Package ‘DTFM’

May 7, 2026

Type Package

Title Distributed Online Covariance Matrix Tests for Truncated Factor Model

Version 0.1.5

Description The truncated factor model is a statistical model designed to handle specific data structures in data analysis. 'DTFM' is a powerful tool designed to efficiently process and analyze distributed datasets. The philosophy of the package is described in Guo et al. (2023) <doi:10.1007/s00180-022-01270-z>.

License MIT + file LICENSE

Suggests rmarkdown, psych

Depends R (>= 3.5.0)

RoxygenNote 7.3.3

Encoding UTF-8

Language en-US

Author Beibei Wu [aut],
Guangbao Guo [aut, cre]

Maintainer Guangbao Guo <ggb11111111@163.com>

Imports SOPC, MASS, mvtnorm, matrixcalc, stats, tmvtnorm

NeedsCompilation no

LazyData true

Repository CRAN

Date/Publication 2026-03-10 20:50:16 UTC

Contents

admission_predict 2
Chinese_Herbal_Tea 2
CLX 4
cm13 5
concrete 6

FanPC_TFM 6
GOOG 7
LC 8
new_energy_vehicle 9
protein 10
real_estate_valuation 11
review 11
riboflavin 13
riboflavinv100 13
syk 14
taxi_trip_pricing 15
TFM 15
ttest.TFM 16
winequality.white 17
yacht_hydrodynamics 18

Index **19**

`admission_predict` *Admission Prediction Data*

Description

A dataset containing parameters relevant to graduate admission prediction.

Usage

```
admission_predict
```

Format

A data frame representing student profiles.

`Chinese_Herbal_Tea` *Chinese Herbal Tea Consumer Survey Data*

Description

A questionnaire survey on consumers' perceptions and intentions regarding Chinese herbal tea (CHT). The dataset is used in the real-data analysis for covariance-based hypothesis testing and related factor-structure exploration.

Usage

```
Chinese_Herbal_Tea
```

Format

A data frame with 723 rows and 15 variables:

a1_price_reasonable Perceived price reasonableness of Chinese herbal tea products (Likert 5-point; 1 = strongly disagree, 5 = strongly agree).

a2_brand_awareness Perceived brand awareness/familiarity influencing evaluation or purchase intention (Likert 5-point).

a3_packaging_appeal Perceived attractiveness of packaging/appearance (Likert 5-point).

a4_taste_flavor Perceived taste and flavor satisfaction (Likert 5-point).

b1_nutrition_value Perceived nutritional value (Likert 5-point). Note: a rare value `-2` appears and is recommended to be recoded to `NA`.

b2_safety_assurance Perceived safety assurance (e.g., ingredient safety, quality control) (Likert 5-point).

b3_health_benefit Perceived health benefits (Likert 5-point).

c1_celebrity_endorsement Influence of celebrity endorsement on purchase/consumption intention (Likert 5-point).

c2_ip_collaboration Influence of IP/brand collaborations (co-branding) on purchase/consumption intention (Likert 5-point).

c3_tcm_institution_collab Influence of collaboration with traditional Chinese medicine (TCM) institutions/organizations on purchase/consumption intention (Likert 5-point).

c4_discount_promotion Influence of discounts and promotions on purchase/consumption intention (Likert 5-point).

q26_gov_support_increases_willingness Agreement that government support/policies would increase willingness to purchase/consume Chinese herbal tea (Likert 5-point).

q27_future_purchase_intent Future purchase/consumption intention (Likert 5-point).

q28_future_recommend_intent Future intention to recommend Chinese herbal tea to others (Likert 5-point).

q29_satisfaction Overall satisfaction with Chinese herbal tea products/experience (Likert 5-point).

Details

Most variables are measured on a 5-point Likert scale (coded as integers 1–5), where larger values indicate stronger agreement/more positive evaluation. One variable contains a rare special code (`-2`) that should be treated as missing/invalid in downstream analysis.

Source

Consumer survey dataset on Chinese herbal tea perceptions, marketing influences, and behavioral intentions.

CLX *Two-Sample Covariance Test by Cai, Liu and Xia (2013)*

Description

Given two sets of data matrices X and Y, where X is an $n_1 \times p$ matrix and Y is an $n_2 \times p$ matrix, this function conducts a hypothesis test for the equality of two covariance matrices. The null hypothesis is

$$H_0 : \Sigma_1 = \Sigma_2,$$

where Σ_1 and Σ_2 are the covariance matrices of X and Y, respectively. The test is based on the method proposed by Cai, Liu and Xia (2013). When the p-value is smaller than the significance level (usually 0.05), the null hypothesis is rejected.

