Package 'LUCIDus'

July 22, 2020

Type Package

Title Latent Unknown Clustering with Integrated Data

Version 2.1.0

Author Yinqi Zhao, David V. Conti, Cheng Peng, Zhao Yang

Maintainer Yinqi Zhao <yinqiz@usc.edu>

Description An implementation for the 'LUCID' model (Peng (2019) <doi:10.1093/bioinformatics/btz667>) to jointly estimate latent unknown clusters/subgroups with integrated data.
An EM algorithm is used to obtain the latent cluster assignment and model parameter estimates.
Feature selection is achieved by applying the L1 regularization method.

Depends R (>= 3.6.0)

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

URL https://github.com/Yinqi93/LUCIDus

Suggests knitr, rmarkdown

VignetteBuilder knitr

Imports mclust, nnet, networkD3, parallel, boot, lbfgs, glasso, glmnet

NeedsCompilation no

Repository CRAN

Date/Publication 2020-07-22 08:00:03 UTC

R topics documented:

`boot.lucid` *Bootstrap method of inference for LUCID*

Description

This function provides SEs of parameter estimates from a LUCID model through bootstrap method.

Usage

```
boot.lucid(G, Z, Y, CoG = NULL, CoY = NULL, model, R = 100, n = detectCores())
```

Arguments

`G`	Genetic features/environmental exposures, a `matrix`.
`Z`	Biomarkers/other omics data, a `matrix`.
`Y`	Disease outcome, it is suggested to transform it into a n by 1 `matrix`.
`CoG`	Optional, matrix. Covariates to be adjusted for estimating the latent cluster.
`CoY`	Optional, matrix. Covariates to be adjusted for estimating the outcome.
`model`	A LUCID model fitted by `est.lucid`.
`R`	Number of bootstrap iterations.
`n`	Number of CPU cores to be used in the bootstrap

Value

A list of estimates with their 95 percent CI.

Author(s)

Yinqi Zhao, Cheng Peng, Zhao Yang, David V. Conti

References

Cheng Peng, Jun Wang, Isaac Asante, Stan Louie, Ran Jin, Lida Chatzi, Graham Casey, Duncan C Thomas, David V Conti, A Latent Unknown Clustering Integrating Multi-Omics Data (LUCID) with Phenotypic Traits, Bioinformatics, , btz667, https://doi.org/10.1093/bioinformatics/btz667.

Examples

```
## Not run:
fit1 <- est.lucid(G = sim2[, 1:10], Z = sim2[, 11:20], Y = as.matrix(sim2[, 21]),
K = 2, family = "binary")
chk <- Sys.getenv("_R_CHECK_LIMIT_CORES_", "")
if (nzchar(chk) && chk == "TRUE") {
 # use 2 cores in CRAN/Travis/AppVeyor
 num_workers <- 2L
} else {
 num_workers <- parallel::detectCores()
}
boot1 <- boot.lucid(G = sim2[, 1:10], Z = sim2[, 11:20], Y = as.matrix(sim2[, 21]),
 model = fit1, R = 100, n = num_workers)

## End(Not run)
```

---

def.control *Control parameters for EM algorithm*

Description

Control parameters for EM algorithm

Usage

```
def.control(tol = 0.001, max_itr = 1000, max_tot.itr = 10000)
```

Arguments

tol	Convergence criteria for the EM algorithm. Default is 0.001.
max_itr	Maximum number of iterations in each try of fitting process, integer, default is 1000.
max_tot.itr	Maximum number of total iterations, integer, default is 10000.

Value

A list of tolerance settings for LUCID.

Author(s)

Yinqi Zhao, Cheng Peng, Zhao Yang, David V. Conti

`def.tune` *Define tuning parameters of regularization for LUCID model.*

Description

Define tuning parameters of regularization for LUCID model.

