

Package: circularNet (via r-universe)

July 14, 2026

Title Circular Graphical Model Estimation

Version 0.1.0

Description Provides methods for estimating circular graphical models using maximum likelihood estimation (MLE) and circular mean squared error (CMSE) approaches. The package includes tools for model fitting, network construction, network evaluation, and visualization. The CMSE-based methodology is related to Dar (2023) <<https://open.metu.edu.tr/handle/11511/102577>>. The package supports both simulated circular data and real-world applications, including gene-expression network analysis.

License MIT + file LICENSE

Encoding UTF-8

Imports graphics, igraph, stats

Suggests knitr, rmarkdown

VignetteBuilder knitr

Depends R (>= 4.1.0)

Config/roxygen2/version 8.0.0

NeedsCompilation no

Author Fatemeh Sarasir [aut, cre] (ORCID: <<https://orcid.org/0009-0003-5432-1352>>), Vilda Purutcuoglu [aut] (ORCID: <<https://orcid.org/0000-0002-3913-9005>>), Elif Dogan Dar [aut] (ORCID: <<https://orcid.org/0000-0001-7969-3767>>)

Maintainer Fatemeh Sarasir <fsarasir@gmail.com>

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

Date/Publication 2026-07-14 16:30:09 UTC

RemoteUrl <https://github.com/cran/circularNet>

RemoteRef HEAD

RemoteSha f6c888302bd85f29a8649925fd05a2bde252ef12

Contents

build_network	2
CMSE	3
evaluate_network	3
fit_circular_model	4
log_likelihood	5
mod_twopi	6
model_fit_cmse	6
plot_network	7
residual_fnc	8
Index	9

build_network	<i>Build Adjacency Matrix</i>
---------------	-------------------------------

Description

Constructs a binary network from estimated coefficients using thresholding.

Usage

```
build_network(beta_hat, threshold = 0.5)
```

Arguments

beta_hat	Coefficient matrix.
threshold	Threshold value.

Value

Binary adjacency matrix.

Examples

```
set.seed(1)

data <- matrix(
  runif(100, -pi, pi),
  ncol = 5
)

fit <- fit_circular_model(data)

build_network(
  fit,
  threshold = 0.2
)
```

CMSE	<i>Circular Mean Squared Error</i>
------	------------------------------------

Description

Computes the circular mean squared error.

Usage

```
CMSE(par, data)
```

Arguments

par	Parameter vector.
data	Data matrix (p x n).

Value

Numeric CMSE value.

Examples

```
data <- matrix(
  runif(20, -pi, pi),
  nrow = 4
)

par <- rep(0, nrow(data))

CMSE(par, data)
```

evaluate_network	<i>Evaluate Estimated Network</i>
------------------	-----------------------------------

Description

Computes performance metrics comparing an estimated network with a reference network.

Usage

```
evaluate_network(est, truth)
```

Arguments

est	Estimated adjacency matrix.
truth	Ground-truth adjacency matrix.

Value

List of evaluation metrics.

Examples

```
est <- matrix(
  c(0,1,0,
    1,0,1,
    0,1,0),
  nrow = 3,
  byrow = TRUE
)

truth <- matrix(
  c(0,1,0,
    1,0,0,
    0,0,0),
  nrow = 3,
  byrow = TRUE
)

evaluate_network(est, truth)
```

fit_circular_model	<i>Fit Circular Graphical Model</i>
--------------------	-------------------------------------

Description

Estimates a circular graphical model using node-wise regression and likelihood-based optimization.

Usage

```
fit_circular_model(data)
```

Arguments

data Matrix of circular observations (rows = observations, columns = variables).

Value

A coefficient matrix.

Examples

```

set.seed(1)

data <- matrix(
  runif(100, -pi, pi),
  ncol = 5
)

fit <- fit_circular_model(data)
round(fit, 2)

```

log_likelihood

*Log-Likelihood for Circular Model***Description**

Computes the negative log-likelihood used for parameter estimation in the likelihood-based circular graphical model.

Usage

```
log_likelihood(params, theta, phi, mu_theta, mu_phi)
```

Arguments

params	Parameter vector containing alpha, beta coefficients, and kappa.
theta	Response variable.
phi	Predictor matrix.
mu_theta	Mean of theta.
mu_phi	Mean of phi.

Value

Negative log-likelihood value.

Examples

```

theta <- runif(20, -pi, pi)
phi <- matrix(runif(40, -pi, pi), ncol = 2)
params <- c(0.1, 0.1, 0.1, 1)

log_likelihood(
  params,
  theta,
  phi,
  mean(theta),

```

```
colMeans(phi)
)
```

mod_twopi	<i>Wrap Angles to (-pi, pi)</i>
-----------	---------------------------------

Description

Transforms numeric values to the interval (-pi, pi).

Usage

```
mod_twopi(x)
```

Arguments

x Numeric vector of angles.

Value

Numeric vector of wrapped angles.

Examples

```
mod_twopi(c(-4 * pi, -3, 0, 3, 4 * pi))
```

model_fit_cmse	<i>Fit Circular Model Using CMSE</i>
----------------	--------------------------------------

Description

Fits a circular regression model using CMSE optimization.

Usage

```
model_fit_cmse(data)
```

Arguments

data Data matrix (p x n), where rows correspond to variables and columns correspond to samples.

Value

A list containing:

- beta: coefficient matrix
- code: optimization convergence codes

Examples

```
set.seed(1)

data <- matrix(
  runif(40, -pi, pi),
  nrow = 4
)

model_fit_cmse(data)
```

plot_network	<i>Plot Estimated Network</i>
--------------	-------------------------------

Description

Visualizes an adjacency matrix as a network graph.

Usage

```
plot_network(adj_matrix)
```

Arguments

adj_matrix Adjacency matrix.

Value

A network plot.

Examples

```
adj <- matrix(
  c(0,1,1,
    1,0,0,
    1,0,0),
  nrow = 3,
  byrow = TRUE
)

plot_network(adj)
```

residual_fnc	<i>Compute Residuals</i>
--------------	--------------------------

Description

Computes predicted and observed circular values.

Usage

```
residual_fnc(par, data)
```

Arguments

par	Parameter vector.
data	Data matrix (p x n).

Value

List with predicted and observed values.

Examples

```
data <- matrix(
  runif(20, -pi, pi),
  nrow = 4
)

par <- rep(0, nrow(data))

residual_fnc(par, data)
```

Index

`build_network`, [2](#)

CMSE, [3](#)

`evaluate_network`, [3](#)

`fit_circular_model`, [4](#)

`log_likelihood`, [5](#)

`mod_twopi`, [6](#)

`model_fit_cmse`, [6](#)

`plot_network`, [7](#)

`residual_fnc`, [8](#)