correction` argument is set to "IWT".

References

- Pini, Alessia, and Simone Vantini. 2016. "The interval testing procedure: a general framework for inference in functional data analysis." *Biometrics* 72 (3): 835–845.
- Pini, Alessia, and Simone Vantini. 2017. "Interval-Wise Testing for Functional Data." *Journal of Nonparametric Statistics* 29 (2): 407–24.

See Also

`functional_one_sample_test()` for the interface function and `plot.fos()` and `IWTimage()` for plotting the results.

Examples

```
# Performing the IWT for one population
IWT_result <- iwt1(NASAtemp$paris, mu = 4, n_perm = 10L)

# Selecting the significant components at 5% level
which(IWT_result$adjusted_pvalues < 0.05)
# Performing the IWT for one population
IWT_result <- IWT1(NASAtemp$paris, mu = 4, B = 10L)

# Plotting the results of the IWT
plot(IWT_result, xrange = c(0, 12), main = "Paris temperatures")

# Plotting the p-value heatmap
IWTimage(IWT_result, abscissa_range = c(0, 12))

# Selecting the significant components at 5% level
which(IWT_result$adjusted_pval < 0.05)
```

ITP2bspline

*Two population Interval Wise Testing procedure***Description**

The function implements the Interval Wise Testing procedure for testing mean differences between two functional populations. Functional data are tested locally and unadjusted and adjusted p-value functions are provided. The unadjusted p-value function controls the point-wise error rate. The adjusted p-value function controls the interval-wise error rate.

Usage

```
ITP2bspline(
  data1,
  data2,
  mu = 0,
  B = 1000,
  paired = FALSE,
  order = 2,
  nknots = dim(data1)[2]
)

ITP2fourier(
  data1,
  data2,
  mu = 0,
  B = 1000,
  paired = FALSE,
  maxfrequency = floor(dim(data1)[2]/2)
)
```

```

ITP2pafourier(
  data1,
  data2,
  mu = 0,
  B = 1000,
  paired = FALSE,
  maxfrequency = floor(dim(data1)[2]/2)
)

iwt2(
  data1,
  data2,
  mu = 0,
  dx = NULL,
  n_perm = 1000L,
  paired = FALSE,
  alternative = c("two.sided", "less", "greater"),
  standardize = FALSE,
  verbose = FALSE,
  aggregation_strategy = c("integral", "max"),
  recycle = TRUE
)

IWT2(
  data1,
  data2,
  mu = 0,
  dx = NULL,
  B = 1000L,
  paired = FALSE,
  alternative = c("two.sided", "less", "greater"),
  statistic = c("Integral", "Max", "Integral_std", "Max_std"),
  verbose = FALSE,
  recycle = TRUE
)

```

Arguments

- | | |
|-------|--|
| data1 | Either a numeric matrix or an object of class <code>fda::fd</code> specifying the data in the first sample. If the data is provided within a matrix, it should be of shape $n_1 \times J$ and it should contain in each row one of the n_1 functions in the sample and in columns the evaluation of each function on a same uniform grid of size J . |
| data2 | Either a numeric matrix or an object of class <code>fda::fd</code> specifying the data in the second sample. If the data is provided within a matrix, it should be of shape $n_2 \times J$ and it should contain in each row one of the n_2 functions in the sample and in columns the evaluation of each function on a same uniform grid of size J . |

mu	Either a numeric value or a numeric vector or an object of class <code>fda::fd</code> specifying the functional mean difference under the null hypothesis. If mu is a constant, then a constant function is used. If mu is a numeric vector, it must correspond to evaluation of the mean difference function on the same grid that has been used to evaluate the data samples. Defaults to 0.
B	An integer value specifying the number of permutations to use for the local testing procedure. Defaults to 1000L.
paired	A boolean value specifying whether a paired test should be performed. Defaults to FALSE.
order	Order of the B-spline basis expansion. Defaults to 2L.
nknots	Number of knots of the B-spline basis expansion. Defaults to <code>dim(data1)[2]</code> .
maxfrequency	The maximum frequency to be used in the Fourier basis expansion of data. Defaults to <code>floor(dim(data1)[2] / 2)</code> , leading to an interpolating expansion.
dx	A numeric value specifying the step of the uniform grid on which the data are evaluated. If NULL, the step is automatically inferred from the data. Defaults to NULL.
n_perm	An integer value specifying the number of permutations to use for the local testing procedure. Defaults to 1000L.
alternative	A string specifying the type of alternative hypothesis. Choices are "two.sided", "less" or "greater". Defaults to "two.sided".
standardize	A boolean value specifying whether to standardize the test statistic. Defaults to FALSE.
verbose	A boolean value specifying whether to print the progress of the computation. Defaults to FALSE.
aggregation_strategy	A string specifying the strategy to aggregate the point-wise test statistics for the correction procedure. Possible values are "integral" and "max". Defaults to "integral".
recycle	A boolean value specifying whether to recycle the test statistic values across permutations for the IWT procedure. Defaults to TRUE.
statistic	A string specifying the test statistic to use. Possible values are: • "Integral": Integral of the squared sample mean difference. • "Max": Maximum of the squared sample mean difference. • "Integral_std": Integral of the squared t-test statistic. • "Max_std": Maximum of the squared t-test statistic. Defaults to "Integral".

Value

An object of class `fts` containing the following components:

- `data`: A numeric matrix of shape $n \times J$ containing the evaluation of the $n = n_1 + n_2$ functions on a **common** uniform grid of size p .

- `group_labels`: An integer vector of size $n = n_1 + n_2$ containing the group membership of each function.
- `mu`: A numeric vector of shape J containing the evaluation of the functional mean difference under the null hypothesis on the same uniform grid used to evaluate the functional samples.
- `unadjusted_pvalues`: A numeric vector of size J containing the evaluation of the unadjusted p-value function on the **same** uniform grid used to evaluate the functional samples.
- `adjusted_pvalues`: A numeric vector of size J containing the evaluation of the adjusted p-value function on the **same** uniform grid used to evaluate the functional samples.
- `correction_method`: A string containing the correction method used to compute the adjusted p-value function.

Optionally, the list may contain the following components:

- `global_pvalue`: A numeric value containing the global p-value. Only present if the correction argument is set to "Global".
- `pvalue_matrix`: A numeric matrix of shape $p \times p$ containing the p-values of the interval-wise tests. Element i, j contains the p-value of the test performed on the interval indexed by $j, j + 1, \dots, j + (p - i)$. Only present if the correction argument is set to "IWT".

References

- Pini, Alessia, and Simone Vantini. 2016. "The interval testing procedure: a general framework for inference in functional data analysis." *Biometrics* 72 (3): 835–845.
- Pini, Alessia, and Simone Vantini. 2017. "Interval-Wise Testing for Functional Data." *Journal of Nonparametric Statistics* 29 (2): 407–24.
- Pini, Alessia, Simone Vantini, Bianca Maria Colosimo, and Marco Grasso. 2018. "Domain-Selective Functional Analysis of Variance for Supervised Statistical Profile Monitoring of Signal Data." *Journal of the Royal Statistical Society Series C: Applied Statistics* 67 (1): 55–81.
- Abramowicz, Konrad, Charlotte K Häger, Alessia Pini, Lina Schelin, Sara Sjöstedt de Luna, and Simone Vantini. 2018. "Nonparametric Inference for Functional-on-Scalar Linear Models Applied to Knee Kinematic Hop Data After Injury of the Anterior Cruciate Ligament." *Scandinavian Journal of Statistics* 45 (4): 1036–61.

See Also

`global2()`, `tw2()`, `pct2()`, `fdr2()` for calling directly one of the other tests, `functional_two_sample_test()` for calling the interface test and `plot.fts()` for plotting the results.

Examples

```
# Performing the IWT for two populations
IWT_result <- iwt2(NASAtemp$paris, NASAtemp$milan, n_perm = 10L)

# Plotting the results of the IWT
plot(
  IWT_result,
  xrange = c(0, 12),
  title = "IWT results for testing mean differences"
```

```

)

# Plotting the p-value heatmap
IWTimage(IWT_result, abscissa_range = c(0, 12))

# Selecting the significant components at 5% level
which(IWT_result$adjusted_pvalues < 0.05)

```

ITPaovbspline	<i>Interval Testing Procedure for testing unctional analysis of variance with B-spline basis</i>
---------------	--

Description

The function implements the Interval Testing Procedure for testing for significant differences between several functional population evaluated on a uniform grid, in a functional analysis of variance setting. Data are represented by means of the B-spline basis and the significance of each basis coefficient is tested with an interval-wise control of the Family Wise Error Rate. The default parameters of the basis expansion lead to the piece-wise interpolating function.

Usage

```

ITPaovbspline(
  formula,
  order = 2,
  nknots = dim(stats::model.response(stats::model.frame(formula)))[2],
  B = 1000,
  method = "residuals"
)

```

Arguments

formula	An object of class "formula" (or one that can be coerced to that class): a symbolic description of the model to be fitted.
order	Order of the B-spline basis expansion. The default is order=2.
nknots	Number of knots of the B-spline basis expansion. The default is nknots=dim(data1)[2].
B	The number of iterations of the MC algorithm to evaluate the p-values of the permutation tests. The default is B=1000.
method	Permutation method used to calculate the p-value of permutation tests. Choose "residuals" for the permutations of residuals under the reduced model, according to the Freedman and Lane scheme, and "responses" for the permutation of the responses, according to the Manly scheme.

