def_tune .......................................................... 7
est_lucid ........................................................ 8
G1 .............................................................. 10
G2 .............................................................. 10
plot_lucid ...................................................... 11
pred_lucid ...................................................... 12
sem_lucid ....................................................... 13
summary_lucid .................................................. 15
tune_lucid ...................................................... 16
Y1 .............................................................. 18
Y2 .............................................................. 18
Z1 .............................................................. 19
Z1_Incomp ...................................................... 19
Z2 .............................................................. 20

Index ......................................................... 21

--

boot_lucid

Bootstrap method to estimate variability of latent clusters

Description

boot_lucid provides SEs of parameter estimates from a LUCID model through bootstrapping.

Usage

```r
boot_lucid(
  G = NULL,
  CoG = NULL,
  Z = NULL,
  Y,
  CoY = NULL,
  useY = TRUE,
  family = "binary",
  K = 2,
  Pred = TRUE,
  initial = def_initial(),
  itr_tol = def_tol(),
  tunepar = def_tune(),
  R = 100,
  DeltaE = TRUE,
  NCPUs = detectCores() - 1
)
```
**boot_lucid**

**Arguments**

G  Genetic features, a matrix
CoG  Covariates to be included in the G->X path
Z  Biomarker data, a matrix
Y  Disease outcome, a vector
CoY  Covariates to be included in the X->Y path
useY  Using Y or not, default is TRUE
family  "binary" or "normal" for Y
K  Pre-specified # of latent clusters, default is 2
Pred  Flag to compute posterior probability of latent cluster with fitted model, default is TRUE
initial  A list of initial model parameters will be returned for integrative clustering
itr_tol  A list of tolerance settings will be returned for integrative clustering
tunepar  A list of tuning parameters and settings will be returned for integrative clustering
R  The number of bootstrap replicates, default is 100
DeltaE  Flag to return the difference in parameter estimate across latent clusters, default is TRUE
NCpus  The number of processes to be used in parallel computing, default is total number of cores minus 1

**Value**

`boot_lucid` returns an object of list containing a "boot" class object of LUCID fit and a summary of bootstrap results:

- **Bootstrap**  an object of "boot" class after bootstrapping a LUCID model
- **Results**  A summary of bootstrap includes original estimate, a bias of the bootstrap estimate, standard error of the bootstrap estimate, and three types of bootstrap confidence intervals based on normal approximation, basic, percentile bootstrap methods

**Author(s)**

Cheng Peng, Zhao Yang, David V. Conti

**References**


Examples

```r
## Not run:
boot_lucid(G = G1, CoG = CoG, Z = Z1, Y = Y1, CoY = CoY, family = "binary", R=500)

## End(Not run)
```

---

### CoG

**Covariate Set in the G->X path**

**Description**

A simulated dataset containing one of the optional components to run `est_lucid`, `plot_lucid`, and `tune_lucid`. The variables are as follows:

**Usage**

CoG

**Format**

A set with 2000 rows and 5 variables:

- **GC1** - **GC3** Three continuous covariates
- **GC4**, **GC5** Two binary covariates

---

### CoY

**Covariate Set in the X->Y path**

**Description**

A simulated dataset containing one of the optional components to run `est_lucid`, `plot_lucid`, and `tune_lucid`. The variables are as follows:

**Usage**

CoY

**Format**

A set with 2000 rows and 5 variables:

- **YC1** - **YC3** Three continuous covariates
- **YC4**, **YC5** Two binary covariates
def_initial

Define initial values of parameters for clustering

Description

Defines initial values of model parameters in est_lucid, sem_lucid, & tune_lucid fitting.

Usage

```r
def_initial(
  init_b = NULL,
  init_m = NULL,
  init_s = NULL,
  init_g = NULL,
  init_pcluster = NULL
)
```

Arguments

*init_b*  
Initial model parameters of $\beta$, genetic effects parameter: $K \times (\text{ncol}(G)+1)$ dimensional matrix, each row refers to a latent cluster and the first column is the intercept.

*init_m*  
Initial model parameters of $\mu$, biomarker mean effects parameters: $K \times \text{ncol}(Z)$ dimensional matrix, each row refers to a latent cluster.

*init_s*  
Initial model parameters of $\Sigma$, biomarker covariance matrix: a list of $K \times \text{ncol}(Z) \times \text{ncol}(Z)$ matrices.

