Package 'GGMncv'

July 12, 2020

Type Package

Title Gaussian Graphical Models with Non-Convex Penalties

Version 1.1.0

Date 2020-07-03

Description Estimate Gaussian graphical models with non-convex penalties, including
the atan Wang and Zhu (2016) <doi:10.1155/2016/6495417>,
seamless L0 Dicker, Huang, and Lin (2013) <doi:10.5705/ss.2011.074>,
exponential Wang, Fan, and Zhu <doi:10.1007/s10463-016-0588-3>,
smooth integration of counting and absolute deviation Lv and Fan (2009) <doi:10.1214/09-AOS683>,
logarithm Mazumder, Friedman, and Hastie (2011) <doi:10.1198/jasa.2011.tm09738>,
Lq, smoothly clipped absolute deviation Fan and Li (2001) <doi:10.1198/016214501753382273>,
and minimax concave penalty Zhang (2010) <doi:10.1214/09-AOS729>. There are also extensions
for computing variable inclusion probabilities, multiple regression coefficients, and
statistical inference <doi:10.1214/15-EJS1031>.

License GPL-2

Depends R (>= 4.0.0)

Imports Rcpp (>= 1.0.4.6), Rdpack (>= 0.11-1), reshape, ggplot2 (>= 3.3.0), glassoFast (>= 1.0), numDeriv (>= 2016.8-1.1), MASS (>= 7.3-51.5), methods, psych (>= 1.9.12.31), stats, utils

Suggests qgraph (>= 1.6.5)

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

LinkingTo Rcpp, RcppArmadillo

RdMacros Rdpack

BugReports https://github.com/donaldRwilliams/GGMncv/issues

NeedsCompilation yes

Author Donald Williams [aut, cre]

Maintainer Donald Williams <drwwilliams@ucdavis.edu>

Repository CRAN

Date/Publication 2020-07-11 22:00:18 UTC

R topics documented:

`coef.ggmncv` *Regression Coefficients from* `ggmncv` *Objects*

Description

Regression Coefficients from `ggmncv` Objects

Usage

```
## S3 method for class 'ggmncv'
coef(object, ...)
```

Arguments

`object`	An Object of classs `ggmncv`
`...`	Currently ignored

Value

A matrix of regression coefficients

Note

The matrix of coefficients can be accessed by removing the class from the returned object (e.g., `unclass(coefs)`).

Examples

```
# data
Y <- GGMncv::ptsd

# correlations
S <- cor(Y)

# fit model
fit <- GGMncv(S, n = nrow(Y))

coefs <- coef(fit)
```

`constrained` *Constrained Precision Matrix*

Description

Compute the maximum likelihood estimate, given certain elements are constrained to zero (e.g., an adjacency matrix). This approach is described in Hastie et al. (2015).

Usage

```
constrained(Sigma, adj)
```

Arguments

`Sigma`	Covariance matrix
`adj`	Matrix with contraints. A zero indicates that element should be constrained to zero.

Value

A list containing the inverse covariance matrix and the covariance matrix.

Note

The algorithm is written in `c++`.

Examples

```
# data
Y <- GGMncv::ptsd[,1:5]

# columns
p <- ncol(Y)

# contstraint matrix
constraints <- matrix(0,p,p)

# set one value to zero
constraints[2,3] <- 1
constraints[3,2] <-1

# estimate, given constraints
fit <- constrained(cor(Y), adj = constraints)
Theta <- fit$Theta
```

---

desparsify *De-sparsified Graphical Lasso Estimator*

Description

Compute the desparsified glasso estimator with the approach described in Equation 7 of Jankova and Van De Geer (2015).

Usage

```
desparsify(object, ...)
```

Arguments

`object`	An object of class ggmncv
`...`	Currently ignored

Value

The de-sparsified estimates, including

- `Theta` De-sparsified precision matrix
- `P` De-sparsified partial correlation matrix

Note

This assumes the Gaussian data.

References

Jankova J, Van De Geer S (2015). "Confidence intervals for high-dimensional inverse covariance estimation." *Electronic Journal of Statistics*, **9**(1), 1205–1229.

Examples

```
# data
Y <- GGMncv::ptsd

# fit model
fit <- GGMncv(cor(Y), n = nrow(Y))

desparsify(fit)
```

GGMncv *GGMncv*

Description

Estimate Gaussian graphical models with non-convex penalties.

