# Package 'LACE'

March 28, 2020

**Version** 1.0.0

**Date** 2020-03-28

**Title** Longitudinal Analysis of Cancer Evolution (LACE)

**Maintainer** Daniele Ramazzotti <daniele.ramazzotti@yahoo.com>

**Depends** R (>= 3.6.0)

**Imports** graphics, grDevices, igraph, parallel, RColorBrewer, Rfast, stats, utils

**Suggests** BiocGenerics, BiocStyle, testthat, knitr

**Name** LACE: an R package for the inference of longitudinal cancer evolution models

**Description** LACE is an algorithmic framework that processes single-cell somatic mutation profiles from cancer samples collected at different time points and in distinct experimental settings, to produce longitudinal models of cancer evolution. The approach solves a Boolean Matrix Factorization problem with phylogenetic constraints, by maximizing a weighed likelihood function computed on multiple time points.

**Encoding** UTF-8

**LazyData** TRUE

**License** file LICENSE

**URL** https://github.com/BIMIB-DISCo/LACE

**BugReports** https://github.com/BIMIB-DISCo/LACE

**biocViews** BiomedicalInformatics

**RoxygenNote** 7.0.2

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Daniele Ramazzotti [cre, aut],
Fabrizio Angaroni [aut],
Davide Maspero [aut],
Alex Graudenzi [aut]

## R topics documented:

---

`compute.mutation.distance` *compute.mutation.distance*

---

## Description

Compute mutation distance from LACE corrected genotype.

## Usage

```
compute.mutation.distance(inference)
```

## Arguments

| | |
|---|---|
| `inference` | Results of the inference by LACE. |

## Examples

```
data(inference)
mutation_distance <- compute.mutation.distance(inference)
```

---

`data` *mutation data from Rambow, Florian, et al. "Toward minimal residual disease-directed therapy in melanoma." Cell 174.4 (2018): 843-855.*

---

## Description

mutation data from Rambow, Florian, et al. "Toward minimal residual disease-directed therapy in melanoma." Cell 174.4 (2018): 843-855.

## Usage

```
data(data)
```

## Format

list of mutation data for four time points

## Value

list of mutational data for a total of 475 single cells

## Source

Rambow, Florian, et al. "Toward minimal residual disease-directed therapy in melanoma." Cell 174.4 (2018): 843-855.

inference *results obtained with the function LACE on the provided input data from Rambow, Florian, et al. "Toward minimal residual disease-directed therapy in melanoma." Cell 174.4 (2018): 843-855.*

## Description

results obtained with the function LACE on the provided input data from Rambow, Florian, et al. "Toward minimal residual disease-directed therapy in melanoma." Cell 174.4 (2018): 843-855.

## Usage

```
data(inference)
```

## Format

results obtained with the function LACE on the provided input data

## Value

results obtained with the function LACE on the provided input data

LACE *LACE*

## Description

Perform inference of the maximum likelihood clonal tree from longitudinal data.

## Usage

```
LACE(
  D,
  lik_w = NULL,
  alpha = NULL,
  beta = NULL,
  initialization = NULL,
  keep_equivalent = TRUE,
  check_indistinguishable = TRUE,
  num_rs = 50,
  num_iter = 10000,
  n_try_bs = 500,
  learning_rate = 1,
  marginalize = FALSE,
  num_processes = Inf,
  seed = NULL,
  verbose = TRUE,
  log_file = ""
)
```