*init_g*  
Initial model parameters of $\gamma$, outcome effects parameter: a vector with a length of $K$ for binary $Y$ or $2K$ for continuous $Y$. For binary $Y$, they are log odds in $K$ clusters; for continuous $Y$, they are $K$ cluster-specific means followed by standard deviations in $K$ clusters.

*init_pcluster*  
Initial probabilities of latent clusters.

Value

A list of initial model parameters will be returned for integrative clustering.

Author(s)

Cheng Peng, Zhao Yang, David V. Conti
def_tol

Define maximum number of iteration and convergence

Description
Defines tolerance settings in est_lucid, sem_lucid, & tune_lucid fitting.

Usage

```r
def_tol(
    MAX_ITR = 100,
    MAX_TOT_ITR = 10000,
    reltol = 1e-08,
    tol_b = 1e-04,
    tol_m = 1e-04,
    tol_s = 1e-04,
    tol_g = 1e-04,
    tol_p = 1e-04,
    tol_sem = 0.001
)
```

Arguments

- **MAX_ITR**: Maximum number of iterations, integer, default is 100
- **MAX_TOT_ITR**: Maximum number of total iterations, integer, default is 10000
- **reltol**: Convergence cut-off using a relative tolerance, default is 1e-8
- **tol_b**: Convergence criteria of $\beta$, genetic effects parameter, default is 1e-4
- **tol_m**: Convergence criteria of $\mu$, biomarker mean effects parameters, default is 1e-4
- **tol_s**: Convergence criteria of $\Sigma$, biomarker covariance matrix, default is 1e-4
- **tol_g**: Convergence criteria of $\gamma$, outcome effects parameter, default is 1e-4
- **tol_p**: Convergence criteria of the probability of latent clusters, default is 1e-4
- **tol_sem**: Convergence criteria of SEM, default is 1e-3

Value

A list of tolerance settings will be returned for integrative clustering.

Author(s)

Cheng Peng, Zhao Yang, David V. Conti
def_tune

Define tuning parameters for regularization during integrative clustering

Description

Defines selection options and tuning parameters in est_lucid, sem_lucid fitting.

Usage

```r
def_tune(
  Rho_G = -9,
  Rho_Z_InvCov = 0,
  Rho_Z_CovMu = 0,
  Select_G = FALSE,
  Select_Z = FALSE
)
```

Arguments

- **Rho_G**: Penalty for selection on genetic data, numeric, default is -9 using a sequence of penalties
- **Rho_Z_InvCov**: Penalty for the inverse of covariance of biomarkers, numeric, default is 0
- **Rho_Z_CovMu**: Penalty for the product of covariance and mean of biomarkers, numeric, default is 0
- **Select_G**: Flag to do model selection on genetic data, default is FALSE
- **Select_Z**: Flag to do model selection on biomarker data, default is FALSE

Value

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

Author(s)

Cheng Peng, Zhao Yang, David V. Conti
est_lucid

Estimating latent clusters with multi-omics data

Description

est_lucid estimates an integrated cluster assignment of genetic effects using complete biomarker data with/without disease outcomes. Options to produce sparse solutions for cluster-specific parameter estimates under a circumstance of analyzing high-dimensional data are also provided. An IntClust object will be produced.

Usage

est_lucid(
  G = NULL,
  CoG = NULL,
  Z = NULL,
  Y,
  CoY = NULL,
  useY = TRUE,
  family = "binary",
  K = 2,
  Pred = TRUE,
  initial = def_initial(),
  itr_tol = def_tol(),
  tunepar = def_tune()
)

Arguments

G          Genetic features, a matrix
CoG        Covariates to be included in the G->X path
Z          Biomarker data, a matrix, can be incomplete and have missing values
Y          Disease outcome, a vector
CoY        Covariates to be included in the X->Y path
useY       Using Y or not, default is TRUE
family     "binary" or "normal" for Y
K           Pre-specified # of latent clusters, default is 2
Pred       Flag to compute posterior probability of latent cluster with fitted model, default is TRUE
initial    A list of initial model parameters will be returned for integrative clustering
itr_tol    A list of tolerance settings will be returned for integrative clustering
tunepar    A list of tuning parameters and settings will be returned for integrative clustering
est_lucid

Value

est_lucid returns an object of list containing parameters estimates, predicted probability of latent clusters, and other features:

- **beta**: Estimates of genetic effects, matrix
- **mu**: Estimates of cluster-specific biomarker means, matrix
- **sigma**: Estimates of cluster-specific biomarker covariance matrix, list
- **gamma**: Estimates of cluster-specific disease risk, vector
- **pcluster**: Probability of cluster, when G is null
- **pred**: Predicted probability of belonging to each latent cluster