Value

ITPaovbspline returns an object of `class` "ITPaov". The function summary is used to obtain and print a summary of the results. An object of class "ITPaov" is a list containing at least the following components:

<code>call</code>	The matched call.
<code>design_matrix</code>	The design matrix of the functional-on-scalar linear model.
<code>basis</code>	String vector indicating the basis used for the first phase of the algorithm. In this case equal to "B-spline".
<code>coeff</code>	Matrix of dimensions $c(n, p)$ of the p coefficients of the B-spline basis expansion. Rows are associated to units and columns to the basis index.
<code>coeff_regr</code>	Matrix of dimensions $c(L+1, p)$ of the p coefficients of the B-spline basis expansion of the intercept (first row) and the L effects of the covariates specified in formula. Columns are associated to the basis index.
<code>pval_F</code>	Unadjusted p-values of the functional F-test for each basis coefficient.
<code>pval_matrix_F</code>	Matrix of dimensions $c(p, p)$ of the p-values of the multivariate F-tests. The element (i, j) of matrix <code>pval_matrix</code> contains the p-value of the joint NPC test of the components $(j, j+1, \dots, j+(p-i))$.
<code>adjusted_pval_F</code>	Adjusted p-values of the functional F-test for each basis coefficient.
<code>pval_factors</code>	Unadjusted p-values of the functional F-tests on each factor of the analysis of variance, separately (rows) and each basis coefficient (columns).
<code>pval_matrix_factors</code>	Array of dimensions $c(L+1, p, p)$ of the p-values of the multivariate F-tests on factors. The element (l, i, j) of array <code>pval_matrix</code> contains the p-value of the joint NPC test on factor l of the components $(j, j+1, \dots, j+(p-i))$.
<code>adjusted_pval_factors</code>	adjusted p-values of the functional F-tests on each factor of the analysis of variance (rows) and each basis coefficient (columns).
<code>data_eval</code>	Evaluation on a fine uniform grid of the functional data obtained through the basis expansion.
<code>coeff_regr_eval</code>	Evaluation on a fine uniform grid of the functional regression coefficients.
<code>fitted_eval</code>	Evaluation on a fine uniform grid of the fitted values of the functional regression.
<code>residuals_eval</code>	Evaluation on a fine uniform grid of the residuals of the functional regression.
<code>R2_eval</code>	Evaluation on a fine uniform grid of the functional R-squared of the regression.
<code>heatmap_matrix_F</code>	Heatmap matrix of p-values of functional F-test (used only for plots).
<code>heatmap_matrix_factors</code>	Heatmap matrix of p-values of functional F-tests on each factor of the analysis of variance (used only for plots).

References

- A. Pini and S. Vantini (2017). The Interval Testing Procedure: Inference for Functional Data Controlling the Family Wise Error Rate on Intervals. *Biometrics* 73(3): 835–845.
- Pini, A., Vantini, S., Colosimo, B. M., & Grasso, M. (2018). Domain-selective functional analysis of variance for supervised statistical profile monitoring of signal data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 67(1), 55-81.
- Abramowicz, K., Hager, C. K., Pini, A., Schelin, L., Sjostedt de Luna, S., & Vantini, S. (2018). Nonparametric inference for functional-on-scalar linear models applied to knee kinematic hop data after injury of the anterior cruciate ligament. *Scandinavian Journal of Statistics* 45(4), 1036-1061.
- D. Freedman and D. Lane (1983). A Nonstochastic Interpretation of Reported Significance Levels. *Journal of Business & Economic Statistics* 1.4, 292-298.
- B. F. J. Manly (2006). *Randomization, Bootstrap and Monte Carlo Methods in Biology*. Vol. 70. CRC Press.

Examples

```
temperature <- rbind(NASAtemp$milan,NASAtemp$paris)
groups <- c(rep(0,22),rep(1,22))

# Performing the ITP
ITP_result <- ITPaovbspline(temperature ~ groups,B=5,nknots=20,order=3)

# Summary of the ITP results
summary(ITP_result)

# Plot of the ITP results
plot(
  ITP_result,
  main = "NASA data",
  plot_adjval = TRUE,
  xlab = "Day",
  xrange = c(1, 365)
)
```

ITPimage

Heatmap plot of the Interval Testing Procedure results

Description

Plotting function creating a graphical output of the ITP: the p-value heat-map, the plot of the corrected p-values, and the plot of the functional data.

Usage

```
ITPimage(ITP_result, alpha = 0.05, abscissa_range = c(0, 1), nlevel = 20)
```

Arguments

ITP_result	Results of the ITP, as created by ITP1bspline , ITP1fourier , ITP2bspline , ITP2fourier , and ITP2pafourier .
alpha	Threshold for the interval-wise error rate used for the hypothesis test. The default is $\alpha=0.05$.
abscissa_range	Range of the plot abscissa. The default is $c(0, 1)$.
nlevel	Number of desired color levels for the p-value heatmap. The default is $nlevel=20$.

References

A. Pini and S. Vantini (2017). The Interval Testing Procedure: Inference for Functional Data Controlling the Family Wise Error Rate on Intervals. *Biometrics* 73(3): 835–845.

Examples

```
# Performing the ITP for two populations with the B-spline basis
ITP_result <- ITP2bspline(
  NASAtemp$milan, NASAtemp$paris,
  nknots = 20,
  B = 10L
)

# Plotting the results of the ITP
ITPimage(ITP_result, abscissa_range=c(0,12))

# Selecting the significant components for the radius at 5% level
which(ITP_result$corrected_pval < 0.05)
```

ITPlmbspline	<i>Interval Testing Procedure for testing Functional-on-Scalar Linear Models with B-spline basis</i>
--------------	--

Description

The function is used to fit and test functional linear models. It can be used to carry out regression, and analysis of variance. It implements the Interval Testing Procedure for testing the significance of the effects of scalar covariates on a functional population evaluated on a uniform grid. Data are represented by means of the B-spline basis and the significance of each basis coefficient is tested with an interval-wise control of the Family Wise Error Rate. The default parameters of the basis expansion lead to the piece-wise interpolating function.

Usage

```
ITPlmbspline(
  formula,
  order = 2,
  nknots = dim(stats::model.response(stats::model.frame(formula)))[2],
```

```

    B = 10000,
    method = "residuals"
)

```

Arguments

formula	An object of class "formula" (or one that can be coerced to that class): a symbolic description of the model to be fitted.
order	Order of the B-spline basis expansion. The default is order=2.
nknots	Number of knots of the B-spline basis expansion. The default is nknots=dim(data1)[2].
B	The number of iterations of the MC algorithm to evaluate the p-values of the permutation tests. The default is B=1000.
method	Permutation method used to calculate the p-value of permutation tests. Choose "residuals" for the permutations of residuals under the reduced model, according to the Freedman and Lane scheme, and "responses" for the permutation of the responses, according to the Manly scheme.

Value

ITPlmbspline returns an object of class "ITPlm". The function summary is used to obtain and print a summary of the results. An object of class "ITPlm" is a list containing at least the following components:

call	The matched call.
design_matrix	The design matrix of the functional-on-scalar linear model.
basis	String vector indicating the basis used for the first phase of the algorithm. In this case equal to "B-spline".
coeff	Matrix of dimensions $c(n, p)$ of the p coefficients of the B-spline basis expansion. Rows are associated to units and columns to the basis index.
coeff_regr	Matrix of dimensions $c(L+1, p)$ of the p coefficients of the B-spline basis expansion of the intercept (first row) and the L effects of the covariates specified in formula. Columns are associated to the basis index.
pval_F	Unadjusted p-values of the functional F-test for each basis coefficient.
pval_matrix_F	Matrix of dimensions $c(p, p)$ of the p-values of the multivariate F-tests. The element (i, j) of matrix pval_matrix contains the p-value of the joint NPC test of the components $(j, j+1, \dots, j+(p-i))$.
adjusted_pval_F	Adjusted p-values of the functional F-test for each basis coefficient.
pval_t	Unadjusted p-values of the functional t-tests for each partial regression coefficient including the intercept (rows) and each basis coefficient (columns).
pval_matrix_t	Array of dimensions $c(L+1, p, p)$ of the p-values of the multivariate t-tests. The element (l, i, j) of array pval_matrix contains the p-value of the joint NPC test on covariate l of the components $(j, j+1, \dots, j+(p-i))$.
adjusted_pval_t	adjusted p-values of the functional t-tests for each partial regression coefficient including the intercept (rows) and each basis coefficient (columns).

<code>data_eval</code>	Evaluation on a fine uniform grid of the functional data obtained through the basis expansion.
<code>coeff_regr_eval</code>	Evaluation on a fine uniform grid of the functional regression coefficients.
<code>fitted_eval</code>	Evaluation on a fine uniform grid of the fitted values of the functional regression.
<code>residuals_eval</code>	Evaluation on a fine uniform grid of the residuals of the functional regression.
<code>R2_eval</code>	Evaluation on a fine uniform grid of the functional R-squared of the regression.
<code>heatmap_matrix_F</code>	Heatmap matrix of p-values of functional F-test (used only for plots).
<code>heatmap_matrix_t</code>	Heatmap matrix of p-values of functional t-tests (used only for plots).

References

- A. Pini and S. Vantini (2017). The Interval Testing Procedure: Inference for Functional Data Controlling the Family Wise Error Rate on Intervals. *Biometrics* 73(3): 835–845.
- Pini, A., Vantini, S., Colosimo, B. M., & Grasso, M. (2018). Domain-selective functional analysis of variance for supervised statistical profile monitoring of signal data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 67(1), 55-81.
- Abramowicz, K., Hager, C. K., Pini, A., Schelin, L., Sjostedt de Luna, S., & Vantini, S. (2018). Nonparametric inference for functional-on-scalar linear models applied to knee kinematic hop data after injury of the anterior cruciate ligament. *Scandinavian Journal of Statistics* 45(4), 1036-1061.
- D. Freedman and D. Lane (1983). A Nonstochastic Interpretation of Reported Significance Levels. *Journal of Business & Economic Statistics* 1(4), 292-298.
- B. F. J. Manly (2006). *Randomization, Bootstrap and Monte Carlo Methods in Biology*. Vol. 70. CRC Press.

