

Longitudinal Analysis of Cancer Evolution with LACE

***Daniele Ramazzotti¹, Fabrizio Angaroni², Davide Maspero^{2,3},
Gianluca Ascolani², Isabella Castiglioni^{4,5}, Rocco Piazza¹,
Marco Antoniotti², and Alex Graudenzi⁵***

¹School of Medicine and Surgery, Univ. of Milan-Bicocca, Monza, Italy.

²Dept. of Informatics, Systems and Communication, Univ. of Milan-Bicocca, Milan, Italy.

³Fondazione IRCCS Istituto Nazionale dei Tumori, Milan, Italy.

⁴Department of Physics "Giuseppe Occhialini", Univ. of Milan-Bicocca, Milan, Italy.

⁵Inst. of Molecular Bioimaging and Physiology, Consiglio Nazionale delle Ricerche (IBFM-CNR), Segrate, Milan, Italy.

March 3, 2020

Overview. 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.

In this vignette, we give an overview of the package by presenting its main functions.

Contents

1	Changelog	2
2	Using the LACE R package	2
3	sessionInfo()	4

1 Changelog

1.0.0 package released on January 2020.

2 Using the LACE R package

We now present an example of longitudinal analysis of cancer evolution with LACE using single-cell data obtained from Rambow, Florian, et al. "Toward minimal residual disease-directed therapy in melanoma." Cell 174.4 (2018): 843-855. The data comprises point mutations for four time points: (1) before treatment, (2) 4 days treatment, (3) 28 days treatment and finally (4) 57 days treatment.

We first load the data.

```
library("LACE")
data(data)
names(data)

## [1] "T1_before_treatment" "T2_4_days_treatment" "T3_28_days_treatment"
## [4] "T4_57_days_treatment"
```

We setup the main parameter in order to perform the inference. First of all, as the three data points may potentially provide sequencing for an unbalanced number of cells, we weight each time point as follow $w_s = (1 - \frac{n_s}{n_T}) / (y - 1)$ in order to account for this. In the formula, e.g., the weight for time point s (w_s) is calculated based on the number of cells observed in the time point (n_s) and the total number of cells in the three time points (n_T). The denominator ($y - 1$, with y being the number of time points, i.e., 3 in our case) aims at normalizing the weights to sum to one.

```
lik_weights = c(0.2308772, 0.2554386, 0.2701754, 0.2435088)
```

The second main parameter to be defined as input is represented by the false positive and false negative error rates, i.e., alpha and beta. We can specify a different rate per time point as a list of rates. When multiple set of rates are provided, LACE performs a grid search in order to estimate the best set of error rates.

```
alpha = list()
alpha[[1]] = c(0.02, 0.01, 0.01, 0.01)
alpha[[2]] = c(0.10, 0.05, 0.05, 0.05)
beta = list()
```

Longitudinal Analysis of Cancer Evolution with LACE

```
beta[[1]] = c(0.10,0.05,0.05,0.05)
beta[[2]] = c(0.10,0.05,0.05,0.05)
head(alpha)

## [[1]]
## [1] 0.02 0.01 0.01 0.01
##
## [[2]]
## [1] 0.10 0.05 0.05 0.05

head(beta)

## [[1]]
## [1] 0.10 0.05 0.05 0.05
##
## [[2]]
## [1] 0.10 0.05 0.05 0.05
```

We can now perform the inference as follow.

```
inference = LACE(D = data,
                 lik_w = lik_weights,
                 alpha = alpha,
                 beta = beta,
                 keep_equivalent = FALSE,
                 num_rs = 5,
                 num_iter = 10,
                 n_try_bs = 5,
                 num_processes = NA,
                 seed = 12345,
                 verbose = FALSE)
```

We notice that the inference resulting on the command above should be considered only as an example; the parameters `num_rs`, `num_iter` and `n_try_bs` representing the number of steps performed during the inference are downscaled to reduce execution time. We refer to the Manual for discussion on default values. We provide within the package results of inferences performed with correct parameters as `RData`.

```
data(inference)
print(names(inference))

## [1] "B" "C" "clones_prevalence"
## [4] "relative_likelihoods" "joint_likelihood" "clones_summary"
## [7] "equivalent_solutions" "error_rates"
```

LACE returns a list of eight elements as results. Namely, B and C provide respectively the maximum likelihood longitudinal tree and cells attachments; clones prevalence, the estimated prevalence of any observed clone; relative likelihoods and joint likelihood the estimated likelihoods for each time point and the weighted 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; notice that in the example we disabled this feature by setting equivalent solutions parameter to `FALSE`. Finally, error rates provide the best error rates (alpha and beta) as estimated by the grid search.

3 sessionInfo()

- R version 3.6.1 (2019-07-05), x86_64-apple-darwin15.6.0
- Locale: C/it_IT.UTF-8/it_IT.UTF-8/C/it_IT.UTF-8/it_IT.UTF-8
- Running under: macOS Catalina 10.15.3
- Matrix products: default
- BLAS:
/Library/Frameworks/R.framework/Versions/3.6/Resources/lib/libRblas.0.dylib
- LAPACK:
/Library/Frameworks/R.framework/Versions/3.6/Resources/lib/libRlapack.dylib
- Base packages: base, datasets, grDevices, graphics, methods, stats, utils
- Other packages: LACE 1.0.0, knitr 1.27
- Loaded via a namespace (and not attached): BiocManager 1.30.10, BiocStyle 2.14.4, Rcpp 1.0.3, RcppZiggurat 0.1.5, Rfast 1.9.8, compiler 3.6.1, digest 0.6.23, evaluate 0.14, highr 0.8, htmltools 0.4.0, magrittr 1.5, parallel 3.6.1, rlang 0.4.2, rmarkdown 2.0, stringi 1.4.5, stringr 1.4.0, tools 3.6.1, xfun 0.12, yaml 2.2.0