Author(s)

Cheng Peng, Zhao Yang, David V. Conti

References


Examples

```r
# Integrate clustering without feature selection
set.seed(10)
IntClusFit <- est_lucid(G=G1,Z=Z1,Y=Y1,K=2,family="binary",Pred=TRUE)

## Not run:
# Re-run the model with covariates in the G->X path
IntClusCoFit1 <- est_lucid(G=G1,CoG=CoG,Z=Z1,Y=Y1,K=2,family="binary",Pred=TRUE)

# Re-run the model with covariates in the X->Y path
IntClusCoFit2 <- est_lucid(G=G1,Z=Z1,Y=Y1,CoY=CoY,K=2,family="binary",Pred=TRUE)

# Re-run the model with covariates in both G->X and X->Y paths
IntClusCoFit3 <- est_lucid(G=G1,CoG=CoG,Z=Z1,Y=Y1,CoY=CoY,K=2,family="binary",Pred=TRUE)

# Model fit with incomplete biomarker data and covariates in both G->X & X->Y paths
IntClusCoFit3_Incomp <- est_lucid(G=G1,CoG=CoG,Z=Z1_Incomp,Y=Y1,CoY=CoY,K=2,family="binary")

## End(Not run)
```
Genetic Features Set 1

Description
A simulated dataset containing one of the components to run est_lucid, plot_lucid, and tune_lucid. The variables are as follows:

Usage
G1

Format
A set with 2000 rows and 10 variables:

CG1 - CG5 Causal SNPs
NG1 - NG5 Null SNPs

Genetic Features Set 2

Description
A simulated dataset containing one of the components to run sem_lucid. The variables are as follows:

Usage
G2

Format
A set with 2000 rows and 10 variables:

CG1 - CG5 Causal SNPs
NG1 - NG5 Null SNPs
plot_lucid

Plot Sankey diagram for integrative clustering

Description
plot_lucid generates a Sankey diagram for the results of integrative clustering based on an IntClust object.

Usage
plot_lucid(x, switch = FALSE, colorScale = default)

Arguments
- x: An IntClust class object
- switch: An indicator to do label switching with a descending order in gamma or not, the default is FALSE
- colorScale: D3 color scheme for the Sankey diagram

Author(s)
Cheng Peng, Zhao Yang, David V. Conti

References

Examples
# Run the model with covariates in the G->X path
IntClusCoFit1 <- est_lucid(G=G1,CoG=CoG,Z=Z1,Y=Y1,K=2,family="binary",Pred=TRUE)

# Visualize the results of integrative clustering
plot_lucid(IntClusCoFit1)
pred_lucid Model Predictions for LUCID

Description
pred_lucid produces predicted values for latent clusters and outcome with an IntClust object and new data.

Usage
pred_lucid(Fit = NULL, G = NULL, CoG = NULL, Z = NULL, Y = NULL, CoY = NULL)

Arguments
Fit An IntClust class object
G Genetic effects, a matrix
CoG Covariates to be included in the G->X path
Z Biomarker data, a matrix
Y Disease outcome, a vector; default is NULL
CoY Covariates to be included in the X->Y path

Value
pred_lucid returns a list containing predicted values.

pred_cluster predicted probabilities for latent clusters with/without the outcome
pred_outcome predicted values for outcome

Author(s)
Cheng Peng, Zhao Yang, David V. Conti

References

Examples
set.seed(10)
IntClusFit <- est_lucid(G=G1,Z=Z1,Y=Y1,K=2,family="binary",Pred=TRUE)
GPred <- G2[1:20,1]; ZPred <- Z2[1:20,]
PRED <- pred_lucid(Fit = IntClusFit, G=GPred, CoG = NULL, Z=ZPred, CoY = NULL)
**Description**

`sem_lucid` provides standard errors (SE) of parameter estimates when performing latent cluster analysis with multi-omics data. SEs are obtained through supplemented EM-algorithm (SEM).

