

# Package ‘LACE’

March 31, 2020

**Version** 1.0.0

**Date** 2020-03-31

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

**Maintainer** Davide Maspero <d.maspero@campus.unimib.it>

**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 . . . . .	2
data . . . . .	2

inference . . . . .	3
LACE . . . . .	3
longitudinal.tree.plot . . . . .	5
<b>Index</b>	<b>7</b>

---

compute.mutation.distance
<i>compute.mutation.distance</i>

---

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

---

**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	<i>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.</i>
-----------	--

---

**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	<i>LACE</i>
------	-------------

---

**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**

<code>D</code>	Mutation data from multiple experiments for a list of driver genes.
<code>lik_w</code>	Weight for each data point. If not provided, weights to correct for sample sizes are used.
<code>alpha</code>	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.
<code>beta</code>	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.
<code>initialization</code>	Starting point of the mcmc; if not provided, a random starting point is used.
<code>keep_equivalent</code>	Boolean. Shall I return results (B and C) at equivalent likelihood with the best returned solution?
<code>check_indistinguishable</code>	Boolean. Shall I remove any indistinguishable event from input data prior inference?
<code>num_rs</code>	Number of restarts during mcmc inference.
<code>num_iter</code>	Maximum number of mcmc steps to be performed during the inference.
<code>n_try_bs</code>	Number of steps without change in likelihood of best solution after which to stop the mcmc.
<code>learning_rate</code>	Parameter to tune the probability of accepting solutions at lower values during mcmc. Value of <code>learning_rate</code> = 1 (default), set a probability proportional to the difference in likelihood; values of <code>learning_rate</code> greater than 1 increase the chance of accepting solutions at lower likelihood during mcmc while values lower than 1 decrease such probability.
<code>marginalize</code>	Boolean. Shall I marginalize C when computing likelihood?
<code>num_processes</code>	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.
<code>seed</code>	Seed for reproducibility.
<code>verbose</code>	Boolean. Shall I print to screen information messages during the execution?
<code>log_file</code>	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,
  show_plot = TRUE,
  filename = "lg_output.xml",
  labels = "mutations",
  clone_labels = NULL,
  show_prev = TRUE,
  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.
show_plot	If TRUE (default) output the longitudinal tree to the current graphical device.
filename	Specify the name of the file where to save the longitudinal tree. Dot or graphml formats are supported and are chosen based on the extension of the filename (.dot or .xml).

<code>labels</code>	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.
<code>clone_labels</code>	Character vector that specifies the name of the nodes (genotypes). If it is NULL (default), nodes will be labeled as specified by "label" parameter.
<code>show_prev</code>	If TRUE (default) add to clones label the corresponding prevalence.
<code>label.cex</code>	Specify the size of the labels.
<code>iter_max</code>	Maximum number of iteration to be used to remove intersecting edges.
<code>size</code>	Specify size of the nodes. The final area is proportional with the node prevalence.
<code>size2</code>	Specify the size of the second dimension of the nodes. If NULL (default), it is set equal to "size".
<code>tk_plot</code>	If TRUE, uses tkplot function from igraph library to plot an interactive tree. Default is FALSE.
<code>tp_mark</code>	If TRUE (default) the function draws different colored area under the nodes in different time points.
<code>tp_mark_alpha</code>	Specify the alpha value of the area drawn when <code>tp_mark = TRUE</code> .
<code>legend</code>	If TRUE (default) a legend will be displayed on the plot.
<code>legend_position</code>	Specify the legend position.
<code>legend_cex</code>	Specify size of the legend text.

**Value**

An igraph object `g` with the longitudinal tree inferred by LACE.

**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

`compute.mutation.distance`, [2](#)

`data`, [2](#)

`inference`, [3](#)

`LACE`, [3](#)

`longitudinal.tree.plot`, [5](#)