Examples

```
# Defining the covariates
temperature <- rbind(NASAtemp$milan,NASAtemp$paris)
groups <- c(rep(0,22),rep(1,22))

# Performing the ITP
#ITP_result <- ITPlmbspline(temperature ~ groups,B=100,nknots=20)
# Summary of the ITP results
#summary(ITP_result)

# Plot of the ITP results
#plot(ITP_result,main='NASA data', plot_adjpv = TRUE,xlab='Day',xrange=c(1,365))

#plot(ITP_result,main='NASA data', plot_adjpv = TRUE,xlab='Day',xrange=c(1,365))
```

IWTaov	<i>Interval Wise Testing procedure for testing functional analysis of variance</i>
--------	--

Description

The function implements the Interval Wise Testing procedure for testing mean differences between several functional populations in a one-way or multi-way functional analysis of variance framework. Functional data are tested locally and unadjusted and adjusted p-value functions are provided. The unadjusted p-value function controls the point-wise error rate. The adjusted p-value function controls the interval-wise error rate.

Usage

```
IWTaov(
  formula,
  dx = NULL,
  B = 1000L,
  method = c("residuals", "responses"),
  recycle = TRUE
)

iwt_aov(
  formula,
  dx = NULL,
  n_perm = 1000L,
  method = c("residuals", "responses"),
  recycle = TRUE
)
```

Arguments

formula	An object of class <code>stats::formula</code> (or one that can be coerced to that class) specifying the model to be fitted in a symbolic fashion. The response (left-hand side) can be either a matrix of dimension $n \times J$ containing the pointwise evaluations of n functions on the same grid of J points, or an object of class <code>fda::fd</code> .
dx	A numeric value specifying the discretization step of the grid used to evaluate functional data when it is provided as objects of class <code>fda::fd</code> . Defaults to <code>NULL</code> , in which case a default value of <code>0.01</code> is used which corresponds to a grid of size <code>100L</code> . Unused if functional data is provided in the form of matrices.
B	An integer value specifying the number of permutations used to evaluate the p-values of the permutation tests. Defaults to <code>1000L</code> . Passed as <code>n_perm</code> in <code>iwt_aov()</code> , <code>twt_aov()</code> and <code>global_aov()</code> .
method	A string specifying the permutation scheme. "residuals" permutes residuals under the reduced model (Freedman-Lane scheme); "responses" permutes the responses (Manly scheme). Defaults to "residuals".

recycle	A boolean value specifying whether the recycled version of the interval-wise testing procedure should be used. See Pini and Vantini (2017) for details. Defaults to TRUE.
n_perm	An integer value specifying the number of permutations for the permutation tests. Defaults to 1000L.

Value

An object of class `faov` containing the following components:

- `call`: The matched call.
- `design_matrix`: The design matrix of the functional ANOVA model.
- `unadjusted_pval_F`: A numeric vector of length J containing the unadjusted p-value function of the global F-test evaluated on the grid.
- `adjusted_pval_F`: A numeric vector of length J containing the adjusted p-value function of the global F-test evaluated on the grid.
- `unadjusted_pval_factors`: A numeric matrix with one row per factor containing the unadjusted p-value functions of the per-factor F-tests.
- `adjusted_pval_factors`: A numeric matrix with one row per factor containing the adjusted p-value functions of the per-factor F-tests.
- `data_eval`: A numeric matrix containing the functional data evaluated on the grid.
- `coeff_regr_eval`: A numeric matrix containing the functional regression coefficients evaluated on the grid.
- `fitted_eval`: A numeric matrix containing the fitted values of the functional regression evaluated on the grid.
- `residuals_eval`: A numeric matrix containing the residuals of the functional regression evaluated on the grid.
- `R2_eval`: A numeric vector containing the functional R-squared evaluated on the grid.

Optionally, the list may contain the following components:

- `pval_matrix_F`: A matrix of dimensions $p \times p$ of p-values of the interval-wise F-tests. Element (i, j) contains the p-value of the test on the interval $(j, j + 1, \dots, j + (p - i))$. Present only if correction is "IWT".
- `pval_matrix_factors`: An array of dimensions $L \times p \times p$ of p-values of the per-factor interval-wise F-tests. Element (l, i, j) contains the p-value of the joint test on factor l and interval $(j, j + 1, \dots, j + (p - i))$. Present only if correction is "IWT".
- `global_pval_F`: Global p-value of the overall F-test. Present only if correction is "Global".
- `global_pval_factors`: A numeric vector of global p-values of the per-factor F-tests. Present only if correction is "Global".

References

- Pini, A., & Vantini, S. (2017). Interval-wise testing for functional data. *Journal of Nonparametric Statistics*, 29(2), 407-424.
- Pini, A., Vantini, S., Colosimo, B. M., & Grasso, M. (2018). Domain-selective functional analysis of variance for supervised statistical profile monitoring of signal data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 67(1), 55-81.
- Abramowicz, K., Hager, C. K., Pini, A., Schelin, L., Sjostedt de Luna, S., & Vantini, S. (2018). Nonparametric inference for functional-on-scalar linear models applied to knee kinematic hop data after injury of the anterior cruciate ligament. *Scandinavian Journal of Statistics* 45(4), 1036-1061.
- D. Freedman and D. Lane (1983). A Nonstochastic Interpretation of Reported Significance Levels. *Journal of Business & Economic Statistics* 1.4, 292-298.
- B. F. J. Manly (2006). *Randomization, Bootstrap and Monte Carlo Methods in Biology*. Vol. 70. CRC Press.

See Also

[iwt_aov\(\)](#), [twtaov\(\)](#) and [global_aov\(\)](#) for calling a specific correction directly. [plot_faov\(\)](#) for plotting the results and [summary_faov\(\)](#) for summarizing the results.

Examples

```
temperature <- rbind(NASAtemp$milan, NASAtemp$paris)
groups <- c(rep(0, 22), rep(1, 22))

# Performing the IWT
IWT_result <- IWTaov(temperature ~ groups, B = 10L)

# Summary of the IWT results
summary(IWT_result)

# Plot of the IWT results
graphics::layout(1)
plot(IWT_result)

# All graphics on the same device
graphics::layout(matrix(1:4, nrow = 2, byrow = FALSE))
plot(
  IWT_result,
  main = "NASA data",
  plot.adj.pval = TRUE,
  xlab = "Day",
  xrange = c(1, 365)
)
```

IWTimage

*Heatmap plot of the Interval Wise Testing Procedure results***Description**

Plotting function creating a ggplot2 graphical output of the IWT: the p-value heat-map, the adjusted p-value function, and the functional data, assembled via **patchwork**.

Usage

```
IWTimage(
  IWT_result,
  alpha = 0.05,
  abscissa_range = c(0, 1),
  nlevel = 20L,
  plot_unadjusted = FALSE
)
```

Arguments

IWT_result	Results of the IWT, as created by <code>functional_one_sample_test()</code> , <code>iwt1()</code> , <code>functional_two_sample_test()</code> , <code>iwt2()</code> , or the legacy functions <code>IWT1()</code> and <code>IWtaov()</code> . When using <code>functional_two_sample_test()</code> or <code>iwt2()</code> , correction must be "IWT".
alpha	Threshold for the interval-wise error rate used for the hypothesis test. Regions where the adjusted p-value is below alpha are highlighted. The default is alpha = 0.05.
abscissa_range	Range of the plot abscissa. The default is c(0, 1).
nlevel	Number of desired color levels for the p-value heatmap. The default is nlevel = 20.
plot_unadjusted	Flag indicating if the unadjusted p-value function has to be overlaid (dashed line) on the adjusted p-value panel. The default is FALSE.

Value

An object of class patchwork containing the assembled ggplot2 panels, returned invisibly. The plot is also printed as a side effect.

References

Pini, A., & Vantini, S. (2018). Interval-wise testing for functional data. *Journal of Nonparametric Statistics*, 29(2), 407-424.

Pini, A., Vantini, S., Colosimo, B. M., & Grasso, M. (2018). Domain-selective functional analysis of variance for supervised statistical profile monitoring of signal data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 67(1), 55-81.

Abramowicz, K., Hager, C. K., Pini, A., Schelin, L., Sjostedt de Luna, S., & Vantini, S. (2018). Nonparametric inference for functional-on-scalar linear models applied to knee kinematic hop data after injury of the anterior cruciate ligament. *Scandinavian Journal of Statistics* 45(4), 1036-1061.

See Also

See [plot.fos\(\)](#), [plot.fts\(\)](#), [plot.flm\(\)](#) and [plot.faov\(\)](#) for the plot method applied to the IWT results of one- and two-population tests, linear models, and ANOVA, respectively.

Examples

```
# Performing the IWT for one population
IWT_result <- functional_one_sample_test(
  NASAtemp$paris, mu = 4, n_perm = 10L
)

# Plotting the results of the IWT
IWTimage(IWT_result, abscissa_range = c(0, 12))

# Selecting the significant components at 5% level
which(IWT_result$adjusted_pvalues < 0.05)
```

IWTlm	<i>Interval-wise testing procedure for testing functional-on-scalar linear models</i>
-------	---

Description

The function is used to fit and test functional linear models. It can be used to carry out regression, and analysis of variance. It implements the interval-wise testing procedure (IWT) for testing the significance of the effects of scalar covariates on a functional population.

Usage

```
IWTlm(
  formula,
  dx = NULL,
  B = 1000L,
  method = c("residuals", "responses"),
  recycle = TRUE
)

iwt_lm(
  formula,
  dx = NULL,
  n_perm = 1000L,
  method = c("residuals", "responses"),
  recycle = TRUE
)
```

Arguments

formula	An object of class <code>stats::formula</code> (or one that can be coerced to that class) specifying the model to be fitted in a symbolic fashion. The response (left-hand side) can be either a matrix of dimension $n \times J$ containing the pointwise evaluations of n functions on the same grid of J points, or an object of class <code>fda::fd</code> .
dx	A numeric value specifying the discretization step of the grid used to evaluate functional data when it is provided as objects of class <code>fda::fd</code> . Defaults to NULL, in which case a default value of 0.01 is used which corresponds to a grid of size 100L. Unused if functional data is provided in the form of matrices.
B	An integer value specifying the number of permutations used to evaluate the p-values of the permutation tests. Defaults to 1000L. Passed as <code>n_perm</code> in <code>iwt_aov()</code> , <code>twt_aov()</code> and <code>global_aov()</code> .
method	A string specifying the permutation scheme. "residuals" permutes residuals under the reduced model (Freedman-Lane scheme); "responses" permutes the responses (Manly scheme). Defaults to "residuals".
recycle	A boolean value specifying whether the recycled version of the interval-wise testing procedure should be used. See Pini and Vantini (2017) for details. Defaults to TRUE.
n_perm	An integer value specifying the number of permutations for the permutation tests. Defaults to 1000L.

