Package: COAP (via r-universe)

October 28, 2026

Type Package

Title High-Dimensional Covariate-Augmented Overdispersed Poisson Factor Model

Version 1.3

Date 2025-03-27

Author Wei Liu [aut, cre], Qingzhi Zhong [aut]

Maintainer Wei Liu <liuweideng@gmail.com>

Description A covariate-augmented overdispersed Poisson factor model is proposed to jointly perform a high-dimensional Poisson factor analysis and estimate a large coefficient matrix for overdispersed count data. More details can be referred to Liu et al. (2024) <doi:10.1093/biomtc/ujae031>.

License GPL-3

Depends irlba, R (>= 3.5.0)

Imports MASS, stats, Rcpp (>= 1.0.10)

URL https://github.com/feiyoung/COAP

BugReports https://github.com/feiyoung/COAP/issues

Suggests knitr, rmarkdown

LinkingTo Rcpp, RcppArmadillo

VignetteBuilder knitr

Encoding UTF-8

RoxygenNote 7.1.2

NeedsCompilation yes

Repository https://cran.r-universe.dev

Date/Publication 2025-03-27 11:30:02 UTC

RemoteUrl https://github.com/cran/COAP

RemoteRef HEAD

RemoteSha 86729fb3981d2652fcde2c258cf292cc618cf4c3

Contents

gendata_simu 2
RR_COAP 3
selectParams 5

Index **7**

`gendata_simu` *Generate simulated data*

Description

Generate simulated data from covariate-augmented Poisson factor models

Usage

```
gendata_simu(
  seed = 1,
  n = 300,
  p = 50,
  d = 20,
  q = 6,
  rank0 = 3,
  rho = c(1.5, 1),
  sigma2_eps = 0.1,
  seed.beta = 1
)
```

Arguments

`seed`	a postive integer, the random seed for reproducibility of data generation process.
`n`	a postive integer, specify the sample size.
`p`	a postive integer, specify the dimension of count variables.
`d`	a postive integer, specify the dimension of covariate matrix.
`q`	a postive integer, specify the number of factors.
`rank0`	a postive integer, specify the rank of the coefficient matrix.
`rho`	a numeric vector with length 2 and positive elements, specify the signal strength of regression coefficient and loading matrix, respectively.
`sigma2_eps`	a positive real, the variance of overdispersion error.
`seed.beta`	a postive integer, the random seed for reproducibility of data generation process by fixing the regression coefficient matrix beta.

Details

None

Value

return a list including the following components: (1) X, the high-dimensional count matrix; (2) Z, the high-dimensional covriate matrix; (3) bbeta0, the low-rank large coefficient matrix; (4) B0, the loading matrix; (5) H0, the factor matrix; (6) rank: the true rank of bbeta0; (7) q: the true number of factors.

References

None

See Also

RR_COAP

Examples

```
n <- 300; p <- 100
d <- 20; q <- 6; r <- 3
datlist <- gendata_simu(n=n, p=p, d=20, q=q, rank0=r)
str(datlist)
```

RR_COAP *Fit the COAP model*

Description

Fit the covariate-augmented overdispersed Poisson factor model

Usage

```
RR_COAP(
  X_count,
  multiFac = rep(1, nrow(X_count)),
  Z = matrix(1, nrow(X_count), 1),
  rank_use = 5,
  q = 15,
  epsELBO = 1e-05,
  maxIter = 30,
  verbose = TRUE,
  joint_opt_beta = FALSE,
  fast_svd = TRUE
)
```

Arguments

X_count	a count matrix, the observed count matrix.
multiFac	an optional vector, the normalization factor for each unit; default as full-one vector.
Z	an optional matrix, the covariate matrix; default as a full-one column vector if there is no additional covariates.
rank_use	an optional integer, specify the rank of the regression coefficient matrix; default as 5.
q	an optional string, specify the number of factors; default as 15.
epsELBO	an optional positive vlaue, tolerance of relative variation rate of the envidence lower bound value, defualt as '1e-5'.
maxIter	the maximum iteration of the VEM algorithm. The default is 30.
verbose	a logical value, whether output the information in iteration.
joint_opt_beta	a logical value, whether use the joint optimization method to update bbeta. The default is FALSE, which means using the separate optimization method.
fast_svd	a logical value, whether use the fast SVD algorithm in the update of bbeta; default is TRUE.

Details

None

Value

return a list including the following components: (1) H, the predicted factor matrix; (2) B, the estimated loading matrix; (3) bbeta, the estimated low-rank large coefficient matrix; (4) invLambda, the inverse of the estimated variances of error; (5) H0, the factor matrix; (6) ELBO: the ELBO value when algorithm stops; (7) ELBO_seq: the sequence of ELBO values.

References

Liu, W. and Q. Zhong (2024). High-dimensional covariate-augmented overdispersed poisson factor model. arXiv preprint arXiv:2402.15071.

See Also

None

Examples

```
n <- 300; p <- 100
d <- 20; q <- 6; r <- 3
datlist <- gendata_simu(n=n, p=p, d=20, q=q, rank0=r)
str(datlist)
fitlist <- RR_COAP(X_count=datlist$X, Z = datlist$Z, q=6, rank_use=3)
str(fitlist)
```

`selectParams` *Select the parameters in COAP models*

Description

Select the number of factors and the rank of coefficient matrix in the covariate-augmented overdispersed Poisson factor model

Usage

```
selectParams(
  X_count,
  Z,
  multiFac = rep(1, nrow(X_count)),
  q_max = 15,
  r_max = 24,
  threshold = c(0.1, 0.01),
  verbose = TRUE,
  ...
)
```

Arguments

`X_count`	a count matrix, the observed count matrix.
`Z`	an optional matrix, the covariate matrix; default as a full-one column vector if there is no additional covariates.
`multiFac`	an optional vector, the normalization factor for each unit; default as full-one vector.
`q_max`	an optional string, specify the upper bound for the number of factors; default as 15.
`r_max`	an optional integer, specify the upper bound for the rank of the regression coefficient matrix; default as 24.
`threshold`	an optional 2-dimensional positive vector, specify the the thresholds that filters the singular values of beta and B, respectively.
`verbose`	a logical value, whether output the information in iteration.
`...`	other arguments passed to the function `RR_COAP`.

Details

The threshold is to filter the singular values with low signal, to assist the identification of underlying model structure.

Value

return a named vector with names ‘hr‘ and ‘hq‘, the estimated rank and number of factors.

References

None

See Also

RR_COAP

Examples

```
n <- 300; p <- 100
d <- 20; q <- 6; r <- 3
datlist <- gendata_simu(seed=30, n=n, p=p, d=20, q=q, rank0=r)
str(datlist)
set.seed(1)
para_vec <- selectParams(X_count=datlist$X, Z = datlist$Z)
print(para_vec)
```

Index

gendata_simu, 2

RR_COAP, 3, *3*, *5, 6*

selectParams, 5