

Package ‘DREGAR’

May 7, 2026

Type Package

Title Regularized Estimation of Dynamic Linear Regression in the Presence of Autocorrelated Residuals (DREGAR)

Version 0.1.4.0

Date 2025-05-17

Depends R(>= 2.10.0)

Imports msgps

Maintainer Hamed Haselimashhadi <hamedhaseli@gmail.com>

Description A penalized/non-penalized implementation for dynamic regression in the presence of autocorrelated residuals (DREGAR) using iterative penalized/ordinary least squares. It applies Mallows CP, AIC, BIC and GCV to select the tuning parameters.

License GPL (>= 2)

LazyLoad no

Repository CRAN

NeedsCompilation no

Date/Publication 2025-05-17 11:40:02 UTC

Author Hamed Haselimashhadi [aut, cre]

Contents

| | |
|--------------|----------|
| dregar2 | 2 |
| dregar6 | 3 |
| generateAR | 4 |
| sim.dregar | 5 |
| Index | 6 |

| | |
|---------|---|
| dregar2 | <i>Estimating (just) adaptive-DREGAR coefficients using an iterative 2-step procedure</i> |
|---------|---|

Description

Estimating coefficients for penalized/non-penalized dynamic regression in the presence of autocorrelated residuals using an iterative 2-step procedure.

Usage

```
dregar2(data, da = 0, ar = 0, mselection = 4,
        normalize = FALSE, penalized = TRUE,
        iteration = 15)
```

Arguments

| | |
|------------|---|
| data | Data matrix of order (time, response, covariates) |
| da | A vector of lags. Autoregressive orders for response. For example 1:p for all lags from 1 to p |
| ar | A vector of lags. Autoregressive orders for residuals. For example 1:q for all lags from 1 to q |
| mselection | Model selection criteria. Choosing among 1 (CP), 2 (AIC), 3 (GCV) and 4 (BIC) |
| normalize | Logical flag. Setting to TRUE to normalize data prior to analysis |
| penalized | Logical flag. Setting to TRUE to estimate coefficients through penalized likelihood. Otherwise the algorithm applies iterative OLS. |
| iteration | The number of iterations |

Author(s)

Hamed Haselimashhadi <hamedhaseli@gmail.com>

See Also

[dregar6](#), [generateAR](#), [sim.dregar](#)

Examples

```
par(mfrow=c(2,2))
m=sim.dregar(n=500 , beta=1:4, phi=generateAR(2), theta=.3,
            n.z.coeffs=3 , plot=TRUE) # generating data
r=dregar2(data = m$rawdata,da = 1:3,ar = 1:2,mselection = 4,
          penalized = 1 )# estimating parameters using 2-step adaptive-DREGAR
round(cbind(
  true      = c(phi=c(m$phi,0),theta=c(m$theta,0),beta=m$beta),
  estimates = c(phi=r$phi,theta=r$theta,beta=r$beta)
```

```

)
,3
)
plot(r$obj)
acf(r$res, main='Residual ACF')
pacf(r$res,main='Residual PACF')

```

| | |
|---------|---|
| dregar6 | <i>Estimating adaptive/non-adaptive DREGAR coefficients using an iterative 6-step procedure</i> |
|---------|---|

Description

Estimating coefficients for penalized dynamic regression in the presence of autocorrelated residuals using an iterative 6-step procedure.

Usage

```
dregar6(data , da, ar, mselection = 4, type = "alasso",
         normalize = FALSE, iteration = 15, intercept=FALSE)
```

Arguments

| | |
|------------|--|
| data | Data matrix of order (time, response, covariates) |
| da | A vector of lags. Autoregressive orders for the response. For example 1:p for all lags from 1 to p |
| ar | A vector of lags. Autoregressive orders for residuals. For example 1:q for all lags from 1 to q |
| mselection | Model selection criteria. Choosing among 1 (CP), 2 (AIC), 3 (GCV) and 4 (BIC) |
| type | Type of penalty. Choosing between 'enet' and 'alasso' for DREGAR and adaptive-DREGAR penalties. |
| normalize | Logical flag. Setting to TRUE to normalise data prior to analysis |
| iteration | The number of iterations |
| intercept | Logical flag. Setting to TRUE to have intercept in the model. |

Author(s)

Hamed Haselimashhadi <hamedhaseli@gmail.com>

See Also

[dregar2](#), [generateAR](#), [sim.dregar](#)

Examples

```

par(mfrow=c(2,2))
m=sim.dregar(n=500 , beta=1:4, phi=generateAR(2), theta=.1,
            n.z.coefs=3 , plot=TRUE) # generating data
r=dregar6(data=m$rawdata, da = 1:3,
          ar = 1:2,mselection = 4,
          type='alasso')# estimating parameters using (non-apdative) DREGAR
round(cbind(
  true      = c(phi=c(m$phi,0),theta=c(m$theta,0),beta=m$beta),
  estimates = c(phi=r$phi,theta=r$theta,beta=r$beta)
),3
)
plot(r$mod.phi,main='phi')
plot(r$mod.theta,main='theta')
plot(r$mod.beta,main='beta')

```

generateAR

*Generating stationary autoregressive coefficients***Description**

Generating stationary autoregressive coefficients

Usage

```

generateAR(n = 1, l = -1, u = 1 ,
          min.distance = .Machine$double.eps ,
          sort.coeff = FALSE)

```

Arguments

| | |
|--------------|--|
| n | The number of coefficients |
| l | Lower bound for coefficients |
| u | Upper bound for coefficients |
| min.distance | Minimum distance amongst all pair of coefficients |
| sort.coeff | Logical flag. If TRUE, then resulting coefficients are sorted descending |

Author(s)

Hamed Haselimashhadi <hamedhaseli@gmail.com>

See Also

[dregar2](#) , [dregar6](#) , [sim.dregar](#)

Examples

```

generateAR(3 , -1 , 1 , .01 )

```

sim.dregar

*Simulating data from DREGAR model***Description**

Simulating a mean zero Gaussian lagged response regression in the presence of autocorrelated residuals

Usage

```
sim.dregar(n = 500 , beta = 1, ind = FALSE ,
           phi = .3, theta = .5, var = 1 , n.z.coeffs=0,
           shuffle = FALSE , plot = FALSE )
```

Arguments

| | |
|------------|--|
| n | The number of data points to be simulated |
| beta | Regression coefficients |
| ind | Logical flag. If TRUE then observations are assumed to be independent. Otherwise they are generated from random AR(1) processes. In both cases, variables are assumed to be mutually independent and follow Gaussian distribution. |
| phi | Dynamic coefficient(s) |
| theta | Residuals coefficient(s) |
| var | Variance of the error term |
| n.z.coeffs | Number of zero coefficients if needed |
| shuffle | Logical flag. If TRUE shuffle coefficients. Otherwise data are grouped corresponded to non-zero and zero coefficients. |
| plot | Logical flag. Plot response |

Author(s)

Hamed Haselimashhadi <hamedhaseli@gmail.com>

See Also

[dregar2](#) , [dregar6](#) , [generateAR](#)

Examples

```
simdata=sim.dregar(n = 100 , beta = 1,
                  ind = TRUE , phi = .40 , theta = -.25,
                  var = 1 , plot = TRUE)
str(simdata)
```

Index

dregar2, [2](#), [3–5](#)

dregar6, [2](#), [3](#), [4](#), [5](#)

generateAR, [2](#), [3](#), [4](#), [5](#)

sim.dregar, [2–4](#), [5](#)