

# Package: CBPE (via r-universe)

July 3, 2024

**Type** Package

**Title** Correlation-Based Penalized Estimators

**Version** 0.1.0

**Maintainer** Mina Norouzirad <mina.norouzirad@gmail.com>

**Description** Provides correlation-based penalty estimators for both linear and logistic regression models by implementing a new regularization method that incorporates correlation structures within the data. This method encourages a grouping effect where strongly correlated predictors tend to be in or out of the model together. See Tutz and Ulbricht (2009) <doi:10.1007/s11222-008-9088-5> and Algamal and Lee (2015) <doi:10.1016/j.eswa.2015.08.016>.

**License** GPL (>= 2)

**URL** <https://github.com/mnrzrad/CBPE>

**Encoding** UTF-8

**RoxygenNote** 7.3.2

**Depends** R (>= 3.5)

**Imports** stats

**NeedsCompilation** no

**Author** Mohammad Arashi [ctb]  
(<<https://orcid.org/0000-0002-5881-9241>>), Mahdi Rahimi [ctb],  
Mina Norouzirad [aut, cre, cph]  
(<<https://orcid.org/0000-0003-0311-6888>>), FCT, I.P. [fnd]  
(under the scope of the projects UIDB/00297/2020 and  
UIDP/00297/2020 (NovaMath))

**Repository** CRAN

**Date/Publication** 2024-07-02 15:30:02 UTC

## Contents

CBPLinearE . . . . .	2
CBPLogisticE . . . . .	3

**Description**

This function computes the correlation-based estimator for linear regression models.

**Usage**

```
CBPLinearE(X, y, lambda)
```

**Arguments**

X	A numeric matrix of predictors where rows represent observations and columns represent variables.
y	A numeric vector of response variables.
lambda	A regularization parameter.

**Details**

The correlation-based penalized linear estimator is calculated as:

$$\hat{\beta} = \operatorname{argmin} \left\{ \sum_{i=1}^n (y_i - \mathbf{x}_i^\top \boldsymbol{\beta})^2 + \lambda \sum_{i=1}^{p-1} \sum_{j>i} \left( \frac{(\beta_i - \beta_j)^2}{1 - \rho_{ij}} + \frac{(\beta_i + \beta_j)^2}{1 + \rho_{ij}} \right) \right\}$$

where  $\rho_{ij}$  denotes the (empirical) correlation between the  $i$ th and the  $j$ th predictor.

**Value**

A numeric vector of the estimated coefficients for the specified model.

**References**

Tutz, G., Ulbricht, J. (2009). Penalized regression with correlation-based penalty. *Stat Comput* 19, 239–253.

**Examples**

```
set.seed(42)
n <- 100
p <- 4
X <- matrix(rnorm(n * p), n, p)
beta_true <- c(0.5, -1, 2, 5)
y <- X %*% beta_true + rnorm(n)
lambda <- 0.1
```

```
result <- CBPLinearE(X, y, lambda = lambda)
print(result)
```

---

CBPLogisticE

*Correlation-based Estimator for Logistic Regression Models*


---

## Description

This function computes the correlation-based estimator for logistic regression models.

## Usage

```
CBPLogisticE(X, y, lambda, max_iter = 100, tol = 1e-06)
```

## Arguments

X	A numeric matrix of predictors where rows represent observations and columns represent variables.
y	A numeric vector of binary outcomes (0 or 1).
lambda	A regularization parameter.
max_iter	An integer specifying the maximum number of iterations for the logistic regression algorithm. Default is 100.
tol	A numeric value specifying the convergence tolerance for the logistic regression algorithm. Default is 1e-10.

## Details

The correlation-based penalized logistic estimator is calculated as:

$$\hat{\beta} = \operatorname{argmin} \left\{ \sum_{i=1}^n (y_i \ln(\pi_i) + (1 - y_i) \ln(1 - \pi_i)) + \lambda \sum_{i=1}^{p-1} \sum_{j>i} \left( \frac{(\beta_i - \beta_j)^2}{1 - \rho_{ij}} + \frac{(\beta_i + \beta_j)^2}{1 + \rho_{ij}} \right) \right\}$$

where  $\pi_i = \Pr(y_i = 1 | \mathbf{x}_i)$  and  $\rho_{ij}$  denotes the (empirical) correlation between the  $i$ th and the  $j$ th predictor.

## Value

A numeric vector of the estimated coefficients for the specified model.

## References

Algamil, Z. Y., & Lee, M. H. (2015). Penalized logistic regression with the adaptive LASSO for gene selection in high-dimensional cancer classification. *Expert Systems with Applications*, 42(23), 9326-9332.

**Examples**

```
set.seed(42)
n <- 100
p <- 4
X <- matrix(rnorm(n * p), n, p)
beta_true <- c(0.5, -1, 2, 5)
y <- rbinom(n, 1, 1 / (1 + exp(-X %*% beta_true)))
lambda <- 0.1

result <- CBPLogisticE(X, y, lambda)
print(result)
```

# Index

CBPLinearE, 2  
CBPLogisticE, 3