## Arguments

| | |
|---|---|
| D | Mutation data from multiple experiments for a list of driver genes. |
| lik_w | Weight for each data point. If not provided, weights to correct for sample sizes are used. |
| alpha | False positive error rate provided as list of elements; if a vector of alpha (and beta) is provided, the inference is performed for multiple values and the solution at maximum-likelihood is returned. |
| beta | False negative error rate provided as list of elements; if a vector of beta (and alpha) is provided, the inference is performed for multiple values and the solution at maximum-likelihood is returned. |
| initialization | Starting point of the mcmc; if not provided, a random starting point is used. |
| keep_equivalent | Boolean. Shall I return results (B and C) at equivalent likelihood with the best returned solution? |
| check_indistinguishable | Boolean. Shall I remove any indistinguishable event from input data prior inference? |
| num_rs | Number of restarts during mcmc inference. |
| num_iter | Maximum number of mcmc steps to be performed during the inference. |
| n_try_bs | Number of steps without change in likelihood of best solution after which to stop the mcmc. |
| learning_rate | Parameter to tune the probability of accepting solutions at lower values during mcmc. Value of learning_rate = 1 (default), set a probability proportional to the difference in likelihood; values of learning_rate greater than 1 inclease the chance of accepting solutions at lower likelihood during mcmc while values lower than 1 decrease such probability. |
| marginalize | Boolean. Shall I marginalize C when computing likelihood? |
| num_processes | Number of processes to be used during parallel execution. To execute in single process mode, this parameter needs to be set to either NA or NULL. |
| seed | Seed for reproducibility. |
| verbose | Boolean. Shall I print to screen information messages during the execution? |
| log_file | log file where to print outputs when using parallel. If parallel execution is disabled, this parameter is ignored. |

## Value

A list of 9 elements: B, C, clones_prevalence, relative_likelihoods, joint_likelihood, clones_summary and error_rates. Here, B returns the maximum likelihood longitudinal clonal tree, C the attachment of cells to clones, corrected_genotypes the corrected genotypes and clones_prevalence clones' prevalence; relative_likelihoods and joint_likelihood are respectively the likelihood of the solutions at each individual time points and the joint likelihood; clones_summary provide a summary of association of mutations to clones. In equivalent_solutions, solutions (B and C) with likelihood equivalent to the best solution are returned. Finally error_rates provides the best values of alpha and beta among the considered ones.

## Examples

```
data(data)
inference = LACE(D = data,
                 lik_w = c(0.2308772,0.2554386,0.2701754,0.2435088),
                 alpha = list(c(0.10,0.05,0.05,0.05)),
                 beta = list(c(0.10,0.05,0.05,0.05)),
                 keep_equivalent = FALSE,
                 num_rs = 5,
                 num_iter = 10,
                 n_try_bs = 5,
                 num_processes = NA,
                 seed = 12345,
                 verbose = FALSE)
```

---

`longitudinal.tree.plot`

*longitudinal.tree.plot*

---

## Description

Plot a longitudinal tree inferred by LACE.

## Usage

```
longitudinal.tree.plot(
  inference,
  labels = "mutations",
  clone_labels = NULL,
  label.cex = 1,
  iter_max = 100,
  size = 500,
  size2 = NULL,
  tk_plot = FALSE,
  tp_mark = TRUE,
  tp_mark_alpha = 0.5,
  legend = TRUE,
  legend_position = "topleft",
  legend_cex = 0.8
)
```

## Arguments

| | |
|---|---|
| `inference` | Results of the inference by LACE. |
| `labels` | Specify which type of label should be placed on the tree; options are, "mutations": parental edges are labeled with the acquired mutation between the two nodes (genotypes); "clones": nodes (genotypes) are labeled with their last acquired mutation; "both": either nodes and edges are labeled as specified above; "none": no labels will show on the longitudinal tree. |
| `clone_labels` | Character vector that specifies the name of the nodes (genotypes). If it is NULL (default), nodes will be labeled as specified by "label" parameter. |

| | |
|---|---|
| label.cex | Specify the size of the labels. |
| iter_max | Maximum number of iteration to be used to remove intersecting edges. |
| size | Specify size of the nodes. The final area is proportional with the node prevalence. |
| size2 | Specify the size of the second dimension of the nodes. If NULL (default), it is set equal to "size". |
| tk_plot | If TRUE, uses tkplot function from igraph library to plot an interactive tree. Default is FALSE. |
| tp_mark | If TRUE (defaul) the function draws different colored area under the nodes in different time points. |
| tp_mark_alpha | Specify the alpha value of the area drawed when tp_mark = TRUE. |
| legend | If TRUE (default) a legend will be displayed on the plot. |
| legend_position | Specify the legend position. |
| legend_cex | Specify size of the legend text. |

## Examples

```
data(inference)
clone_labels = c("ARPC2","PRAME","HNRNPC","COL1A2","RPL5","CCT8")
longitudinal.tree.plot(inference = inference,
                       labels = "clones",
                       clone_labels = clone_labels,
                       legend_position = "topleft")
```

# Index