Value

An object of class `flm` containing the following components:

- `call`: The matched call.
- `design_matrix`: The design matrix of the functional-on-scalar linear model.
- `unadjusted_pval_F`: A numeric vector of length J containing the unadjusted p-value function of the global F-test evaluated on the grid.
- `adjusted_pval_F`: A numeric vector of length J containing the adjusted p-value function of the global F-test evaluated on the grid.
- `unadjusted_pval_part`: A numeric matrix with one row per model term containing the unadjusted p-value functions of the per-term t-tests.
- `adjusted_pval_part`: A numeric matrix with one row per model term containing the adjusted p-value functions of the per-term t-tests.
- `data_eval`: A numeric matrix containing the functional response evaluated on the grid.
- `coeff_regr_eval`: A numeric matrix containing the functional regression coefficients evaluated on the grid.
- `fitted_eval`: A numeric matrix containing the fitted values of the functional regression evaluated on the grid.
- `residuals_eval`: A numeric matrix containing the residuals of the functional regression evaluated on the grid.
- `R2_eval`: A numeric vector containing the functional R-squared evaluated on the grid.

Optionally, the list may contain the following components:

- `pval_matrix_F`: A matrix of dimensions $p \times p$ of p-values of the interval-wise F-tests. Element (i, j) contains the p-value of the test on the interval $(j, j + 1, \dots, j + (p - i))$. Present only if correction is "IWT".
- `pval_matrix_part`: An array of dimensions $(L + 1) \times p \times p$ of p-values of the per-term interval-wise t-tests. Element (l, i, j) contains the p-value of the joint test on term l and interval $(j, j + 1, \dots, j + (p - i))$. Present only if correction is "IWT".
- `global_pval_F`: Global p-value of the overall F-test. Present only if correction is "Global".
- `global_pval_part`: A numeric vector of global p-values of the per-term t-tests. Present only if correction is "Global".

References

- A. Pini and S. Vantini (2017). The Interval Testing Procedure: Inference for Functional Data Controlling the Family Wise Error Rate on Intervals. *Biometrics* 73(3): 835–845.
- Pini, A., Vantini, S., Colosimo, B. M., & Grasso, M. (2018). Domain-selective functional analysis of variance for supervised statistical profile monitoring of signal data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 67(1), 55-81.
- Abramowicz, K., Hager, C. K., Pini, A., Schelin, L., Sjostedt de Luna, S., & Vantini, S. (2018). Nonparametric inference for functional-on-scalar linear models applied to knee kinematic hop data after injury of the anterior cruciate ligament. *Scandinavian Journal of Statistics* 45(4), 1036-1061.
- D. Freedman and D. Lane (1983). A Nonstochastic Interpretation of Reported Significance Levels. *Journal of Business & Economic Statistics* 1(4), 292-298.
- B. F. J. Manly (2006). *Randomization, Bootstrap and Monte Carlo Methods in Biology*. Vol. 70. CRC Press.

See Also

`iwt_lm()`, `twt_lm()` and `global_lm()` for calling a specific correction directly. `plot.flm()` for plotting the results and `summary.flm()` for summarizing the results.

Examples

```
# Defining the covariates
temperature <- rbind(NASAtemp$milan[, 1:100], NASAtemp$paris[, 1:100])
groups <- c(rep(0, 22), rep(1, 22))

# Performing the IWT
IWT_result <- IWTlm(temperature ~ groups, B = 2L)
# Summary of the IWT results
summary(IWT_result)

# Plot of the IWT results
plot(
  IWT_result,
  main = "NASA data",
  plot_adjpv = TRUE,
  xlab = "Day",
```

```
xrange = c(1, 365)
)
```

NASAtemp

NASA daily temperatures data set

Description

It contains the daily mean temperatures registered from July 1983 to June 2005 and stored in the NASA database Earth Surface Meteorology for Solar Energy of two different geographical locations: the region (45-46 North, 9-10 East), including the city of Milan (Italy), and the region (48-49 North, 2-3 East), including the city of Paris (France).

Usage

NASAtemp

Format

List of 2 elements:

- milan: Matrix of dimensions c(22, 365) containing the daily mean temperatures of the region (45-46 North, 9-10 East), including the city of Milan (Italy) registered from July 1983 to June 2005 (22 years).
- paris: Matrix of dimensions c(22, 365) containing the daily mean temperatures of the region (48-49 North, 2-3 East), including the city of Paris (France) registered from July 1983 to June 2005 (22 years).

Source

These data were obtained from the NASA Langley Research Center Atmospheric Science Data Center Surface meteorological and Solar Energy (SSE) web portal supported by the NASA LaRC POWER Project. Data are freely available at: [NASA Surface Meteorology and Solar Energy, A Renewable Energy Resource web site \(release 6.0\)](#).

pct2

Two population Partition Closed Testing procedure

Description

The function implements the Partition Closed Testing procedure for testing mean differences between two functional populations. Functional data are tested locally and unadjusted and adjusted p-value functions are provided. The unadjusted p-value function controls the point-wise error rate. The adjusted p-value function controls the family-wise error rate asymptotically.

Usage

```

pct2(
  data1,
  data2,
  partition,
  mu = 0,
  dx = NULL,
  n_perm = 1000L,
  paired = FALSE,
  alternative = c("two.sided", "less", "greater"),
  standardize = FALSE,
  verbose = FALSE,
  aggregation_strategy = c("integral", "max")
)

PCT2(
  data1,
  data2,
  partition,
  mu = 0,
  dx = NULL,
  B = 1000L,
  paired = FALSE,
  alternative = c("two.sided", "less", "greater"),
  statistic = c("Integral", "Max", "Integral_std", "Max_std"),
  verbose = FALSE
)

```

Arguments

data1	Either a numeric matrix or an object of class <code>fda::fd</code> specifying the data in the first sample. If the data is provided within a matrix, it should be of shape $n_1 \times J$ and it should contain in each row one of the n_1 functions in the sample and in columns the evaluation of each function on a same uniform grid of size J .
data2	Either a numeric matrix or an object of class <code>fda::fd</code> specifying the data in the second sample. If the data is provided within a matrix, it should be of shape $n_2 \times J$ and it should contain in each row one of the n_2 functions in the sample and in columns the evaluation of each function on a same uniform grid of size J .
partition	An integer vector of length J specifying the membership of each point of the domain to an element of the partition. Only used and must be set if the correction argument is set to "PCT".
mu	Either a numeric value or a numeric vector or an object of class <code>fda::fd</code> specifying the functional mean difference under the null hypothesis. If <code>mu</code> is a constant, then a constant function is used. If <code>mu</code> is a numeric vector, it must correspond to evaluation of the mean difference function on the same grid that has been used to evaluate the data samples. Defaults to $\emptyset$.

<code>dx</code>	A numeric value specifying the step of the uniform grid on which the data are evaluated. If <code>NULL</code> , the step is automatically inferred from the data. Defaults to <code>NULL</code> .
<code>n_perm</code>	An integer value specifying the number of permutations to use for the local testing procedure. Defaults to <code>1000L</code> .
<code>paired</code>	A boolean value specifying whether a paired test should be performed. Defaults to <code>FALSE</code> .
<code>alternative</code>	A string specifying the type of alternative hypothesis. Choices are <code>"two.sided"</code> , <code>"less"</code> or <code>"greater"</code> . Defaults to <code>"two.sided"</code> .
<code>standardize</code>	A boolean value specifying whether to standardize the test statistic. Defaults to <code>FALSE</code> .
<code>verbose</code>	A boolean value specifying whether to print the progress of the computation. Defaults to <code>FALSE</code> .
<code>aggregation_strategy</code>	A string specifying the strategy to aggregate the point-wise test statistics for the correction procedure. Possible values are <code>"integral"</code> and <code>"max"</code> . Defaults to <code>"integral"</code> .
<code>B</code>	An integer value specifying the number of permutations to use for the local testing procedure. Defaults to <code>1000L</code> .
<code>statistic</code>	A string specifying the test statistic to use. Possible values are: • <code>"Integral"</code>: Integral of the squared sample mean difference. • <code>"Max"</code>: Maximum of the squared sample mean difference. • <code>"Integral_std"</code>: Integral of the squared t-test statistic. • <code>"Max_std"</code>: Maximum of the squared t-test statistic. Defaults to <code>"Integral"</code> .

Value

An object of class `fts` containing the following components:

- `data`: A numeric matrix of shape $n \times J$ containing the evaluation of the $n = n_1 + n_2$ functions on a **common** uniform grid of size p .
- `group_labels`: An integer vector of size $n = n_1 + n_2$ containing the group membership of each function.
- `mu`: A numeric vector of shape J containing the evaluation of the functional mean difference under the null hypothesis on the same uniform grid used to evaluate the functional samples.
- `unadjusted_pvalues`: A numeric vector of size J containing the evaluation of the unadjusted p-value function on the **same** uniform grid used to evaluate the functional samples.
- `adjusted_pvalues`: A numeric vector of size J containing the evaluation of the adjusted p-value function on the **same** uniform grid used to evaluate the functional samples.
- `correction_method`: A string containing the correction method used to compute the adjusted p-value function.

Optionally, the list may contain the following components:

- `global_pvalue`: A numeric value containing the global p-value. Only present if the correction argument is set to "Global".
- `pvalue_matrix`: A numeric matrix of shape $p \times p$ containing the p-values of the interval-wise tests. Element i, j contains the p-value of the test performed on the interval indexed by $j, j + 1, \dots, j + (p - i)$. Only present if the correction argument is set to "IWT".