**Usage**

```r
sem_lucid(
  G = NULL,
  Z = NULL,
  Y,
  family = "binary",
  useY = TRUE,
  K = 2,
  initial = def_initial(),
  itr_tol = def_tol(),
  Pred = TRUE,
  Get_SE = TRUE,
  Ad_Hoc_SE = FALSE
)
```

**Arguments**

- **G**: Genetic features, a matrix
- **Z**: Biomarker data, a matrix
- **Y**: Disease outcome, a vector
- **family**: "binary" or "normal" for Y
- **useY**: Using Y or not, default is TRUE
- **K**: Pre-specified # of latent clusters, default is 2
- **initial**: A list of initial model parameters will be returned for integrative clustering
- **itr_tol**: A list of tolerance settings will be returned for integrative clustering
- **Pred**: Flag to compute predicted disease probability with fitted model, boolean, default is TRUE
- **Get_SE**: Flag to perform SEM to get SEs of parameter estimates, default is TRUE
- **Ad_Hoc_SE**: Flag to fit ad hoc regression models to get SEs of parameter estimates, default is FALSE
Value

sem_lucid returns an object of list containing parameters estimates, their corresponding standard errors, and other features:

- **beta**: Estimates of genetic effects, matrix
- **se_beta**: SEM standard errors of Beta
- **se_ah_beta**: Ad hoc standard errors of Beta
- **mu**: Estimates of cluster-specific biomarker means, matrix
- **se_mu**: SEM standard errors of Mu
- **se_ah_mu**: Ad hoc standard errors of Mu
- **sigma**: Estimates of cluster-specific biomarker covariance matrix, list
- **gamma**: Estimates of cluster-specific disease risk, vector
- **se_gamma**: SEM standard errors of Gamma
- **se_ah_gamma**: Ad hoc standard errors of Gamma
- **pcluster**: Probability of cluster, when G is null
- **pred**: Predicted probability of belonging to each latent cluster

Author(s)

Cheng Peng, Zhao Yang, David V. Conti

References


Examples

```r
## Not run:
sem_lucid(G=G2,Z=Z2,Y=Y2,useY=TRUE,K=2,Pred=TRUE,family="normal",Get_SE=TRUE,
itr_tol = def_tol(MAX_ITR=1000,MAX_TOT_ITR=3000))
## End(Not run)
```
**summary_lucid**

*Summarize results for integrative clustering*

**Description**

`summary_lucid` generates a summary for the results of integrative clustering based on an `IntClust` object.

**Usage**

```r
summary_lucid(x, switch = FALSE, order = NULL)
```

**Arguments**

- `x`: An `IntClust` class object
- `switch`: An indicator to do label switching or not, the default is FALSE
- `order`: A customized order for label switching, a vector with a length of K; the default is NULL, which is a descending order in gamma

**Value**

`summary_lucid` returns a list containing important outputs from an `IntClust` object.

- **Beta**: Estimates of genetic effects, matrix
- **Mu**: Estimates of cluster-specific biomarker means, matrix
- **Gamma**: Estimates of cluster-specific disease risk, vector
- **select_G**: A logical vector indicates non-zero genetic features
- **select_Z**: A logical vector indicates non-zero bio-features
- **NoG**: A total # of non-zero genetic features
- **NoZ**: A total # of non-zero bio-features
- **BIC**: Model BIC

**Author(s)**

Cheng Peng, Zhao Yang, David V. Conti

**References**

Examples

# For a testing dataset with 10 genetic features (5 causal) and 4 biomarkers (2 causal)

# Integrative clustering without feature selection
set.seed(10)
IntClusFit <- est_lucid(G=G1,Z=Z1,Y=Y1,K=2,family="binary",Pred=TRUE)

# Check important model outputs
summary_lucid(IntClusFit)

tune_lucid

Parallel Grid Search for Tuning Parameters in Latent Cluster Analysis

Description

tune_lucid fits regularized latent cluster models with various combinations of three tuning parameters based on joint inference across data types to perform a grid-search helping determine an optimal choice of three tuning parameters with minimum model BIC.

Usage

tune_lucid(
  G = NULL,
  CoG = NULL,
  Z = NULL,
  CoY = NULL,
  Y,
  K,
  Family,
  USEY = TRUE,
  initial = def_initial(),
  LRho_g,
  URho_g,
  NoRho_g,
  LRho_z_invcov,
  URho_z_invcov,
  NoRho_z_invcov,
  LRho_z_covmu,
  URho_z_covmu,
  NoRho_z_covmu,
  NoCores = detectCores() - 1
)

Arguments

G          Genetic features, a matrix
CoG        Covariates to be added in G->X path
tune_lucid

