Package ‘dbMC’

July 20, 2021

Title  Confidence Interval for Matrix Completion via De-Biased Estimator

Version  1.0.0


Imports  softImpute
License  GPL-2
Encoding  UTF-8
RoxygenNote  7.1.1
Suggests  rmarkdown, knitr
VignetteBuilder  knitr
NeedsCompilation  no

Author  The Tien Mai [aut, cre]
Maintainer  The Tien Mai <t.t.mai@medisin.uio.no>
Repository  CRAN
Date/Publication  2021-07-20 07:20:08 UTC

R topics documented:

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>CI_mc</td>
<td>2</td>
</tr>
<tr>
<td>dbmc</td>
<td>3</td>
</tr>
<tr>
<td>P_Omega</td>
<td>4</td>
</tr>
<tr>
<td>Index</td>
<td>5</td>
</tr>
</tbody>
</table>
CI_mc  

*compute the confidence intervals (CIs) from the de-biased estimator*

**Description**

This function returns a CI for an entries of interest with a significant level alpha.

**Usage**

CI_mc(i, j, alpha = 0.05, missfrac, X.db, est_rank, sigma2 = 1)

**Arguments**

- **i**: the row index of the interest entry X_{i,j}
- **j**: the row index of the interest entry X_{i,j}
- **alpha**: confidence level, default is 0.05
- **missfrac**: the missing proportion in the underlying matrix. It is the total of missing entries over total entries.
- **X.db**: the de-biased estimated matrix from the 'dbmc' function.
- **est_rank**: the (estimated) low-rank of the underlying matrix or the rank of the de-biased estimator.
- **sigma2**: the noise-variance level.

**Value**

CI confidence interval.

(i,j) the location of the entry at i-th row and j-th column.

v.ij the estimated variance of the limiting Gaussian distribution.

**References**

dbmc

de-biased estimator

Description

de-biased low-rank matrix completion estimator

This function compute a de-biased estimator from a rank-\(r\) matrix completion using the algorithms from the package "softImpute".

Usage

dbmc(x, ximp, entries_miss, est_rank)

Arguments

x
    the initial matrix with missing entries
ximp
    an imputed matrix, output from the package "softImpute".
entries_miss
    the missing indices
est_rank
    the rank of the matrix x, or the estimated rank from the package "softImpute".

Value

x.db the de-baised estimation matrix.

References


Examples

# simulated data
require(softImpute)
n = 100
p = 100
J = 2 # the true low-rank
np = n*p
sig2 = 1
missfrac = 0.5
# xtrue is the underlying matrix that we do not know and want to recover it
xtrue = matrix(rnorm(n*J),n,J)%*%matrix(rnorm(J*p),J,p)
# generating missing entries locations
imiss = sample(np,np*missfrac,replace=FALSE)
# xna is the observed matrix with missing entries
xna = xtrue + matrix(rnorm(np, sd = sig2),nr = n,nc = p)
xna[imiss] = NA
lamda = 2.5*sig2*sqrt(n*p)
# note that we only have xna as our initial data
# first, fit a softImpute method
fit1 = softImpute(xna, type = 'als')
# complete the matrix by a softImpute method
ximp = complete(xna, fit1)
mean((ximp - xtrue)^2); rankMatrix(ximp, .1)[1]
# now, de-biased the softImpute method
x.db = dbmc(x = xna,
    ximp = ximp,
    entries_miss = imiss,
    est_rank = 2)
mean((x.db - xtrue)^2); rankMatrix(x.db, .1)[1]

---

**P_Omega**

*projection onto observation set*

**Description**

This function returns a matrix where the missing entries are replaced by 0s.

**Usage**

P_Omega(a, entri)

**Arguments**

- **a**
  - a matrix.
- **entri**
  - missing entries location.

**Value**

Return a matrix whose its missing entries are replaced by 0s.
Index

CI_mc, 2

dbmc, 3

P_Omega, 4