References

- Vsevolozhskaya, Olga A, Mark C Greenwood, GJ Bellante, Scott L Powell, Rick L Lawrence, and Kevin S Repasky. 2013. "Combining Functions and the Closure Principle for Performing Follow-up Tests in Functional Analysis of Variance." *Computational Statistics & Data Analysis* 67: 175–84.
- Vsevolozhskaya, Olga, Mark Greenwood, and Dmitri Holodov. 2014. "Pairwise comparison of treatment levels in functional analysis of variance with application to erythrocyte hemolysis." *The Annals of Applied Statistics* 8 (2): 905–25. <https://doi.org/10.1214/14-AOAS723>.

See Also

`global2()`, `twt2()`, `iwt2()`, `fdr2()` for calling directly one of the other tests, `functional_two_sample_test()` for calling the interface test and `plot.fts()` for plotting the results.

Examples

```
# Performing the PCT for two populations
# Choosing as partition the 4 seasons of the year
partition <- c(
  rep(1, 31 + 28 + 21),
  rep(2, 10 + 30 + 31 + 21),
  rep(3, 9 + 31 + 31 + 23),
  rep(4, 7 + 31 + 30 + 21),
  rep(1, 10)
)
partition <- factor(partition)

PCT_result <- PCT2(NASAtemp$paris, NASAtemp$milan, partition = partition)

# Plotting the results of the PCT
plot(
  PCT_result,
  xrange = c(0, 12),
  title = "PCT results for testing mean differences"
)

# Selecting the significant components at 5% level
which(PCT_result$adjusted_pvalues < 0.05)
```

plot.faov

*Plot for Functional ANOVA Test Results***Description**

The S3 methods `autoplot.faov()` and `plot.faov()` are methods for plotting results of functional analysis of variance tests. They visualize the functional data and the adjusted p-values obtained from the testing procedures for mean comparison of multiple groups. The plots highlight significant effects at two levels of significance, `alpha1` and `alpha2`, using shaded areas.

Usage

```
## S3 method for class 'faov'
autoplot(
  object,
  xrange = c(0, 1),
  alpha1 = 0.05,
  alpha2 = 0.01,
  plot_adjpv = FALSE,
  ylim = NULL,
  col = 1,
  ylabel = "Functional Data",
  title = NULL,
  linewidth = 0.5,
  type = "l",
  ...
)
```

```
## S3 method for class 'faov'
plot(
  x,
  xrange = c(0, 1),
  alpha1 = 0.05,
  alpha2 = 0.01,
  plot_adjpv = FALSE,
  ylim = NULL,
  col = 1,
  ylab = "Functional Data",
  main = NULL,
  lwd = 0.5,
  type = "l",
  ...
)
```

Arguments

`object, x` An object of class `faov`, usually a result of a call to `functional_anova_test()`, `iwt_aov()`, `twt_aov()` or `global_aov()`.

xrange	A length-2 numeric vector specifying the range of the x-axis for the plots. Defaults to $c(0, 1)$. This should match the domain of the functional data.
alpha1	A numeric value specifying the first level of significance used to select and display significant effects. Defaults to $\alpha_1 = 0.05$.
alpha2	A numeric value specifying the second level of significance used to select and display significant effects. Defaults to $\alpha_2 = 0.01$.
plot_adjval	A boolean value specifying whether the plots of adjusted p-values should be displayed. Defaults to FALSE.
ylim	A 2-length numeric vector specifying the range of the y-axis. Defaults to NULL, which determines automatically the range from functional data.
col	An integer specifying the color for the plot of functional data. Defaults to 1L.
ylabel, ylab	A string specifying the label of the y-axis of the functional data plot. Defaults to "Functional Data".
title, main	A string specifying the title of the functional data plot. Defaults to NULL in which case no title is displayed.
linewidth, lwd	A numeric value specifying the width of the line for the functional data plot. Note that the line width for the adjusted p-value plot will be twice this value. Defaults to 0.5.
type	A string specifying the type of plot for the functional data. Defaults to "l" for lines.
...	Other arguments passed to specific methods. Not used in this function.

Value

The `autoplot.faov()` function creates a ggplot object that displays the functional data and the adjusted p-values. The significant intervals at levels α_1 and α_2 are highlighted in the plots. The `plot.faov()` function is a wrapper around `autoplot.faov()` that prints the plot directly.

References

- Pini, A., & Vantini, S. (2017). Interval-wise testing for functional data. *Journal of Nonparametric Statistics*, 29(2), 407-424.
- Pini, A., Vantini, S., Colosimo, B. M., & Grasso, M. (2018). Domain-selective functional analysis of variance for supervised statistical profile monitoring of signal data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 67(1), 55-81.
- Abramowicz, K., Hager, C. K., Pini, A., Schelin, L., Sjøstedt de Luna, S., & Vantini, S. (2018). Nonparametric inference for functional-on-scalar linear models applied to knee kinematic hop data after injury of the anterior cruciate ligament. *Scandinavian Journal of Statistics* 45(4), 1036-1061.

See Also

[IWTimage\(\)](#) for the plot of p-values heatmaps (for IWT).

Examples

```

temperature <- rbind(NASAtemp$milan, NASAtemp$paris)
groups <- c(rep(0, 22), rep(1, 22))

# Performing the TWT
TWT_result <- functional_anova_test(
  temperature ~ groups,
  correction = "TWT",
  B = 5L
)

# Plotting the results of the TWT
plot(
  TWT_result,
  xrange = c(0, 12),
  main = "TWT results for testing mean differences"
)

```

plot.flm

Plot for Functional Linear Model Test Results

Description

The S3 methods `autoplot.flm()` and `plot.flm()` are methods for plotting results of functional-on-scalar linear model tests. They visualize the functional regression coefficients and the adjusted p-values obtained from the testing procedures. The plots highlight significant effects at two levels of significance, α_1 and α_2 , using shaded areas.

Usage

```

## S3 method for class 'flm'
autoplot(
  object,
  xrange = c(0, 1),
  alpha1 = 0.05,
  alpha2 = 0.01,
  plot_adjpv = FALSE,
  col = c(1, grDevices::rainbow(dim(object$adjusted_pval_part)[1])),
  ylim = NULL,
  ylabel = "Functional Data",
  title = NULL,
  linewidth = 1,
  type = "l",
  ...
)

## S3 method for class 'flm'
plot(

```

```

    x,
    xrange = c(0, 1),
    alpha1 = 0.05,
    alpha2 = 0.01,
    plot_adjpv = FALSE,
    ylim = NULL,
    col = 1,
    ylab = "Functional Data",
    main = NULL,
    lwd = 0.5,
    type = "l",
    ...
  )

```

Arguments

object, x	An object of class <code>flm</code> , usually a result of a call to <code>functional_lm_test()</code> , <code>iwt_lm()</code> , <code>twl_lm()</code> or <code>global_lm()</code> .
xrange	A length-2 numeric vector specifying the range of the x-axis for the plots. Defaults to <code>c(0, 1)</code> . This should match the domain of the functional data.
alpha1	A numeric value specifying the first level of significance used to select and display significant effects. Defaults to <code>alpha1 = 0.05</code> .
alpha2	A numeric value specifying the second level of significance used to select and display significant effects. Defaults to <code>alpha2 = 0.01</code> .
plot_adjpv	A boolean value specifying whether the plots of adjusted p-values should be displayed. Defaults to <code>FALSE</code> .
col	An integer specifying the color for the plot of functional data. Defaults to <code>1L</code> .
ylim	A 2-length numeric vector specifying the range of the y-axis. Defaults to <code>NULL</code> , which determines automatically the range from functional data.
ylabel, ylab	A string specifying the label of the y-axis of the functional data plot. Defaults to <code>"Functional Data"</code> .
title, main	A string specifying the title of the functional data plot. Defaults to <code>NULL</code> in which case no title is displayed.
linewidth, lwd	A numeric value specifying the width of the line for the functional data plot. Note that the line width for the adjusted p-value plot will be twice this value. Defaults to <code>0.5</code> .
type	A string specifying the type of plot for the functional data. Defaults to <code>"l"</code> for lines.
...	Other arguments passed to specific methods. Not used in this function.

Value

The `autoplot.flm()` function creates a `ggplot` object that displays the functional data and the adjusted p-values. The significant intervals at levels `alpha1` and `alpha2` are highlighted in the plots. The `plot.flm()` function is a wrapper around `autoplot.flm()` that prints the plot directly.

References

Pini, A., & Vantini, S. (2017). Interval-wise testing for functional data. *Journal of Nonparametric Statistics*, 29(2), 407-424.

Pini, A., Vantini, S., Colosimo, B. M., & Grasso, M. (2018). Domain-selective functional analysis of variance for supervised statistical profile monitoring of signal data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 67(1), 55-81.

Abramowicz, K., Hager, C. K., Pini, A., Schelin, L., Sjøstedt de Luna, S., & Vantini, S. (2018). Nonparametric inference for functional-on-scalar linear models applied to knee kinematic hop data after injury of the anterior cruciate ligament. *Scandinavian Journal of Statistics* 45(4), 1036-1061.

See Also

`IWTimage()` for the plot of p-values heatmaps (for IWT).

Examples

```
temperature <- rbind(NASAtemp$milan[, 1:100], NASAtemp$paris[, 1:100])
groups <- c(rep(0, 22), rep(1, 22))

# Performing the IWT
IWT_result <- IWTlm(temperature ~ groups, B = 2L)

# Summary of the IWT results
summary(IWT_result)

# Plot of the IWT results
plot(
  IWT_result,
  main = "NASA data",
  plot_adjval = TRUE,
  xlab = "Day",
  xrange = c(1, 365)
)
```

plot.fos

Plot for Functional One-Sample Test Results

Description

The S3 methods `autoplot.fos()` and `plot.fos()` are methods for plotting results of functional one-sample tests. They visualize the functional data and the adjusted p-values obtained from the testing procedures for testing the center of symmetry of a functional population. The plots highlight significant effects at two levels of significance, `alpha1` and `alpha2`, using shaded areas.