Usage

```
GGMncv(
  x,
  n,
  penalty = "atan",
  ic = "bic",
  lambda = NULL,
  n_lambda = 50,
  gamma = NULL,
  select = FALSE,
  L0_learn = FALSE,
  refit = FALSE,
  LLA = TRUE,
  initial = "sicm",
  method = "pearson",
  progress = TRUE,
  store = TRUE,
  vip = FALSE,
  vip_iter = 1000,
  ...
)
```

Arguments

`x`	There are 2 options: either a `n` by `p` data matrix or a `p` by `p` correlation matrix.
`n`	Numeric. Sample size.
`penalty`	Character string. Which penalty should be used (defaults to `atan`).
`ic`	Character string. Which information criterion should be used (defaults to `bic`), give `select = TRUE`. The options include `aic`, `ebic` (ebic_gamma defaults to `0.5`; see details), `ric`, or any generalized information criterion provided in section 5 of Kim et al. (2012). The options are `gic_1` (i.e., `bic`) to `gic_7`.
`lambda`	Numeric. Tuning parameter governing the degrees of penalization. Defaults to `NULL` which results in fixing lambda to `sqrt(log(p)/n)`.
`n_lambda`	Numeric. The number of regularization/thresholding parameters (defaults to `100`).
`gamma`	Numeric. Hyperparameter for the penalty function. Defaults to `0.1` which is the recommended value for the default penalty (see details).
`select`	Logical. Should lambda be selected with BIC (defaults to `FALSE`)?
`L0_learn`	Logical. Should lambda be selected based on the non-regularized precision matrix (defaults to `FALSE`; see details).
`refit`	Logical. Should the precision matrix be refitted, given the adjacency matrix (defaults to `FALSE`)? When set to `TRUE`, this provides the **non-regularized**, maximum likelihood estimate with constraints.
`LLA`	Logical. Should the local linear approximation be used for maximizing the penalized likelihood ? The default is `TRUE` (see details). Setting to `FALSE` results in the so-called one-step approach.
`initial`	Character string. Which initial values should be used for the one-step approach (i.e., `LLA = FALSE`) ? Default is the sample inverse covariance matrix (`sicm`). Options include `sicm` and `lw` (Ledoit and Wolf shrinkage estimator; Ledoit and Wolf 2004).
`method`	Character string. Which correlation coefficient should be computed. One of "pearson" (default), "spearman", or "polychoric".
`progress`	Logical. Should a progress bar be included (defaults to `TRUE`) ? Note that this only applies when `select = TRUE`.
`store`	Logical. Should all of the fitted models be saved (defaults to `NULL`). Note this only applies when `select = TRUE`. and ignored otherwise (the one model is saved.)
`vip`	Logical. Should variable inclusion "probabilities" be computed (defaults to `FALSE`)?
`vip_iter`	Numeric. How many bootstrap sample for computing `vip` (defaults to 1000) ? Note also that only the default lambda is implemented (select is not implemented).
`...`	Currently ignored.

Details

Several of the penalties are (continuous) approximations to the L0 penalty, that is, best subsets model selection. However, the solution does not require enumerating all possible models which results in a computationally efficient algorithm.

L0 approximations:

- Atan: `penalty = "atan"` (Wang and Zhu 2016). This is currently the default.
- Seamless L0: `penalty = "selo"` (Dicker et al. 2013).
- Exponential: `penalty = "exp"` (Wang et al. 2018)
- Log: `penalty = "log"` (Mazumder et al. 2011).
- Sica: `penalty = "sica"` (Lv and Fan 2009)

Additional penalties:

- SCAD: `penalty = "scad"` (Fan and Li 2001).
- MCP: `penalty = "mcp"` (Zhang 2010).
- Adaptive lasso `penalty = "adapt"` (Zou 2006)
- Lasso `penalty = "lasso"` (Tibshirani 1996)

Gamma

The code gamma argument corresponds to additional hyperparameter for each penalty. The defaults are set to the recommended values from the respective papers.

L0_learn

`L0_learn` is perhaps a misnomer, in that best subsets solution is not computed. This option corresponds to the following steps, assuming `select = TRUE`:

1. Estimte the graph for a given lambda value
2. Refit the precision matrix, given the contraints from step 1. This results in the maximum likelihood estimate (non-regularized).
3. Compute BIC for the refitted graph from step 2.
4. After repeating steps 1-3 for lambda value, select the graph according to BIC.

Note that this is most useful in datasets that have more nodes than variables (i.e., low-dimensional).

LLA

The local linear approximate is for non-covex penalties is described in (Fan et al. 2009). This is essentially a weighted (g)lasso. Note that by default `LLA = TRUE`. This can be set to `FALSE` when n is much larger than p (e.g., this can improve power). This is due to the work of (Zou and Li 2008), which suggested that, so long as the starting values are good, then it is possible to use a one-step estimator. In the case of low-dimensional data, the sample based inverse covariance matrix is used to compute the lambda matrix. This is expected to work well, assuming that n is sufficiently larger than p. For high-dimensional data, the initial values for obtaining the lambda matrix are obtained from glasso.