Usage

```
CLX(X, Y, alpha = 0.05)
```

Arguments

X	A numeric matrix with n_1 rows and p columns.
Y	A numeric matrix with n_2 rows and p columns.
alpha	Significance level of the test.

Value

A list with the following components:

stat	The test statistic.
pval	The p-value of the test.
power	The empirical power of the test.
FDR	The false discovery rate.

Examples

```
p <- 500
n1 <- 100
n2 <- 150
X <- matrix(rnorm(n1 * p), ncol = p)
Y <- matrix(rnorm(n2 * p), ncol = p)
CLX(X, Y, alpha = 0.05)
```

cm13 *One-Sample Covariance Test by Cai and Ma (2013)*

Description

Given a data matrix, this function performs a one-sample test for the covariance matrix. The null hypothesis is

$$H_0 : \Sigma_n = \Sigma_0,$$

where Σ_n is the covariance matrix of the data and Σ_0 is a hypothesized covariance matrix. The test procedure is based on the method proposed by Cai and Ma (2013).

Usage

```
cm13(X, Sigma0, alpha)
```

Arguments

`X`	A numeric data matrix with n rows and p columns, where each row represents an observation.
`Sigma0`	A $p \times p$ hypothesized covariance matrix.
`alpha`	Significance level of the test.

Value

A named list with the following components:

statistic The test statistic.

threshold The rejection threshold for the test.

reject Logical; `TRUE` if the null hypothesis is rejected, and `FALSE` otherwise.

Examples

```
p <- 5
n <- 10
X <- matrix(rnorm(n * p), ncol = p)
alpha <- 0.05
Sigma0 <- diag(ncol(X))
cm13(X, Sigma0, alpha)
```

concrete — *Concrete Compressive Strength*

Description

Data about the compressive strength of concrete based on its ingredients and age.

Usage

```
concrete
```

Format

A data frame with component details and strength values.

FanPC_TFM — *Apply the FanPC method to the Truncated factor model*

Description

This function performs Factor Analysis via Principal Component (FanPC) on a given data set. It calculates the estimated factor loading matrix (AF), specific variance matrix (DF), and the mean squared errors.

Usage

```
FanPC_TFM(data, m, A, D, p)
```

Arguments

data	A matrix of input data.
m	The number of principal components.
A	The true factor loadings matrix.
D	The true uniquenesses matrix.
p	The number of variables.

Value

A list containing:

AF	Estimated factor loadings.
DF	Estimated uniquenesses.
MSESigmaA	Mean squared error for factor loadings.
MSESigmaD	Mean squared error for uniquenesses.
LSigmaA	Loss metric for factor loadings.
LSigmaD	Loss metric for uniquenesses.

Examples

```
library(SOPC)
library(MASS)
set.seed(123)
p <- 10
m <- 3
n <- 50
A <- matrix(rnorm(p * m), nrow = p, ncol = m)
D <- diag(runif(p, 0.2, 0.8))
F_mat <- matrix(rnorm(n * m), nrow = n, ncol = m)
E_mat <- MASS::mvrnorm(n, mu = rep(0, p), Sigma = D)
simulated_data <- F_mat %*% t(A) + E_mat
results <- FanPC_TFM(data = simulated_data, m = m, A = A, D = D, p = p)
print(results)
```

GOOG *Daily Google Stock Data*

Description

Daily OHLCV data for Google (ticker: GOOG) from 2018-01-01 to 2020-12-31, split-adjusted.

Usage

```
GOOG
```

Format

A data frame with 756 rows and 10 variables:

open, high, low, close Raw OHLC prices (USD)

adjOpen, adjHigh, adjLow, adjClose Split-adjusted OHLC prices (USD)

volume Raw trading volume (shares)

adjVolume Split-adjusted trading volume (shares)

Source

https://finance.yahoo.com/

LC *Two-Sample Covariance Test by Li and Chen (2012)*

Description

Given two sets of data matrices X and Y, where X is an n1 rows and p cols matrix and Y is an n2 rows and p cols matrix, we conduct hypothesis testing of the covariance matrix between two samples. The null hypothesis is:

Usage

```
LC(X, Y, delta_sigma = NULL, alpha = 0.05)
```

Arguments

X	A matrix of n1 by p.
Y	A matrix of n2 by p.
delta_sigma	A positive definite matrix.
alpha	Significance level.

