

# Package ‘SpaCOAP’

January 20, 2025

**Type** Package

**Title** High-Dimensional Spatial Covariate-Augmented Overdispersed  
Poisson Factor Model

**Version** 1.2

**Date** 2024-05-26

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

**Maintainer** Wei Liu <liuwei8@scu.edu.cn>

**Description** A spatial covariate-augmented overdispersed Poisson factor model is proposed to perform efficient latent representation learning method for high-dimensional large-scale spatial count data with additional covariates.

**License** GPL-3

**URL** <https://github.com/feiyong/SpaCOAP>

**BugReports** <https://github.com/feiyong/SpaCOAP/issues>

**Imports** LaplacesDemon,stats, methods, Matrix, MASS, Rcpp (>= 1.0.10)

**Depends** irlba, R (>= 3.5.0)

**Suggests** knitr, rmarkdown

**LinkingTo** Rcpp, RcppArmadillo

**VignetteBuilder** knitr

**Encoding** UTF-8

**RoxygenNote** 7.1.2

**NeedsCompilation** yes

**Repository** CRAN

**Date/Publication** 2024-05-27 14:50:02 UTC

## Contents

chooseParams . . . . .	2
gendata_spacoap . . . . .	3
SpaCOAP . . . . .	5

---

chooseParams	<i>Select the parameters in COAP models</i>
--------------	---

---

**Description**

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

**Usage**

```
chooseParams(
  X_count,
  Adj_sp,
  H,
  Z = matrix(1, nrow(X_count), 1),
  offset = rep(0, 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 with shape n-by-p.
Adj_sp	a sparse matrix, the weighted adjacency matrix;
H	a n-by-d matrix, the covariate matrix with low-rank regression coefficient matrix;
Z	an optional matrix, the fixed-dimensional covariate matrix with control variables; default as a full-one column vector if there is no additional covariates.
offset	an optional vector, the offset for each unit; default as full-zero 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 <a href="#">SpaCOAP</a> .

**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**

[SpaCOAP](#)

**Examples**

```
width <- 20; height <- 15; p <- 300
d <- 20; k <- 3; q <- 6; r <- 3
datlist <- gendata_spacoap(width=width, height=height, p=p, d=d, k=k, q=q, rank0=r)
set.seed(1)
para_vec <- chooseParams(X_count=datlist$X, Adj_sp=datlist$Adj_sp,
  H= datlist$H, Z = datlist$Z, r_max=6)
print(para_vec)
```

---

gendata_spacoap	<i>Generate simulated data</i>
-----------------	--------------------------------

---

**Description**

Generate simulated data from spaital covariate-augmented Poisson factor models

**Usage**

```
gendata_spacoap(
  seed = 1,
  width = 20,
  height = 30,
  p = 500,
  d = 40,
  k = 3,
  q = 5,
  rank0 = 3,
  eta0 = 0.5,
  bandwidth = 1,
  rho = c(10, 1),
  sigma2_eps = 1,
  seed.beta = 1
)
```

**Arguments**

seed	a positive integer, the random seed for reproducibility of data generation process.
width	a positive integer, specify the width of the spatial grid.
height	a positive integer, specify the height of the spatial grid.
p	a positive integer, specify the dimension of count variables.
d	a positive integer, specify the dimension of covariate matrix with low-rank regression coefficient matrix.
k	a positive integer, specify the dimension of covariate matrix as control variables.
q	a positive integer, specify the number of factors.
rank0	a positive integer, specify the rank of the coefficient matrix.
eta0	a real between 0 and 1, specify the spatial autocorrelation parameter.
bandwidth	a real positive value, specify the bandwidth in calculating the weighted adjacency matrix.
rho	a numeric vector with length 2 and positive elements, specify the signal strength of loading matrix and regression coefficient, respectively.
sigma2_eps	a positive real, the variance of overdispersion error.
seed.beta	a positive 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:

- X - the high-dimensional count matrix;
- Z - the low-dimensional covariate matrix with control variables.
- H - the high-dimensional covariate matrix;
- Adj\_sp - the weighted adjacency matrix;
- alpha0 - the regression coefficient matrix corresponding to Z;
- bbeta0 - the low-rank large regression coefficient matrix corresponding to H;
- B0 - the loading matrix;
- F0 - the latent factor matrix;
- rank0 - the true rank of bbeta0;
- q - the true number of factors;
- eta0 - spatial autocorrelation parameter;
- pos - spatial coordinates for each observation.

**References**

None

**See Also**[SpaCOAP](#)**Examples**

```
width <- 20; height <- 15; p <- 100
d <- 20; k <- 3; q <- 6; r <- 3
datlist <- gendata_spacoap(width=width, height=height, p=p, d=20, k=k, q=q, rank0=r)
str(datlist)
```

SpaCOAP

*Fit the SpaCOAP model***Description**

Fit the spatial covariate-augmented overdispersed Poisson factor model

**Usage**

```
SpaCOAP(
  X_count,
  Adj_sp,
  H,
  Z = matrix(1, nrow(X_count), 1),
  offset = rep(0, nrow(X_count)),
  rank_use = 5,
  q = 15,
  epsELB0 = 1e-08,
  maxIter = 30,
  verbose = TRUE,
  add_IC_inter = FALSE,
  seed = 1,
  algo = 1
)
```

**Arguments**

<code>X_count</code>	a count matrix, the observed count matrix with shape n-by-p.
<code>Adj_sp</code>	a sparse matrix, the weighted adjacency matrix;
<code>H</code>	a n-by-d matrix, the covariate matrix with low-rank regression coefficient matrix;
<code>Z</code>	an optional matrix, the fixed-dimensional covariate matrix with control variables; default as a full-one column vector if there is no additional covariates.
<code>offset</code>	an optional vector, the offset for each unit; default as full-zero vector.
<code>rank_use</code>	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 evidence lower bound value, default as '1e-8'.
maxIter	the maximum iteration of the VEM algorithm. The default is 30.
verbose	a logical value, whether output the information in iteration.
add_IC_inter	a logical value, add the identifiability condition in iterative algorithm or add it after algorithm converges; default as FALSE.
seed	an integer, set the random seed in initialization, default as 1;
algo	an optional integer taking value 1 Or 2, select the algorithm used, default as 1, representing variational EM algorithm.

### Details

None

### Value

return a list including the following components:

- F - the predicted factor matrix;
- B - the estimated loading matrix;
- bbeta - the estimated low-rank large coefficient matrix;
- alpha0 - the estimated regression coefficient matrix corresponding to Z;
- invLambda - the inverse of the estimated variances of error;
- eta - the estimated spatial autocorrelation parameter;
- S - the approximated posterior covariance for each row of F;
- ELBO - the ELBO value when algorithm stops;
- ELBO\_seq - the sequence of ELBO values.
- time\_use - the running time in model fitting of SpaCOAP;

### References

Liu W, Zhong Q. High-dimensional covariate-augmented overdispersed poisson factor model. *Bio-metrics*. 2024 Jun;80(2):ujae031.

### See Also

None

### Examples

```
width <- 20; height <- 15; p <- 100
d <- 20; k <- 3; q <- 6; r <- 3
datlist <- gendata_spacoap(width=width, height=height, p=p, d=20, k=k, q=q, rank0=r)
fitlist <- SpaCOAP(X_count=datlist$X, Adj_sp = datlist$Adj_sp,
H= datlist$H, Z = datlist$Z, q=6, rank_use=3)
str(fitlist)
```

# Index

[chooseParams](#), [2](#)

[gendata\\_spacoap](#), [3](#)

[SpaCOAP](#), [2](#), [3](#), [5](#), [5](#)