Usage

```
def.tune(
  Rho_G = 0,
  Rho_Z_InvCov = 0,
  Rho_Z_CovMu = 0,
  Select_G = FALSE,
  Select_Z = FALSE
)
```

Arguments

`Rho_G`	Numeric. Penalty for selection on genetic features/environmental exposures.
`Rho_Z_InvCov`	Numeric. Penalty for the inverse of the covariance of biomarkers, which will produce a sparse matrix.
`Rho_Z_CovMu`	Numeric. Penalty for the product of the inverse of the covariance of biomarkers, which will produce a sparse matrix for the mean.
`Select_G`	Flag for variable selection in genetic features/environmental exposures. Default is FALSE.
`Select_Z`	Flag for variable selection in biomarkers. Default is FALSE.

Value

A list of tuning parameters and settings will be returned for integrative clustering.

Author(s)

Yinqi Zhao, Cheng Peng, Zhao Yang, David V. Conti

`est.lucid` *Estimate latent unknown clusters with multi-omics data*

Description

This function estimates the latent clusters by integrating genetic features/environmental exposures, biomarkers with/without the outcome of interest. Variable selection is available for analyzing the high-dimensional data.

Usage

```
est.lucid(
  G,
  Z,
  Y,
  CoG = NULL,
  CoY = NULL,
  K = 2,
  family = "normal",
  useY = TRUE,
  control = def.control(),
  tune = def.tune(),
  Z.var.str = NULL
)
```

Arguments

`G`	Genetic features/environmental exposures, a `matrix`.
`Z`	Biomarkers/other omics data, a `matrix`.
`Y`	Disease outcome, it is suggested to transform it into a n by 1 `matrix`.
`CoG`	Optional, matrix. Covariates to be adjusted for estimating the latent cluster.
`CoY`	Optional, matrix. Covariates to be adjusted for estimating the outcome.
`K`	Number of latent clusters.
`family`	Type of outcome Y. It should be choose from "normal", "binary".
`useY`	Whether or not to include the information of Y to estimate the latent clusters. Default is TRUE.
`control`	A list of tolerance parameters used by EM algorithm. See `def.control`.
`tune`	A list of tuning parameters used by variable selection procedure. See `def.tune`
`Z.var.str`	The variance-covariance structure for the biomarkers. See `mclustModelNames` for details.

Value

A list which contains the several features of LUCID, including:

`pars`	Estimates of parameters of LUCID, including beta (estimates of genetic feature/environmental exposure), mu (estimates of cluster-specific biomarker means), sigma (estimates of the cluster-specific biomarker variance-covariance matrix) and gamma(estimates of cluster-specific effect and covariates effect related to the outcome)
`K`	Number of latent cluster
`Z.var.str`	The model used to estimate the cluster-specific variance-covariance matrix, for further details, see `mclust`
`likelihood`	The log likelihood of the LUCID model
`post.p`	Predicted probability of belonging to each latent cluster

Author(s)

Yinqi Zhao, Cheng Peng, Zhao Yang, David V. Conti

References

Cheng Peng, Jun Wang, Isaac Asante, Stan Louie, Ran Jin, Lida Chatzi, Graham Casey, Duncan C Thomas, David V Conti, A Latent Unknown Clustering Integrating Multi-Omics Data (LUCID) with Phenotypic Traits, Bioinformatics, , btz667, https://doi.org/10.1093/bioinformatics/btz667.

Examples

```
## Not run:
set.seed(10)
fit1 <- est.lucid(G = sim1[, 1:10],Z = sim1[, 11:20],Y = as,matrix(sim1[, 21]),
K = 2, family = "binary")
fit2 <- est.lucid(G = sim1[, 1:10],Z = sim1[, 11:20],Y = as,matrix(sim1[, 21]),
K = 2, family = "binary",
tune = def.tune(Select_Z = TRUE, Rho_Z_InvCov = 0.1, Rho_Z_CovMu = 90,
Select_G = TRUE, Rho_G = 0.02))

## End(Not run)
```