Z Biomarker data, a matrix
CoY Covariates to be added in X->Y path
Y Disease outcome, a vector
K Pre-specified # of latent clusters
Family "binary" or "normal" for Y
USEY Using Y or not, default is TRUE
initial A list of initial model parameters will be returned for integrative clustering
LRho_g Lower limit of the penalty for selection on genetic data
URho_g Upper limit of the penalty for selection on genetic data
NoRho_g Number of Rho_g for grid-search
LRho_z_invcov Lower limit of the penalty for the inverse of covariance of biomarkers
URho_z_invcov Upper limit of the penalty for the inverse of covariance of biomarkers
NoRho_z_invcov Number of Rho_z_invcov for grid-search
LRho_z_covmu Lower limit of the penalty for the product of covariance and mean of biomarkers
URho_z_covmu Upper limit of the penalty for the product of covariance and mean of biomarkers
NoRho_z_covmu Number of Rho_z_covmu for grid-search
NoCores Number of CPU cores for parallel grid-search, default is total number of cores minus 1

Value
tune_lucid returns an object of list containing Modelfits, Results, and Optimal:

Modelfits Latent cluster model fits for a combination of given tuning parameters
Results Summary results of grid-search
Optimal Features of the optimal model with minimum BIC in the grid-search summary

Author(s)
Cheng Peng, Zhao Yang, David V. Conti

References

Examples
# For a testing dataset with 10 genetic features (5 causal) and 4 biomarkers (2 causal)
# Parallel grid-search with 8 combinations of tuning parameters
## Not run:
GridSearch <- tune_lucid(G=G1, Z=Z1, Y=Y1, K=2, Family="binary", USEY = TRUE, NoCores = 2,
                        LRho_g = 0.008, URho_g = 0.012, NoRho_g = 2,
                        LRho_z_invcov = 0.04, URho_z_invcov = 0.06, NoRho_z_invcov = 2,
LRho_z_covmu = 90, URho_z_covmu = 100, NoRho_z_covmu = 2)

GridSearch$Results
# Determine the best tuning parameters
GridSearch$Optimal

## End(Not run)

---

### Y1

**Outcome Set 1**

**Description**

A simulated dataset containing one of the components to run est_lucid, plot_lucid, and tune_lucid. The variables are as follows:

**Usage**

Y1

**Format**

A set with 2000 rows and 1 variable:

Y1 A binary outcome

---

### Y2

**Outcome Set 2**

**Description**

A simulated dataset containing one of the components to run sem_lucid. The variables are as follows:

**Usage**

Y2

**Format**

A set with 2000 rows and 1 variable:

Y2 A continuous outcome
Biomarker Set 1

Description
A simulated dataset containing one of the components to run `est_lucid`, `plot_lucid`, and `tune_lucid`. The variables are as follows:

Usage
Z1

Format
A set with 2000 rows and 4 variables:

\[ CZ1, CZ2 \] Causal biomarkers
\[ NZ1, NZ2 \] Null biomarkers

Incomplete Biomarker Set 1

Description
A simulated dataset containing one of the components to run `est_lucid` incomplete option, `plot_lucid`, and `tune_lucid`. The variables are as follows:

Usage
Z1_Incomp

Format
A set with 2000 rows and 4 variables, 500 rows are NAs:

\[ CZ1, CZ2 \] Causal biomarkers
\[ NZ1, NZ2 \] Null biomarkers
Description

A simulated dataset containing one of the components to run sem_lucid. The variables are as follows:

Usage

Z2

Format

A set with 2000 rows and 4 variables:

CZ1, CZ2  Causal biomarkers
NZ1, NZ2  Null biomarkers
Index

*Topic **Grid-search**
  tune_lucid, 16

*Topic **Parameter**
  tune_lucid, 16

*Topic **SEM**
  sem_lucid, 13

*Topic **Tuning**
  tune_lucid, 16

*Topic **bootstrap**
  boot_lucid, 2

*Topic **cluster**
  boot_lucid, 2
  est_lucid, 8
  sem_lucid, 13

*Topic **datasets**
  CoG, 4
  CoY, 4
  G1, 10
  G2, 10
  Y1, 18
  Y2, 18
  Z1, 19
  Z1_Incomp, 19
  Z2, 20

*Topic **latent**
  boot_lucid, 2
  est_lucid, 8
  sem_lucid, 13

boot_lucid, 2

CoG, 4
CoY, 4

def_initial, 5
def_tol, 6
def_tune, 7

est_lucid, 5–7, 8

G1, 10

G2, 10
plot_lucid, 11
pred_lucid, 12

sem_lucid, 5–7, 13
summary_lucid, 15

tune_lucid, 5, 6, 16

Y1, 18
Y2, 18

Z1, 19
Z1_Incomp, 19
Z2, 20