# Package 'ghcm'

October 13, 2022

**Type** Package

**Title** Functional Conditional Independence Testing with the GHCM

**Version** 3.0.0

**Description** A statistical hypothesis test for conditional independence.
Given residuals from a sufficiently powerful regression, it tests whether
the covariance of the residuals is vanishing. It can be applied to both
discretely-observed functional data and multivariate data.
Details of the method can be found in Anton Rask Lundborg, Rajen D. Shah and Jonas
Peters (2021) <arXiv:2101.07108>.

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**Imports** graphics, MASS, refund, stats, utils, CompQuadForm, Rcpp,
splines

**Depends** R (>= 4.0.0)

**RoxygenNote** 7.1.2

**Suggests** testthat, knitr, rmarkdown, bookdown,
GeneralisedCovarianceMeasure, ggplot2, reshape2, dplyr, tidyr

**URL** https://github.com/arlundborg/ghcm

**BugReports** https://github.com/arlundborg/ghcm/issues

**VignetteBuilder** knitr

**LinkingTo** Rcpp

**NeedsCompilation** yes

**Author** Anton Rask Lundborg [aut, cre],
Rajen D. Shah [aut],
Jonas Peters [aut]

**Maintainer** Anton Rask Lundborg <a.lundborg@statslab.cam.ac.uk>

**Repository** CRAN

**Date/Publication** 2022-02-20 16:20:02 UTC

# R topics documented:



---

`ghcm` *ghcm: A package for Functional Conditional Independence Testing*

---

## Description

To learn more about ghcm, start with the vignette: ‘browseVignettes(package = "ghcm")‘

---

`ghcm_sim_data` *GHCM simulated data*

---

## Description

A simulated dataset containing a combination of functional and scalar variables. Y_1 and Y_2 are scalar random variables and are both functions of Z. X, Z and W are functional, Z is a function of X and W is a function of Z.

## Usage

```
ghcm_sim_data

ghcm_sim_data_irregular
```

## Format

`ghcm_sim_data` is a data frame with 500 rows of 5 variables:

**Y_1** Numeric vector.

**Y_2** Numeric vector.

**Z** 500 x 101 matrix.

**X** 500 x 101 matrix.

**W** 500 x 101 matrix.

`ghcm_sim_data_irregular` is a list with 5 elements:

**Y_1** Numeric vector.

**Y_2** Numeric vector.

**Z** 500 x 101 matrix.

**X** A data frame with

**.obs** Integer between 1 and 500 indicating which curve the row corresponds to.

**.index** Function argument that the curve is evaluated at.

**.value** Value of the function.

**W** A data frame with

**.obs** Integer between 1 and 500 indicating which curve the row corresponds to.

**.index** Function argument that the curve is evaluated at.

**.value** Value of the function.

## Details

In `ghcm_sim_data` the functional variables each consists of 101 observations on an equidistant grid on [0, 1].

In `ghcm_sim_data_irregular` the functional variables X and W are instead only observed on a subsample of the original equidistant grid.

## Source

The generation script can be found in the `data-raw` folder of the package.

---

`ghcm_test` *Conditional Independence Test using the GHCM*

---

## Description

Test whether X is independent of Y given Z using the Generalised Hilbertian Covariance Measure. The function is applied to residuals from regressing each of X and Y on Z respectively. Its validity is contingent on the performance of the regression methods. For a more in-depth explanation see the package vignette or the paper mentioned in the references.

## Usage

```
ghcm_test(
  resid_X_on_Z,
  resid_Y_on_Z,
  X_limits = NULL,
  Y_limits = NULL,
  alpha = 0.05
)
```

## Arguments

`resid_X_on_Z, resid_Y_on_Z`

Residuals from regressing X (Y) on Z with a suitable regression method. If X (Y) is uni- or multivariate or functional on a constant, fixed grid, the residuals should be supplied as a vector or matrix with no missing values. If instead X (Y) is functional and observed on varying grids or with missing values, the residuals should be supplied as a "melted" data frame with

**.obs** Integer indicating which curve the row corresponds to.

**.index** Function argument that the curve is evaluated at.

**.value** Value of the function.

Note that in the irregular case, a minimum of 4 observations per curve is required.

`X_limits, Y_limits`

The minimum and maximum values of the function argument of the X (Y) curves. Ignored if X (Y) is not functional.

`alpha` Numeric in the unit interval. Significance level of the test.

## Value

An object of class `ghcm` containing:

`test_statistic` Numeric, test statistic of the test.

`p` Numeric in the unit interval, estimated p-value of the test.

`alpha` Numeric in the unit interval, significance level of the test.

`reject` `TRUE` if `p` < `alpha`, `FALSE` otherwise.

## References

Please cite the following paper: Anton Rask Lundborg, Rajen D. Shah and Jonas Peters: "Conditional Independence Testing in Hilbert Spaces with Applications to Functional Data Analysis" https://arxiv.org/abs/2101.07108

## Examples

```
library(refund)
set.seed(1)
data(ghcm_sim_data)
grid <- seq(0, 1, length.out = 101)

# Test independence of two scalars given a functional variable

m_1 <- pfr(Y_1 ~ lf(Z), data=ghcm_sim_data)
m_2 <- pfr(Y_2 ~ lf(Z), data=ghcm_sim_data)
ghcm_test(resid(m_1), resid(m_2))

# Test independence of a regularly observed functional variable and a
# scalar variable given a functional variable
```

```
m_X <- pffr(X ~ ff(Z), data=ghcm_sim_data, chunk.size=31000)
ghcm_test(resid(m_X), resid(m_1))

# Test independence of two regularly observed functional variables given
# a functional variable

m_W <- pffr(W ~ ff(Z), data=ghcm_sim_data, chunk.size=31000)
ghcm_test(resid(m_X), resid(m_W))



data(ghcm_sim_data_irregular)
n <- length(ghcm_sim_data_irregular$Y_1)
Z_df <- data.frame(.obs=1:n)
Z_df$Z <- ghcm_sim_data_irregular$Z
# Test independence of an irregularly observed functional variable and a
# scalar variable given a functional variable

m_1 <- pfr(Y_1 ~ lf(Z), data=ghcm_sim_data_irregular)
m_X <- pffr(X ~ ff(Z), ydata = ghcm_sim_data_irregular$X,
  data=Z_df, chunk.size=31000)
ghcm_test(resid(m_X), resid(m_1), X_limits=c(0, 1))

# Test independence of two irregularly observed functional variables given
# a functional variable

m_W <- pffr(W ~ ff(Z), ydata = ghcm_sim_data_irregular$W,
  data=Z_df, chunk.size=31000)
ghcm_test(resid(m_X), resid(m_W), X_limits=c(0, 1), Y_limits=c(0, 1))
```

---

`inner_product_matrix_splines` *Computes the matrix of L2 inner products of the splines given in list_of_splines as produced by splines::interpSpline. The splines are assumed to be functions on the interval [from, to].*

---

## Description

Computes the matrix of L2 inner products of the splines given in list_of_splines as produced by splines::interpSpline. The splines are assumed to be functions on the interval [from, to].

## Usage

```
inner_product_matrix_splines(list_of_splines, from, to)
```

## Arguments

| | |
|---|---|
| `list_of_splines` | list of interpSpline objects. |
| `from, to` | limits of integration. |

## Value

matrix of inner products.

# Index