`plot.lucid` *Visualize the LUCID model through a Sankey diagram This function generates a Sankey diagram for the results of integrative clustering based on an* `lucid` *object*

Description

Visualize the LUCID model through a Sankey diagram This function generates a Sankey diagram for the results of integrative clustering based on an `lucid` object

Usage

```
## S3 method for class 'lucid'
plot(x, ...)
```

Arguments

x	A model fitted by est.lucid
...	Other parameters to be passed to plot

Value

A DAG graph created by sankeyNetwork

Author(s)

Cheng Peng, Zhao Yang, David V. Conti

References

Cheng Peng, Jun Wang, Isaac Asante, Stan Louie, Ran Jin, Lida Chatzi, Graham Casey, Duncan C Thomas, David V Conti, A Latent Unknown Clustering Integrating Multi-Omics Data (LUCID) with Phenotypic Traits, Bioinformatics, , btz667, https://doi.org/10.1093/bioinformatics/btz667.

Examples

```
## Not run:
fit1 <- est.lucid(G = G1, Z = Z1, Y = Y1, CoY = CovY, K = 2, family = "binary")
plot(fit1)

## End(Not run)
```

---

predict.lucid *Predict the outcome based on a fitted LUCID model*

Description

Predict the outcome based on a fitted LUCID model

Usage

```
## S3 method for class 'lucid'
predict(object, newG, newZ, newCoG = NULL, newCoY = NULL, response = TRUE, ...)
```

Arguments

object	A model fitted and returned by est.lucid
newG	A new data set of genetic/environmental factors
newZ	A new data set of biomarkers
newCoG	Optional. A new data set of covariates included in the G->X analysis
newCoY	Optional. A new data set of covariates included in the X->Y analysis
response	Report the posterior distribution of cluster assignment (and the probability of binary outcome), default is TRUE
...	Other parameters to be passed to predict

Value

A list contains predicted latent cluster and outcome for each observation

Examples

```
## Not run:
index <- sample(1:2000, 200)
fit <- est.lucid(G = sim1[-index, 1:10], Z = sim1[-index, 11:20], Y = as.matrix(sim1[-index, 21]))
pred <- predict(object = fit, newG = sim1[index, 1:10], newZ = sim1[index, 11:20])

## End(Not run)
```

---

print.lucid *Print the output of* est.lucid

Description

Print the output of est.lucid

Usage

```
## S3 method for class 'lucid'
print(x, ...)
```

Arguments

x	An object of LUCID model, returned by est.lucid
...	Other arguments to be passed to print

print.sumlucid *Print the output of LUCID in a nicer table*

Description

Print the output of LUCID in a nicer table

Usage

```
## S3 method for class 'sumlucid'
print(x, ...)
```

Arguments

x	An object returned by summary.lucid
...	Other parameters to be passed to print

sim1 *simulated dataset 1*

Description

A simulated dataset for integrated clustering with normal outcome. The data is simulated under cluster number K = 2.

Usage

```
sim1
```

Format

A matrix of 22 columns, which are

G1 - G10 Genetic features, G1 to G5 are causal genes contributed to clustering, with OR = 2; G6 to G10 are null genes that is not related to clustering

Z1 - Z10 Biomarkers, Z1 to Z5 are causal biomarkers with delta Z = 4 between 2 clusters, Z6 to Z10 are noises with delta Z = 0. All biomarkers are assumed to be independent with each other

Y Outcome of interest, which follows 2 normal distribution with N(-1, 1) and N(1, 1)

X Latent cluster assignment for each observation

sim2 *simulated dataset 2*

Description

A simulated dataset for integrated clustering with binary outcome. The data is simulated under cluster number K = 2.