Usage

```
## S3 method for class 'fos'
autoplot(
  object,
  xrange = c(0, 1),
  alpha1 = 0.05,
  alpha2 = 0.01,
  ylabel = "Functional Data",
  title = NULL,
  linewidth = 0.5,
  ...
)

## S3 method for class 'fos'
plot(
  x,
  xrange = c(0, 1),
  alpha1 = 0.05,
  alpha2 = 0.01,
  ylabel = "Functional Data",
  title = NULL,
  linewidth = 0.5,
  ...
)

## S3 method for class 'IWT1'
plot(
  x,
  xrange = c(0, 1),
  alpha1 = 0.05,
  alpha2 = 0.01,
  ylab = "Functional Data",
  main = NULL,
  lwd = 1,
  col = 1,
  ylim = NULL,
  type = "l",
  ...
)
```

Arguments

object, x	An object of class <code>fos</code> , usually a result of a call to <code>functional_one_sample_test()</code> or <code>iwt1()</code> .
xrange	A length-2 numeric vector specifying the range of the x-axis for the plots. Defaults to <code>c(0, 1)</code> . This should match the domain of the functional data.
alpha1	A numeric value specifying the first level of significance used to select and display significant effects. Defaults to <code>alpha1 = 0.05</code> .

alpha2	A numeric value specifying the second level of significance used to select and display significant effects. Defaults to $\alpha_2 = 0.01$.
ylabel	A string specifying the label of the y-axis of the functional data plot. Defaults to "Functional Data".
title	A string specifying the title of the plots. Defaults to NULL in which case no title is displayed.
linewidth	A numeric value specifying the width of the line for the functional data plot. Note that the line width for the adjusted p-value plot will be twice this value. Defaults to $\text{linewidth} = 0.5$.
...	Other arguments passed to specific methods. Not used in this function.
ylab	Label of the y-axis (legacy alias for ylabel). Defaults to "Functional Data".
main	Plot title (legacy alias for title). Defaults to NULL.
lwd	Line width (legacy alias for linewidth; divided by 2 for ggplot2 scaling). Defaults to 1.
col, ylim, type	Ignored; retained for backward compatibility only.

Value

The `autoplot.fos()` function creates a ggplot object that displays the functional data (with the null mean function μ overlaid as a dashed reference line) and the adjusted p-values. The significant intervals at levels α_1 and α_2 are highlighted in both panels. The `plot.fos()` function is a wrapper around `autoplot.fos()` that prints the plot directly.

References

- Pini, A., & Vantini, S. (2017). Interval-wise testing for functional data. *Journal of Nonparametric Statistics*, 29(2), 407-424.
- Pini, A., Vantini, S., Colosimo, B. M., & Grasso, M. (2018). Domain-selective functional analysis of variance for supervised statistical profile monitoring of signal data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 67(1), 55-81.
- Abramowicz, K., Hager, C. K., Pini, A., Schelin, L., Sjøstedt de Luna, S., & Vantini, S. (2018). Nonparametric inference for functional-on-scalar linear models applied to knee kinematic hop data after injury of the anterior cruciate ligament. *Scandinavian Journal of Statistics* 45(4), 1036-1061.

See Also

[IWTimage\(\)](#) for the plot of p-value heatmaps (for IWT).

Examples

```
# Performing the IWT for one population
IWT_result <- functional_one_sample_test(
  NASAtemp$paris, mu = 4, n_perm = 10L
)

# Plotting the results
plot(IWT_result, xrange = c(0, 12), title = "Paris temperatures")
```

```
# Selecting the significant components at 5% level
which(IWT_result$adjusted_pvalues < 0.05)
```

plot.fts

Plot for Functional Two-Sample Test Results

Description

The S3 methods `autoplot.fts()` and `plot.fts()` are methods for plotting results of functional two-sample tests. They visualize the functional data and the adjusted p-values obtained from the testing procedures for mean comparison of two groups. The plots highlight significant effects at two levels of significance, `alpha1` and `alpha2`, using shaded areas.

Usage

```
## S3 method for class 'fts'
autoplot(
  object,
  xrange = c(0, 1),
  alpha1 = 0.05,
  alpha2 = 0.01,
  ylabel = "Functional Data",
  title = NULL,
  linewidth = 0.5,
  ...
)

## S3 method for class 'fts'
plot(
  x,
  xrange = c(0, 1),
  alpha1 = 0.05,
  alpha2 = 0.01,
  ylabel = "Functional Data",
  title = NULL,
  linewidth = 0.5,
  ...
)
```

Arguments

<code>object, x</code>	An object of class <code>fts</code> , usually a result of a call to <code>functional_two_sample_test()</code> , <code>iwt2()</code> , <code>twt2()</code> , <code>fdr2()</code> , <code>pct2()</code> or <code>global2()</code> .
<code>xrange</code>	A length-2 numeric vector specifying the range of the x-axis for the plots. Defaults to <code>c(0, 1)</code> . This should match the domain of the functional data.

alpha1	A numeric value specifying the first level of significance used to select and display significant effects. Defaults to $\alpha_1 = 0.05$.
alpha2	A numeric value specifying the second level of significance used to select and display significant effects. Defaults to $\alpha_2 = 0.01$.
ylabel	A string specifying the label of the y-axis of the functional data plot. Defaults to "Functional Data".
title	A string specifying the title of the functional data plot. Defaults to NULL in which case no title is displayed.
linewidth	A numeric value specifying the width of the line for the functional data plot. Note that the line width for the adjusted p-value plot will be twice this value. Defaults to $\text{linewidth} = 0.5$.
...	Other arguments passed to specific methods. Not used in this function.

Value

The `autoplot.fts()` function creates a `ggplot` object that displays the functional data and the adjusted p-values. The significant intervals at levels `alpha1` and `alpha2` are highlighted in the plots. The `plot.fts()` function is a wrapper around `autoplot.fts()` that prints the plot directly.

References

- Pini, A., & Vantini, S. (2017). Interval-wise testing for functional data. *Journal of Nonparametric Statistics*, 29(2), 407-424.
- Pini, A., Vantini, S., Colosimo, B. M., & Grasso, M. (2018). Domain-selective functional analysis of variance for supervised statistical profile monitoring of signal data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 67(1), 55-81.
- Abramowicz, K., Hager, C. K., Pini, A., Schelin, L., Sjostedt de Luna, S., & Vantini, S. (2018). Nonparametric inference for functional-on-scalar linear models applied to knee kinematic hop data after injury of the anterior cruciate ligament. *Scandinavian Journal of Statistics* 45(4), 1036-1061.

See Also

[IWTimage\(\)](#) for the plot of p-values heatmaps (for IWT).

Examples

```
# Performing the TWT for two populations
TWT_result <- functional_two_sample_test(
  NASAtemp$paris, NASAtemp$milan,
  correction = "TWT", n_perm = 10L
)

# Plotting the results of the TWT
plot(
  TWT_result,
  xrange = c(0, 12),
  title = "TWT results for testing mean differences"
)
```

```

# Selecting the significant components at 5% level
which(TWT_result$adjusted_pval < 0.05)

# Performing the IWT for two populations
IWT_result <- functional_two_sample_test(
  NASAtemp$paris, NASAtemp$milan,
  correction = "IWT", n_perm = 10L
)

# Plotting the results of the IWT
plot(
  IWT_result,
  xrange = c(0, 12),
  title = "IWT results for testing mean differences"
)

# Selecting the significant components at 5% level
which(IWT_result$adjusted_pval < 0.05)

```

summary.faov

*Summarizing Functional Analysis of Variance Fits***Description**

summary method for class faov. Returns a summary of the results of the local testing procedure for functional analysis of variance: the minimum adjusted p-values of the F-test on the whole model and on each factor are reported.

Usage

```
## S3 method for class 'faov'
summary(object, ...)
```

Arguments

object	An object of class faov, usually, a result of a call to <code>functional_anova_test()</code> .
...	Further arguments passed to or from other methods.

Value

A list of summary statistics of the fitted functional analysis of variance given in object, using the component call from its arguments, plus:

- **factors:** A $L \times 2$ data frame with one row per factor reporting the minimum adjusted p-value of the corresponding F-test and a significance code.
- **R2:** A 2×1 matrix giving the range of the functional R-squared.
- **ftest:** A 1×2 data frame reporting the minimum adjusted p-value of the global F-test and a significance code.

References

- Pini, A., & Vantini, S. (2017). Interval-wise testing for functional data. *Journal of Nonparametric Statistics*, 29(2), 407-424.
- Pini, A., Vantini, S., Colosimo, B. M., & Grasso, M. (2018). Domain-selective functional analysis of variance for supervised statistical profile monitoring of signal data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 67(1), 55-81.
- Abramowicz, K., Hager, C. K., Pini, A., Schelin, L., Sjostedt de Luna, S., & Vantini, S. (2018). Nonparametric inference for functional-on-scalar linear models applied to knee kinematic hop data after injury of the anterior cruciate ligament. *Scandinavian Journal of Statistics* 45(4), 1036-1061.

See Also

[IWTimage\(\)](#) for the plot of p-value heatmaps and [plot.faov\(\)](#) for the plot of analysis of variance results.

Examples

```
temperature <- rbind(NASatemp$milan, NASatemp$paris)
groups <- c(rep(0, 22), rep(1, 22))

# Performing the IWT
IWT_result <- functional_anova_test(
  temperature ~ groups,
  B = 10L,
  correction = "IWT"
)

# Summary of the IWT results
summary(IWT_result)
```

summary.flm

Summarizing Functional Linear Model Fits

Description

summary method for class flm. Returns a summary of the results of the local testing procedure for a functional-on-scalar linear model: the minimum adjusted p-values of the F-test on the whole model and the t-tests on each covariate are reported.

Usage

```
## S3 method for class 'flm'
summary(object, ...)
```

Arguments

object	An object of class flm, usually, a result of a call to functional_lm_test() .
...	Further arguments passed to or from other methods.