Details

$$H_0 : \Sigma_1 = \Sigma_2$$

Σ_1 and Σ_2 are the sample covariance matrices of X and Y respectively. This test method is based on the test method proposed by Li and Chen (2012). When the pval value is less than the significance coefficient (generally 0.05), the null hypothesis is rejected.

Value

stat	a test statistic value.
pval	a test p_value.
power	a test power value.
FDR	a test FDR value.

Examples

```
p= 500;  n1 = 100; n2 = 150
X=matrix(rnorm(n1*p), ncol=p)
Y=matrix(rnorm(n2*p), ncol=p)
LC(X,Y)
```

new_energy_vehicle *New Energy Vehicle Purchase Intention Survey Data*

Description

A questionnaire survey on consumers' purchase intention toward new energy vehicles (NEVs) and its influencing factors. The dataset includes (i) household vehicle purchase history, (ii) attitudes toward policy/product/economic/firm factors measured on a 5-point Likert scale, and (iii) demographic information.

Usage

```
new_energy_vehicle
```

Format

A data frame with 520 rows and multiple variables:

household_ice_owned Whether the household has purchased an internal-combustion (fuel) vehicle (single choice).

household_nev_owned Whether the household has purchased a new energy vehicle (single choice).

policy_subsidy_intention Effect of subsidy policies (e.g., toll exemptions, lower purchase price, low-interest loans) on NEV purchase intention (Likert 5-point).

policy_license_intention Effect of license-plate policies (e.g., free registration, road-restriction privileges) on NEV purchase intention (Likert 5-point).

environmental_intention Effect of environmental concerns on NEV purchase intention (Likert 5-point).

infrastructure_intention Effect of charging infrastructure convenience on NEV purchase intention (Likert 5-point).

driving_experience_factor Effect of driving experience (product factor) on NEV purchase intention (Likert 5-point).

battery_performance_factor Effect of battery performance (range, lifespan, capacity, charging efficiency) on NEV purchase intention (Likert 5-point).

safety_factor Effect of safety and technology maturity/reliability on NEV purchase intention (Likert 5-point).

depreciation_cost_factor Effect of depreciation/durability concerns (economic factor) on NEV purchase intention (Likert 5-point).

purchase_cost_factor Effect of purchase price (economic factor) on NEV purchase intention (Likert 5-point).

charging_cost_factor Effect of charging cost (economic factor) on NEV purchase intention (Likert 5-point).

maintenance_cost_factor Effect of maintenance/repair cost (economic factor) on NEV purchase intention (Likert 5-point).

service_factor Effect of firm service (pre-sales and after-sales) on NEV purchase intention (Likert 5-point).

brand_factor Effect of brand (firm factor) on NEV purchase intention (Likert 5-point).

technology_advantage_factor Effect of perceived technological advantages (firm factor) on NEV purchase intention (Likert 5-point).

purchase_intent Stated intention to purchase an NEV (Likert 5-point).

recommend_intent Willingness to recommend NEVs to others (Likert 5-point).

repurchase_intent Willingness to prioritize buying an NEV next time (Likert 5-point).

gender Gender (single choice).

age Age group (single choice).

education Education level (single choice).

occupation Occupation (single choice).

hukou Household registration type (rural/urban; single choice).

household_income Average monthly household income (categorical; single choice).

Details

The Likert scale options are: A = Strongly disagree, B = Disagree, C = Neutral, D = Agree, E = Strongly agree.

Source

Consumer survey dataset on NEV purchase intention and influencing factors.

`protein` *Data Frame 'protein'*

Description

This is the Protein Data Set from the UCI Machine Learning Repository. It contains information about protein concentration in different samples.

Usage

```
protein
```

Format

A data frame with 45730 rows and 10 columns.

- `SampleID`: A unique identifier for each sample.
- `Protein1`: Concentration of Protein 1.
- `Protein2`: Concentration of Protein 2.
- `Protein3`: Concentration of Protein 3.

- `Protein4`: Concentration of Protein 4.
- `Protein5`: Concentration of Protein 5.
- `Protein6`: Concentration of Protein 6.
- `Protein7`: Concentration of Protein 7.
- `Protein8`: Concentration of Protein 8.
- `Protein9`: Concentration of Protein 9.
- `Protein10`: Concentration of Protein 10.

`real_estate_valuation` *Real Estate Valuation*

Description

Historical market data of real estate valuation.

Usage

```
real_estate_valuation
```

Format

A data frame with property attributes and price.

`review` *Travel Review Dataset*

Description

This dataset contains travel reviews from TripAdvisor.com, covering destinations in 10 categories across East Asia. Each traveler's rating is mapped to a scale from Terrible (0) to Excellent (4), and the average rating for each category per user is provided.

