Package: dsPUcopulaClient (via r-universe)

May 20, 2026

Type Package

Title Client Utilities for Partition-of-Unity Copula Modelling in DataSHIELD

Version 0.1.0

Description Provides client-side helper functions for interacting with the dsPUcopula DataSHIELD server package. The helpers wrap the remote procedures for fitting partition-of-unity copula models, estimating marginal distributions, simulating copula-based samples and retrieving synthetic data in a privacy-preserving manner.

License MIT + file LICENSE

URL https://github.com/bips-hb/dsPUcopulaClient

BugReports https://github.com/bips-hb/dsPUcopulaClient/issues

Depends R (>= 4.1.0)

Imports DSI

Encoding UTF-8

Language en-GB

Roxygen list(markdown = TRUE)

LazyData false

Repository https://amaendle.r-universe.dev

Date/Publication 2025-10-07 13:42:12 UTC

RemoteUrl https://github.com/bips-hb/dsPUcopulaClient

RemoteRef HEAD

RemoteSha 1ec865cd3fb9f599454028eca2d6eeec954e65f2

Contents

`ds.estimateMarginals` *Estimate marginal distributions inside DataSHIELD*

Description

Computes marginal distribution estimates for the variables identified by the server-side symbol `x` through the `estimateMarginalsDS` helper. The result is stored remotely as `marginals_list` for later reuse.

Usage

```
ds.estimateMarginals(x, datasources = NULL, method = "ecdf")
```

Arguments

`x`	`character(1)`. The name of the server-side object that holds the variables for which the marginals should be estimated.
`datasources`	Optional list of DataSHIELD connections as returned by `DSI::datashield.connections_find()`.
`method`	`character(1)`. The marginal estimation approach to use. The default is `"ecdf"`, but the value must match a method supported on the server side.

Value

Invisibly returns `TRUE` once the remote object has been created.

Examples

```
## Not run:
ds.estimateMarginals("PU_copula_model$data")

## End(Not run)
```

`ds.fitPUcopula` *Fit a Partition-of-Unity Copula Model on the server*

Description

Creates or replaces the `PU_copula_model` object in the connected DataSHIELD sessions by delegating to the server-side `fitPUcopulaDS` function.

Usage

```
ds.fitPUcopula(
  data,
  driver_strength_factor = 0.5,
  bin_size = 3,
  datasources = NULL,
  ...
)
```

Arguments

data	character(1). The name of the server-side data frame that contains the copula samples.
driver_strength_factor	numeric(1). Tuning parameter for the driver strength estimation. Values must be positive.
bin_size	integer(1). Bin size used during the density estimation steps of the algorithm.
datasources	Optional list of DataSHIELD connections. When omitted the currently active connections obtained via DSI::datashield.connections_find() are used.
...	Additional arguments passed through to the server-side fitPUcopulaDS function.

Value

Invisibly returns TRUE once the remote object has been created.

Examples

```
## Not run:
ds.fitPUcopula(data = "D", driver_strength_factor = 0.6, bin_size = 4)

## End(Not run)
```

---

ds.generateSynthetic *Retrieve synthetic data from the server*

Description

Aggregates the synthetic observations generated by the server-side generateSyntheticDS function and returns them to the client.

Usage

```
ds.generateSynthetic(n, datasources = NULL)
```

Arguments

n	integer(1). Number of synthetic observations to retrieve.
datasources	Optional list of DataSHIELD connections to use.

Value

A data frame with n rows containing the synthesised observations.

Examples

```
## Not run:
synth <- ds.generateSynthetic(500)

## End(Not run)
```

---

ds.simulateCopula *Simulate from the fitted copula model*

Description

Generates a simulated sample of size n from the copula model previously stored in PU_copula_model. The simulated uniform scores are kept on the server-side as PU_copula_model_u_sims_clientcode.

Usage

```
ds.simulateCopula(n, datasources = NULL)
```

Arguments

n	integer(1). Number of draws to request from the server.
datasources	Optional list of DataSHIELD connections to use.

Value

Invisibly returns TRUE once the simulated sample has been stored on the server.

Examples

```
## Not run:
ds.simulateCopula(1000)

## End(Not run)
```

Index