Value

A list of summary statistics of the fitted functional linear model given in object, using the component call from its arguments, plus:

- `ttest`: A $(L + 1) \times 2$ data frame with one row per model term (intercept plus each predictor) reporting the minimum adjusted p-value of the corresponding t-test and a significance code.
- `R2`: A 2×1 matrix giving the range of the functional R-squared.
- `ftest`: A 1×2 data frame reporting the minimum adjusted p-value of the functional F-test and a significance code.

References

Pini, A., & Vantini, S. (2017). Interval-wise testing for functional data. *Journal of Nonparametric Statistics*, 29(2), 407-424.

Pini, A., Vantini, S., Colosimo, B. M., & Grasso, M. (2018). Domain-selective functional analysis of variance for supervised statistical profile monitoring of signal data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 67(1), 55-81.

Abramowicz, K., Hager, C. K., Pini, A., Schelin, L., Sjøstedt de Luna, S., & Vantini, S. (2018). Nonparametric inference for functional-on-scalar linear models applied to knee kinematic hop data after injury of the anterior cruciate ligament. *Scandinavian Journal of Statistics* 45(4), 1036-1061.

See Also

`IWTimage()` for the plot of p-value heatmaps and `plot.flm()` for the plot of functional linear model results.

Examples

```
temperature <- rbind(NASAtemp$milan[, 1:100], NASAtemp$paris[, 1:100])
groups <- c(rep(0, 22), rep(1, 22))

# Performing the IWT
IWT_result <- functional_lm_test(
  temperature ~ groups,
  B = 2L,
  correction = "IWT"
)

# Summary of the IWT results
summary(IWT_result)
```

tw2

*Two population Threshold Wise Testing procedure***Description**

The function implements the Threshold Wise Testing procedure for testing mean differences between two functional populations. Functional data are tested locally and unadjusted and adjusted p-value functions are provided. The unadjusted p-value function controls the point-wise error rate. The adjusted p-value function controls the family-wise error rate asymptotically.

Usage

```
tw2(
  data1,
  data2,
  mu = 0,
  dx = NULL,
  n_perm = 1000L,
  paired = FALSE,
  alternative = c("two.sided", "less", "greater"),
  standardize = FALSE,
  verbose = FALSE,
  aggregation_strategy = c("integral", "max")
)

TWT2(
  data1,
  data2,
  mu = 0,
  dx = NULL,
  B = 1000L,
  paired = FALSE,
  alternative = c("two.sided", "less", "greater"),
  statistic = c("Integral", "Max", "Integral_std", "Max_std"),
  verbose = FALSE
)
```

Arguments

data1	Either a numeric matrix or an object of class <code>fda::fd</code> specifying the data in the first sample. If the data is provided within a matrix, it should be of shape $n_1 \times J$ and it should contain in each row one of the n_1 functions in the sample and in columns the evaluation of each function on a same uniform grid of size J .
data2	Either a numeric matrix or an object of class <code>fda::fd</code> specifying the data in the second sample. If the data is provided within a matrix, it should be of shape $n_2 \times J$ and it should contain in each row one of the n_2 functions in the sample

	and in columns the evaluation of each function on a same uniform grid of size J .
mu	Either a numeric value or a numeric vector or an object of class <code>fda:fd</code> specifying the functional mean difference under the null hypothesis. If mu is a constant, then a constant function is used. If mu is a numeric vector, it must correspond to evaluation of the mean difference function on the same grid that has been used to evaluate the data samples. Defaults to $\emptyset$.
dx	A numeric value specifying the step of the uniform grid on which the data are evaluated. If NULL, the step is automatically inferred from the data. Defaults to NULL.
n_perm	An integer value specifying the number of permutations to use for the local testing procedure. Defaults to 1000L.
paired	A boolean value specifying whether a paired test should be performed. Defaults to FALSE.
alternative	A string specifying the type of alternative hypothesis. Choices are "two.sided", "less" or "greater". Defaults to "two.sided".
standardize	A boolean value specifying whether to standardize the test statistic. Defaults to FALSE.
verbose	A boolean value specifying whether to print the progress of the computation. Defaults to FALSE.
aggregation_strategy	A string specifying the strategy to aggregate the point-wise test statistics for the correction procedure. Possible values are "integral" and "max". Defaults to "integral".
B	An integer value specifying the number of permutations to use for the local testing procedure. Defaults to 1000L.
statistic	A string specifying the test statistic to use. Possible values are: • "Integral": Integral of the squared sample mean difference. • "Max": Maximum of the squared sample mean difference. • "Integral_std": Integral of the squared t-test statistic. • "Max_std": Maximum of the squared t-test statistic. Defaults to "Integral".

Value

An object of class `fts` containing the following components:

- `data`: A numeric matrix of shape $n \times J$ containing the evaluation of the $n = n_1 + n_2$ functions on a **common** uniform grid of size p .
- `group_labels`: An integer vector of size $n = n_1 + n_2$ containing the group membership of each function.
- `mu`: A numeric vector of shape J containing the evaluation of the functional mean difference under the null hypothesis on the same uniform grid used to evaluate the functional samples.
- `unadjusted_pvalues`: A numeric vector of size J containing the evaluation of the unadjusted p-value function on the **same** uniform grid used to evaluate the functional samples.

- `adjusted_pvalues`: A numeric vector of size J containing the evaluation of the adjusted p-value function on the **same** uniform grid used to evaluate the functional samples.
- `correction_method`: A string containing the correction method used to compute the adjusted p-value function.

Optionally, the list may contain the following components:

- `global_pvalue`: A numeric value containing the global p-value. Only present if the correction argument is set to "Global".
- `pvalue_matrix`: A numeric matrix of shape $p \times p$ containing the p-values of the interval-wise tests. Element i, j contains the p-value of the test performed on the interval indexed by $j, j + 1, \dots, j + (p - i)$. Only present if the correction argument is set to "IWT".

References

- Abramowicz, Konrad, Alessia Pini, Lina Schelin, Sara Sjöstedt de Luna, Aymeric Stamm, and Simone Vantini. 2023. "Domain Selection and Familywise Error Rate for Functional Data: A Unified Framework." *Biometrics* 79 (2): 1119–32.

See Also

[global2\(\)](#), [iwt2\(\)](#), [pct2\(\)](#), [fdr2\(\)](#) for calling directly one of the other tests, [functional_two_sample_test\(\)](#) for calling the interface test and [plot.fts\(\)](#) for plotting the results.

Examples

```
# Performing the TWT for two populations
TWT_result <- twt2(NASAtemp$paris, NASAtemp$milan)

# Plotting the results of the TWT
plot(
  TWT_result,
  xrange = c(0, 12),
  title = "TWT results for testing mean differences"
)

# Selecting the significant components at 5% level
which(TWT_result$adjusted_pvalues < 0.05)
```

TWTaov

Threshold Wise Testing procedure for testing functional analysis of variance

Description

The function implements the Threshold Wise Testing procedure for testing mean differences between several functional populations in a one-way or multi-way functional analysis of variance framework. Functional data are tested locally and unadjusted and adjusted p-value functions are provided. The unadjusted p-value function controls the point-wise error rate. The adjusted p-value function controls the threshold-wise error rate.

Usage

```
TWtaov(formula, dx = NULL, B = 1000L, method = c("residuals", "responses"))

twtaov(
  formula,
  dx = NULL,
  n_perm = 1000L,
  method = c("residuals", "responses")
)
```

Arguments

formula	An object of class <code>stats::formula</code> (or one that can be coerced to that class) specifying the model to be fitted in a symbolic fashion. The response (left-hand side) can be either a matrix of dimension $n \times J$ containing the pointwise evaluations of n functions on the same grid of J points, or an object of class <code>fda::fd</code> .
dx	A numeric value specifying the discretization step of the grid used to evaluate functional data when it is provided as objects of class <code>fda::fd</code> . Defaults to <code>NULL</code> , in which case a default value of <code>0.01</code> is used which corresponds to a grid of size <code>100L</code> . Unused if functional data is provided in the form of matrices.
B	An integer value specifying the number of permutations used to evaluate the p-values of the permutation tests. Defaults to <code>1000L</code> . Passed as <code>n_perm</code> in <code>iwt_aov()</code> , <code>twtaov()</code> and <code>global_aov()</code> .
method	A string specifying the permutation scheme. "residuals" permutes residuals under the reduced model (Freedman-Lane scheme); "responses" permutes the responses (Manly scheme). Defaults to "residuals".
n_perm	An integer value specifying the number of permutations for the permutation tests. Defaults to <code>1000L</code> .

Value

An object of class `faov` containing the following components:

- `call`: The matched call.
- `design_matrix`: The design matrix of the functional ANOVA model.
- `unadjusted_pval_F`: A numeric vector of length J containing the unadjusted p-value function of the global F-test evaluated on the grid.
- `adjusted_pval_F`: A numeric vector of length J containing the adjusted p-value function of the global F-test evaluated on the grid.
- `unadjusted_pval_factors`: A numeric matrix with one row per factor containing the unadjusted p-value functions of the per-factor F-tests.
- `adjusted_pval_factors`: A numeric matrix with one row per factor containing the adjusted p-value functions of the per-factor F-tests.
- `data_eval`: A numeric matrix containing the functional data evaluated on the grid.

- `coeff_regr_eval`: A numeric matrix containing the functional regression coefficients evaluated on the grid.
- `fitted_eval`: A numeric matrix containing the fitted values of the functional regression evaluated on the grid.
- `residuals_eval`: A numeric matrix containing the residuals of the functional regression evaluated on the grid.
- `R2_eval`: A numeric vector containing the functional R-squared evaluated on the grid.

Optionally, the list may contain the following components:

- `pval_matrix_F`: A matrix of dimensions $p \times p$ of p-values of the interval-wise F-tests. Element (i, j) contains the p-value of the test on the interval $(j, j + 1, \dots, j + (p - i))$. Present only if correction is "IWT".
- `pval_matrix_factors`: An array of dimensions $L \times p \times p$ of p-values of the per-factor interval-wise F-tests. Element (l, i, j) contains the p-value of the joint test on factor l and interval $(j, j + 1, \dots, j + (p - i))$. Present only if correction is "IWT".
- `global_pval_F`: Global p-value of the overall F-test. Present only if correction is "Global".
- `global_pval_factors`: A numeric vector of global p-values of the per-factor F-tests. Present only if correction is "Global".