Usage

```
data(review)
```

Format

A data frame with multiple rows and 10 columns.

1 Unique identifier for each user (Categorical)

2 Average user feedback on art galleries

3 Average user feedback on dance clubs

4 Average user feedback on juice bars

5 Average user feedback on restaurants

6 Average user feedback on museums

7 Average user feedback on resorts

8 Average user feedback on parks and picnic spots

9 Average user feedback on beaches

10 Average user feedback on theaters

Details

The dataset is populated by crawling TripAdvisor.com and includes reviews on destinations in 10 categories across East Asia. Each traveler's rating is mapped as follows:

- Excellent (4)
- Very Good (3)
- Average (2)
- Poor (1)
- Terrible (0)

The average rating for each category per user is used.

Note

This dataset is licensed under a Creative Commons Attribution 4.0 International (CC BY 4.0) license.

Source

UCI Machine Learning Repository

Examples

```
# Load the dataset
data(review)

# Print the first few rows
head(review)

# Access specific columns (note the backticks for numeric names)
review$`1`  # User IDs
mean(review$`5`)  # Average rating for restaurants
```

riboflavin — *Riboflavin Production Data*

Description

This dataset contains measurements of riboflavin (vitamin B2) production by Bacillus subtilis, a Gram-positive bacterium commonly used in industrial fermentation processes. The dataset includes $n = 71$ observations with $p = 4088$ predictors, representing the logarithm of the expression levels of 4088 genes. The response variable is the log-transformed riboflavin production rate.

Usage

```
data(riboflavin)
```

Format

y Log-transformed riboflavin production rate (original name: `q_RIBFLV`). This is a continuous variable indicating the efficiency of riboflavin production by the bacterial strain.

x A matrix of dimension 71×4088 containing the logarithm of the expression levels of 4088 genes. Each column corresponds to a gene, and each row corresponds to an observation (experimental condition or time point).

Note

The dataset is provided by DSM Nutritional Products Ltd., a leading company in the field of nutritional ingredients. The data have been preprocessed and normalized to account for technical variations in the microarray measurements.

Examples

```
# Load the riboflavin dataset
data(riboflavin)

# Display the dimensions of the dataset
print(dim(riboflavin$x))
print(length(riboflavin$y))
```

riboflavinv100 — *Riboflavin Production Data (Top 100 Genes)*

Description

This dataset is a subset of the riboflavin production data by Bacillus subtilis, containing $n = 71$ observations. It includes the response variable (log-transformed riboflavin production rate) and the 100 genes with the largest empirical variances from the original dataset.

Usage

```
data(riboflavinv100)
```

Format

y Log-transformed riboflavin production rate (original name: `q_RIBFLV`). This is a continuous variable indicating the efficiency of riboflavin production by the bacterial strain.

x A matrix of dimension 71×100 containing the logarithm of the expression levels of the 100 genes with the largest empirical variances.

Note

The dataset is provided by DSM Nutritional Products Ltd., a leading company in the field of nutritional ingredients. The data have been preprocessed and normalized.

Examples

```
# Load the riboflavinv100 dataset
data(riboflavinv100)

# Display the dimensions of the dataset
print(dim(riboflavinv100$x))
print(length(riboflavinv100$y))
```

---

syk — *One Sample Covariance Test by Srivastava, Yanagihara, and Kubokawa (2014)*

Description

Given data, it performs 1-sample test for Covariance where the null hypothesis is

$$H_0 : \Sigma_n = \Sigma_0$$

where Σ_n is the covariance of data model and Σ_0 is a hypothesized covariance based on a procedure proposed by Srivastava, Yanagihara, and Kubokawa (2014).

Usage

```
syk(data, Sigma0, alpha)
```

Arguments

`data`	an $(n \times p)$ data matrix where each row is an observation.
`Sigma0`	a $(p \times p)$ given covariance matrix.
`alpha`	level of significance.

Value

a named list containing

statistic a test statistic value.

threshold rejection criterion to be compared against test statistic.

reject a logical; TRUE to reject null hypothesis, FALSE otherwise.

Examples

```
p = 5;n=10
data = matrix(rnorm(n*p), ncol=p)
alpha=0.05
Sigma0=diag(ncol(data))
syk(data, Sigma0, alpha)
```

---

taxi_trip_pricing *Taxi Trip Pricing*

Description

Data recording taxi trip details and pricing.

Usage

```
taxi_trip_pricing
```

Format

A data frame with trip duration, distance, and fare.