Usage

```
sim2
```

Format

A matrix of 22 columns, which are

G1 - G10 Genetic features, G1 to G5 are causal genes contributed to clustering, with OR = 2; G6 to G10 are null genes that is not related to clustering

Z1 - Z10 Biomarkers, Z1 to Z5 are causal biomarkers with delta Z = 4 between 2 clusters, Z6 to Z10 are noises with delta Z = 0. All biomarkers are assumed to be independent with each other

Y Outcome of interest, the odds ratio of the cluster is 2

X Latent cluster assignment for each observation

summary.lucid *Summarize the results of LUCID model*

Description

Summarize the results of LUCID model

Usage

```
## S3 method for class 'lucid'
summary(object, boot.se = NULL, ...)
```

Arguments

object	A model fitted by est.lucid
boot.se	A object returned by boot.lucid, which contains the bootstrap standard error
...	Other parameters to be passed to summary

Value

A list with class "sumlucid", which contains the following object

`Beta`	Estimates of genetic/environmental effects (and effect of covariates if included), matrix
`Mu`	Estimates of cluster-specific biomarker means, matrix
`Gamma`	Estimates of cluster-specific disease risk (and effect of covariates if included), vector
`Family`	Type of Y, binary or normal
`K`	Number of latent clusters
`loglik`	log likelihood of the model
`BIC`	Bayesian Information Criteria of the model
`boot.se`	Bootstrap SE for estimates, an object returned by `boot.lucid`

Author(s)

Yinqi Zhao, Cheng Peng, Zhao Yang, David V. Conti

References

Cheng Peng, Jun Wang, Isaac Asante, Stan Louie, Ran Jin, Lida Chatzi, Graham Casey, Duncan C Thomas, David V Conti, A Latent Unknown Clustering Integrating Multi-Omics Data (LUCID) with Phenotypic Traits, Bioinformatics, , btz667, https://doi.org/10.1093/bioinformatics/btz667.

Examples

```
## Not run:
fit1 <- est.lucid(G = G1, Z = Z1, Y = Y1, CoY = CovY, K = 2, family = "binary", useY = FALSE)
summary(fit1)
fit2 <- est.lucid(G = G1, Z = Z1, Y = Y1, CoY = CovY, K = 2, family = "binary", useY = FALSE,
 tune = def.tune(Select_Z = TRUE, Rho_Z_InvCov = 0.1, Rho_Z_CovMu = 90,
 Select_G = TRUE, Rho_G = 0.02))
summary(fit2)

## End(Not run)
```

`tune.lucid` *Grid search for tuning parameters to fit the LUCID model*

Description

Grid search for tuning parameters to fit the LUCID model

Usage

```
tune.lucid(
  G,
  Z,
  Y,
  CoG = NULL,
  CoY = NULL,
  family = "normal",
  useY = TRUE,
  K = 2:6,
  Rho_G = NULL,
  Rho_Z_InvCov = NULL,
  Rho_Z_CovMu = NULL
)
```

Arguments

G	Genetic features/environmental exposures, a matrix.
Z	Biomarkers/other omics data, a matrix.
Y	Disease outcome, it is suggested to transform it into a n by 1 matrix.
CoG	Optional, matrix. Covariates to be adjusted for estimating the latent cluster.
CoY	Optional, matrix. Covariates to be adjusted for estimating the outcome.
family	Type of outcome Y. It should be choose from "normal", "binary".
useY	Whether or not to include the information of Y to estimate the latent clusters. Default is TRUE.
K	Numeric sequence. Number of latent clusters.
Rho_G	Numeric sequence, Lasso type penalty for selection of G.
Rho_Z_InvCov	Numeric sequence, Lasso type penalty for the inverse covariance structure of Z.
Rho_Z_CovMu	Numeric sequence, Lasso type penalty for the product of covariance matrix and mean of Z

Value

A list. Containing model BICs of different combination of tuning parameters.

Examples

```
## Not run:
tuenpar <- tune.lucid(G = G1, Z = Z1, Y = Y1, family = "binary",
Rho_G = seq(0.01, 0.02, by = 0.005),
Rho_Z_InvCov = seq(0.1, 0.3, by = 0.1),
Rho_Z_CovMu = seq(80, 100, by = 10))

## End(Not run)
```

Index