References

- Abramowicz, K., Pini, A., Schelin, L., Stamm, A., & Vantini, S. (2022). "Domain selection and familywise error rate for functional data: A unified framework. *Biometrics* 79(2), 1119-1132.
- D. Freedman and D. Lane (1983). A Nonstochastic Interpretation of Reported Significance Levels. *Journal of Business & Economic Statistics* 1.4, 292-298.
- B. F. J. Manly (2006). *Randomization, Bootstrap and Monte Carlo Methods in Biology*. Vol. 70. CRC Press.

See Also

`iwt_aov()`, `twt_aov()` and `global_aov()` for calling a specific correction directly. `plot.faov()` for plotting the results and `summary.faov()` for summarizing the results.

Examples

```
temperature <- rbind(NASAtemp$milan, NASAtemp$paris)
groups <- c(rep(0, 22), rep(1, 22))

# Performing the TWT
TWT_result <- TWTaov(temperature ~ groups, B = 100L)

# Summary of the TWT results
summary(TWT_result)

# Plot of the TWT results
layout(1)
plot(TWT_result)
```

```
# All graphics on the same device
layout(matrix(1:4, nrow = 2, byrow = FALSE))
plot(
  TWT_result,
  main = "NASA data",
  plot_adjval = TRUE,
  xlab = "Day",
  xrange = c(1, 365)
)
```

TWTlm

Threshold-wise testing procedure for testing functional-on-scalar linear models

Description

The function is used to fit and test functional linear models. It can be used to carry out regression, and analysis of variance. It implements the Threshold-wise testing procedure (TWT) for testing the significance of the effects of scalar covariates on a functional population.

Usage

```
TWTlm(formula, dx = NULL, B = 1000L, method = c("residuals", "responses"))

twl_lm(
  formula,
  dx = NULL,
  n_perm = 1000L,
  method = c("residuals", "responses")
)
```

Arguments

- | | |
|---------|--|
| formula | An object of class <code>stats::formula</code> (or one that can be coerced to that class) specifying the model to be fitted in a symbolic fashion. The response (left-hand side) can be either a matrix of dimension $n \times J$ containing the pointwise evaluations of n functions on the same grid of J points, or an object of class <code>fda::fd</code> . |
| dx | A numeric value specifying the discretization step of the grid used to evaluate functional data when it is provided as objects of class <code>fda::fd</code> . Defaults to <code>NULL</code> , in which case a default value of <code>0.01</code> is used which corresponds to a grid of size <code>100L</code> . Unused if functional data is provided in the form of matrices. |
| B | An integer value specifying the number of permutations used to evaluate the p-values of the permutation tests. Defaults to <code>1000L</code> . Passed as <code>n_perm</code> in <code>iwt_aov()</code> , <code>twl_aov()</code> and <code>global_aov()</code> . |

method	A string specifying the permutation scheme. "residuals" permutes residuals under the reduced model (Freedman-Lane scheme); "responses" permutes the responses (Manly scheme). Defaults to "residuals".
n_perm	An integer value specifying the number of permutations for the permutation tests. Defaults to 1000L.

Value

An object of class `flm` containing the following components:

- `call`: The matched call.
- `design_matrix`: The design matrix of the functional-on-scalar linear model.
- `unadjusted_pval_F`: A numeric vector of length J containing the unadjusted p-value function of the global F-test evaluated on the grid.
- `adjusted_pval_F`: A numeric vector of length J containing the adjusted p-value function of the global F-test evaluated on the grid.
- `unadjusted_pval_part`: A numeric matrix with one row per model term containing the unadjusted p-value functions of the per-term t-tests.
- `adjusted_pval_part`: A numeric matrix with one row per model term containing the adjusted p-value functions of the per-term t-tests.
- `data_eval`: A numeric matrix containing the functional response evaluated on the grid.
- `coeff_regr_eval`: A numeric matrix containing the functional regression coefficients evaluated on the grid.
- `fitted_eval`: A numeric matrix containing the fitted values of the functional regression evaluated on the grid.
- `residuals_eval`: A numeric matrix containing the residuals of the functional regression evaluated on the grid.
- `R2_eval`: A numeric vector containing the functional R-squared evaluated on the grid.

Optionally, the list may contain the following components:

- `pval_matrix_F`: A matrix of dimensions $p \times p$ of p-values of the interval-wise F-tests. Element (i, j) contains the p-value of the test on the interval $(j, j + 1, \dots, j + (p - i))$. Present only if correction is "IWT".
- `pval_matrix_part`: An array of dimensions $(L + 1) \times p \times p$ of p-values of the per-term interval-wise t-tests. Element (l, i, j) contains the p-value of the joint test on term l and interval $(j, j + 1, \dots, j + (p - i))$. Present only if correction is "IWT".
- `global_pval_F`: Global p-value of the overall F-test. Present only if correction is "Global".
- `global_pval_part`: A numeric vector of global p-values of the per-term t-tests. Present only if correction is "Global".

References

- Abramowicz, K., Pini, A., Schelin, L., Stamm, A., & Vantini, S. (2022). “Domain selection and familywise error rate for functional data: A unified framework. *Biometrics* 79(2), 1119-1132.
- D. Freedman and D. Lane (1983). A Nonstochastic Interpretation of Reported Significance Levels. *Journal of Business & Economic Statistics* 1(4), 292-298.
- B. F. J. Manly (2006). Randomization, *Bootstrap and Monte Carlo Methods in Biology*. Vol. 70. CRC Press.

See Also

[iwt_lm\(\)](#), [twl_lm\(\)](#) and [global_lm\(\)](#) for calling a specific correction directly. [plot.flm\(\)](#) for plotting the results and [summary.flm\(\)](#) for summarizing the results.

Examples

```
# Defining the covariates
temperature <- rbind(NASAtemp$milan, NASAtemp$paris)
groups <- c(rep(0, 22), rep(1, 22))

# Performing the TWT
TWT_result <- TWTlm(temperature ~ groups, B = 100L)
# Summary of the TWT results
summary(TWT_result)

# Plot of the TWT results
plot(
  TWT_result,
  main = "NASA data",
  plot_adjval = TRUE,
  xlab = "Day",
  xrange = c(1, 365)
)
```

Index

* datasets

NASAtemp, 44

autoplot.fao (plot.fao), 48
autoplot.flm (plot.flm), 50
autoplot.fos (plot.fos), 52
autoplot.fos(), 13
autoplot.fts (plot.fts), 55

class, 32, 35

fda::fd, 3, 6, 9, 11, 12, 14, 17, 20, 23, 25, 28,
29, 37, 42, 45, 60, 61, 63, 65

FDR2 (fdr2), 2
fdr2, 2
fdr2(), 16, 22, 30, 47, 55, 62
formula, 31, 35
functional_anova_test, 5
functional_anova_test(), 48, 57
functional_lm_test, 8
functional_lm_test(), 51, 58
functional_one_sample_test, 11
functional_one_sample_test(), 26, 40, 53
functional_two_sample_test, 13
functional_two_sample_test(), 5, 22, 30,
40, 47, 55, 62

Global2 (global2), 19
global2, 19
global2(), 5, 16, 30, 47, 55, 62
global_aov, 17
global_aov(), 6, 7, 9, 18, 19, 23, 37, 39, 42,
48, 63–65
global_lm (Globallm), 22
global_lm(), 10, 24, 43, 51, 67
Globalaov (global_aov), 17
Globallm, 22

ITP1bspline, 25, 34
ITP1fourier, 34
ITP1fourier (ITP1bspline), 25
ITP2bspline, 27, 34
ITP2fourier, 34
ITP2fourier (ITP2bspline), 27
ITP2pafourier, 34
ITP2pafourier (ITP2bspline), 27
ITPaovbspline, 31
ITPimage, 33
ITPlmbspline, 34
IWT1 (ITP1bspline), 25
iwt1 (ITP1bspline), 25
IWT1(), 40
iwt1(), 13, 40, 53
IWT2 (ITP2bspline), 27
iwt2 (ITP2bspline), 27
iwt2(), 5, 16, 22, 40, 47, 55, 62
iwt_aov (IWTaov), 37
iwt_aov(), 6, 7, 9, 18, 19, 23, 37, 39, 42, 48,
63–65
iwt_lm (IWTlm), 41
iwt_lm(), 10, 24, 43, 51, 67
IWTaov, 37
IWTaov(), 40
IWTimage, 40
IWTimage(), 26, 49, 52, 54, 56, 58, 59
IWTlm, 41

NASAtemp, 44

PCT2 (pct2), 44
pct2, 44
pct2(), 5, 16, 22, 30, 55, 62
plot.fao, 48
plot.fao(), 7, 19, 39, 41, 58, 64
plot.flm, 50
plot.flm(), 10, 24, 41, 43, 59, 67
plot.fos, 52
plot.fos(), 13, 26, 41
plot.fts, 55
plot.fts(), 5, 16, 22, 30, 41, 47, 62
plot.IWT1 (plot.fos), 52

stats::formula, [6](#), [9](#), [17](#), [23](#), [37](#), [42](#), [63](#), [65](#)
summary.faov, [57](#)
summary.faov(), [7](#), [19](#), [39](#), [64](#)
summary.flm, [58](#)
summary.flm(), [10](#), [24](#), [43](#), [67](#)

TWT2 (tw2), [60](#)
tw2, [60](#)
tw2(), [5](#), [16](#), [22](#), [30](#), [47](#), [55](#)
tw2_aov (TWTaov), [62](#)
tw2_aov(), [6](#), [7](#), [9](#), [18](#), [19](#), [23](#), [37](#), [39](#), [42](#), [48](#),
 [63–65](#)
tw2_lm (TWTlm), [65](#)
tw2_lm(), [10](#), [24](#), [43](#), [51](#), [67](#)
TWTaov, [62](#)
TWTlm, [65](#)