Model Selection

It is common to select lambda. However, in more recent approaches (see references above), lambda is fixed to `sqrt(log(p)/n)`. This has the advantage of being tuning free and this value is expected to provide competitive performance. It is possible to select lambda by setting `select = TRUE`.

EBIC

When setting `ic = "ebic"`, the additional parameter that determines the additional penalty to BIC is passed via the `...` argument. This must be specificed as `ebic_gamma = 1`, with the default set to `0.5`.

Value

An object of class `ggmncv`, including:

- `Theta` Inverse covariance matrix
- `Sigma` Covariance matrix
- `P` Weighted adjacency matrix
- `adj` Adjacency matrix
- `lambda` Tuning parameter (i.e., sqrt(log(p)/n))
- `fit` glasso fitted model (a list)

References

Dicker L, Huang B, Lin X (2013). "Variable selection and estimation with the seamless-L 0 penalty." *Statistica Sinica*, 929–962.

Fan J, Feng Y, Wu Y (2009). "Network exploration via the adaptive LASSO and SCAD penalties." *The annals of applied statistics*, **3**(2), 521.

Fan J, Li R (2001). "Variable selection via nonconcave penalized likelihood and its oracle properties." *Journal of the American statistical Association*, **96**(456), 1348–1360.

Kim Y, Kwon S, Choi H (2012). "Consistent model selection criteria on high dimensions." *The Journal of Machine Learning Research*, **13**, 1037–1057.

Ledoit O, Wolf M (2004). "A well-conditioned estimator for large-dimensional covariance matrices." *Journal of Multivariate Analysis*, **88**(2), 365–411.

Lv J, Fan Y (2009). "A unified approach to model selection and sparse recovery using regularized least squares." *The Annals of Statistics*, **37**(6A), 3498–3528.

Mazumder R, Friedman JH, Hastie T (2011). "Sparsenet: Coordinate descent with nonconvex penalties." *Journal of the American Statistical Association*, **106**(495), 1125–1138.

Tibshirani R (1996). "Regression shrinkage and selection via the lasso." *Journal of the Royal Statistical Society: Series B (Methodological)*, **58**(1), 267–288.

Wang Y, Fan Q, Zhu L (2018). "Variable selection and estimation using a continuous approximation to the L0 penalty." *Annals of the Institute of Statistical Mathematics*, **70**(1), 191–214.

Wang Y, Zhu L (2016). “Variable selection and parameter estimation with the Atan regularization method.” *Journal of Probability and Statistics*.

Zhang C (2010). “Nearly unbiased variable selection under minimax concave penalty.” *The Annals of statistics*, **38**(2), 894–942.

Zou H (2006). “The adaptive lasso and its oracle properties.” *Journal of the American statistical association*, **101**(476), 1418–1429.

Zou H, Li R (2008). “One-step sparse estimates in nonconcave penalized likelihood models.” *Annals of statistics*, **36**(4), 1509.

Examples

```
# data
Y <- GGMncv::ptsd[,1:10]

S <- cor(Y)

# fit model
fit <- GGMncv(S, n = nrow(Y))

# plot
qgraph::qgraph(fit$P)
```

`ggm_compare` *Compare Gaussian Graphical Models*

Description

Compare Gaussian graphical models with the de-sparsified estimator of (Jankova and Van De Geer 2015).

Usage

```
ggm_compare(object_1, object_2, method = "fdr", alpha = 0.05, ...)
```

Arguments

`object_1`	An object of class ggmncv
`object_2`	An object of class ggmncv
`method`	Character string. A correction method for multiple comparison (defaults to `fdr`). Can be abbreviated. See p.adjust.
`alpha`	Numeric. Significance level (defaults to `0.05`).
`...`	Currently ignored.

- `P_diff` De-sparsified partial correlation matrix differences

- adj Adjacency matrix based on the p-values.
- uncorrected Uncorrected p-values
- corrected Corected p-values
- method The approach used for multiple comparisons
- alpha Significance level

Examples

```
# data
Y1 <- MASS::mvrnorm(250, rep(0, 10), Sigma = diag(10))
Y2 <- MASS::mvrnorm(250, rep(0, 10), Sigma = diag(10))

# fit models
fit1 <- GGMncv(Y1, n = nrow(Y1))
fit2 <- GGMncv(Y2, n = nrow(Y2))

# compare
compare_ggms <- ggm_compare(fit1, fit2)
```

---

inference *Statistical Inference for Gaussian Graphical Models*

Description

Compute p-values for each relation based on the de-sparsified precision matrix (Jankova and Van De Geer 2015).