TFM *Truncated Factor Model Data Generator*

Description

The TFM function generates truncated factor model data using methods implemented in the **tmvtnorm** package. It currently supports truncated multivariate normal and truncated multivariate Student-t distributions.

Usage

```
TFM(n, mu, sigma, lower, upper, distribution_type, df = 4)
```

Arguments

`n`	Total number of observations.
`mu`	Mean vector of the distribution.
`sigma`	Covariance matrix of the distribution.
`lower`	Lower bound of the truncation interval.
`upper`	Upper bound of the truncation interval.
`distribution_type`	A character string specifying the distribution type. Possible values are `"truncated_normal"` and `"truncated_student"`.
`df`	Degrees of freedom for the truncated Student-t distribution. Only required when `distribution_type = "truncated_student"`.

Value

A matrix containing the generated truncated factor model data.

Examples

```
set.seed(123)
n <- 100
mu <- c(0, 1)
sigma <- matrix(c(1, 0.7, 0.7, 3), 2, 2)
lower <- c(-2, -3)
upper <- c(3, 3)
X_norm <- TFM(n, mu, sigma, lower, upper,
              distribution_type = "truncated_normal")
X_t <- TFM(n, mu, sigma, lower, upper,
           distribution_type = "truncated_student", df = 5)
```

`ttest.TFM` *T-test for Truncated Factor Model*

Description

This function performs a simple t-test for each variable in the dataset of a truncated factor model and calculates the False Discovery Rate (FDR) and power.

Usage

```
ttest.TFM(X, p, alpha = 0.05)
```

Arguments

`X`	A matrix or data frame of simulated or observed data from a truncated factor model.
`p`	The number of variables (columns) in the dataset.
`alpha`	The significance level for the t-test.

Value

A list containing:

- `FDR` The False Discovery Rate calculated from the rejected hypotheses.
- `Power` The power of the test, representing the proportion of true positives among the non-zero hypotheses.
- `pValues` A numeric vector of p-values obtained from the t-tests for each variable.
- `RejectedHypotheses` A logical vector indicating which hypotheses were rejected based on the specified significance level.

Examples

```
# Load necessary libraries
library(MASS)
library(mvtnorm)

set.seed(100)
# Set parameters for the simulation
p <- 400  # Number of features
n <- 120  # Number of samples
K <- 5    # Number of latent factors
true_non_zero <- 100  # Assume 100 features have non-zero means

# Simulate factor loadings matrix B (p x K)
B <- matrix(rnorm(p * K), nrow = p, ncol = K)

# Simulate factor scores (n x K)
FX <- MASS::mvrnorm(n, rep(0, K), diag(K))

# Simulate noise U (n x p), assuming Student's t-distribution with 3 degrees of freedom
U <- mvtnorm::rmvt(n, df = 3, sigma = diag(p))

# Create the data matrix X based on the truncated factor model
# Non-zero means for the first 100 features
mu <- c(rep(1, true_non_zero), rep(0, p - true_non_zero))
X <- rep(1, n) %*% t(mu) + FX %*% t(B) + U  # The observed data

# Apply the t-test function on the data
results <- ttest.TFM(X, p, alpha = 0.05)

# Print the results
print(results)
```

`winequality.white` *White Wine Quality*

Description

Physicochemical tests of white Portuguese "Vinho Verde" wine.

Usage

```
winequality.white
```

Format

A data frame with chemical properties and quality score.

---

yacht_hydrodynamics *Yacht Hydrodynamics*

Description

Data concerning the hydrodynamics of sailing yachts.

Usage

```
yacht_hydrodynamics
```

Format

A data frame with hull geometry and resistance.

Index

∗ **datasets**
- admission_predict, 2
- Chinese_Herbal_Tea, 2
- concrete, 6
- GOOG, 7
- new_energy_vehicle, 9
- real_estate_valuation, 11
- review, 11
- riboflavin, 13
- riboflavinv100, 13
- taxi_trip_pricing, 15
- winequality.white, 17
- yacht_hydrodynamics, 18

admission_predict, 2

Chinese_Herbal_Tea, 2
CLX, 4
cm13, 5
concrete, 6

FanPC_TFM, 6

GOOG, 7

LC, 8

new_energy_vehicle, 9

protein, 10

real_estate_valuation, 11
review, 11
riboflavin, 13
riboflavinv100, 13

syk, 14

taxi_trip_pricing, 15
TFM, 15
ttest.TFM, 16

winequality.white, 17

yacht_hydrodynamics, 18