Package: dpcc (via r-universe)

November 5, 2024

Type Package

Title Dynamic Programming for Convex Clustering

Version 1.0.0

Author Bingyuan Zhang, Jie Chen, Yoshikazu Terada

Maintainer Bingyuan Zhang <zhang@sigmath.es.osaka-u.ac.jp>

Description Use dynamic programming method to solve l1 convex clustering with identical weights.

License MIT + file LICENSE

Encoding UTF-8

LazyData False

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.1

LinkingTo Rcpp

Imports Rcpp

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Repository https://bingyuan-zhang.r-universe.dev

RemoteUrl https://github.com/bingyuan-zhang/dpcc

RemoteRef HEAD

RemoteSha 02c5bfa1f578dfffaf3737ced8a9dedf9457f6c9

Contents

cdp *L1 convex clustering with a single lambda.*

Description

L1 convex clustering with a single lambda.

Usage

```
cdp(X, lam)
```

Arguments

X	a data matrix of n * p or a data vector with length n.
lam	a tuning parameter.

Details

A list with length p equal to the dimension of the data matrix. Each dimension includes a vector of the estimated centroids.

Value

the estimated centroids.

Examples

```
# generate a data matrix with n = 10 and p = 2.
X = matrix(rnorm(10*2), 10, 2)
lam = find_lambda(X)/2
# set a tuning parameter lambda.
cdp(X, lam)
```

cpaint *L1 convex clustering with a lambda sequence.*

Description

L1 convex clustering with a lambda sequence.

Usage

```
cpaint(X, lam)
```

Arguments

X	a data matrix of n * p or a data vector with length n.
lam	a sequence of lambdas.

Details

A list with length p equal to the dimension of the data matrix. Each dimension includes a sequence of vectors. Each vector includes the estimated centroids with a certain tuning parameter lambda.

Value

A sequence of estimated centroids.

Examples

```
# generate a data matrix with n = 10 and p = 2.
X = matrix(rnorm(10*2), 10, 2)
# set the biggest lambda in the sequence.
lam_max = find_lambda(X)
# set the length of the sequence.
K = 10
# equally separate the sequence with K.
Lam = sapply(1:K, function(i) i/K*lam_max)
cpaint(X,Lam)
```

---

find_lambda *Return the lambda which causes all the points become fused into one cluster.*

Description

Return the lambda which causes all the points become fused into one cluster.

Usage

```
find_lambda(X)
```

Arguments

X	data matrix of n * p

Value

the biggest lambda

Examples

```
X = matrix(rnorm(3*2), 3, 2)
find_lambda(X)
```

Index