Usage

```
inference(object, method = "fdr", alpha = 0.05, ...)
```

Arguments

object	An object of class ggmncv
method	Character string. A correction method for multiple comparison (defaults to fdr). Can be abbreviated. See p.adjust.
alpha	Numeric. Significance level (defaults to 0.05).
...	Currently ignored.

Value

- Theta De-sparsified precision matrix
- adj Adjacency matrix based on the p-values.
- uncorrected Uncorrected p-values
- corrected Corected p-values
- method The approach used for multiple comparisons
- alpha Significance level

Note

This assumes the Gaussian data.

References

Jankova J, Van De Geer S (2015). “Confidence intervals for high-dimensional inverse covariance estimation.” *Electronic Journal of Statistics*, **9**(1), 1205–1229.

Examples

```
# data
Y <- GGMncv::ptsd

# fit model
fit <- GGMncv(cor(Y), n = nrow(Y))


# statistical inference
inference(fit)
```

---

plot.ggmncv *Plot* ggmncv *Objects*

Description

Plot the solution path for the partial correlations, the information criterion solution path, or the variable inclusion ’probabilities’.

Usage

```
## S3 method for class 'ggmncv'
plot(x, type = "pcor_path", size = 1, alpha = 0.5, color = "black", ...)
```

Arguments

x	An object of class ggmncv
type	Character string. Which type should be plotted ? Options included pcor_path, ic_path, or vip.
size	Numeric. The size of the points (vip) or lines (pcor_path or ic_path) The default is 1.
alpha	Numeric. The transparency of the lines. Only for the solution path options.
color	Character string. The color of the points (vip) or lines (pcor_path or ic_path). The default is black.
...	Currently ignored.

Value

A ggplot object

Examples

```
# data
Y <- GGMncv::ptsd[,1:10]

# correlations
S <- cor(Y, method = "spearman")

# fit model
fit <- GGMncv(x = S, n = nrow(Y),
              penalty = "atan",
              vip = TRUE,
              vip_iter = 50)

# plot
plot(fit, size = 4, type = "vip")
```

---

predict.ggmncv   *Predict method for* ggmncv *Objects*

Description

Predicted values based on a ggmncv object

Usage

```
## S3 method for class 'ggmncv'
predict(object, newdata = NULL, ...)
```

Arguments

object	An object of class ggmncv
newdata	An optional data frame in which to look for variables with which to predict. If omitted, the fitted values are used.
...	Currently ignored

Value

A matrix of predicted values

Examples

```
# data
Y <- scale(Sachs)

# test data
Ytest <- Y[1:100,]

# training data
Ytrain <- Y[101:nrow(Y),]

fit <- GGMncv(Ytrain, n = nrow(Ytrain))

pred <- predict(fit,
                newdata = Ytest)

round(apply((pred - Ytest)^2, 2, mean), 2)
```

`print.ggmncv` *Print* ggmncv *Objects*

Description

Print ggmncv Objects

Usage

```
## S3 method for class 'ggmncv'
print(x, ...)
```

Arguments

`x`	An object of class ggmncv
`...`	Currently ignored

`ptsd` *Data: Post-Traumatic Stress Disorder*

Description

A dataset containing items that measure Post-traumatic stress disorder symptoms (Armour et al. 2017). There are 20 variables (p) and 221 observations (n).

Usage

```
data("ptsd")
```

Format

A dataframe with 221 rows and 20 variables

Details

- Intrusive Thoughts
- Nightmares
- Flashbacks
- Emotional cue reactivity
- Psychological cue reactivity
- Avoidance of thoughts
- Avoidance of reminders
- Trauma-related amnesia
- Negative beliefs
- Negative trauma-related emotions
- Loss of interest
- Detachment
- Restricted affect
- Irritability/anger
- Self-destructive/reckless behavior
- Hypervigilance
- Exaggerated startle response
- Difficulty concentrating
- Sleep disturbance

References

Armour C, Fried EI, Deserno MK, Tsai J, Pietrzak RH (2017). "A network analysis of DSM-5 posttraumatic stress disorder symptoms and correlates in US military veterans." *Journal of anxiety disorders*, **45**, 49–59.

Sachs — *Data: Sachs Network*

Description

Protein expression in human immune system cells

Usage

```
data("Sachs")
```

Format

A data frame containing 7466 cells (n = 7466) and flow cytometry measurements of 11 (p = 11) phosphorylated proteins and phospholipids (Sachs et al. 2002)

References

Sachs K, Gifford D, Jaakkola T, Sorger P, Lauffenburger DA (2002). “Bayesian network approach to cell signaling pathway modeling.” *Science’s STKE*, **2002**(148), pe38–pe38.

Examples

```
data("Sachs")
```

Index