

# Quick introduction to SaaRclust

*David Porubsky*

*2018-03-18*

**Package version:** SaaRclust

## Contents

---

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Parameters</b>	<b>2</b>
<b>3</b>	<b>Quick Start</b>	<b>2</b>
3.1	Hard Clustering . . . . .	2
3.2	Soft Clustering . . . . .	2
3.3	Hard & Soft Clustering . . . . .	3
3.4	Export clustered long reads . . . . .	3
3.5	Plot clustering accuracy plots . . . . .	3
<b>4</b>	<b>Session Info</b>	<b>4</b>

## 1 Introduction

---

Strand-seq is a single-cell sequencing technique able to preserve contiguity of individual parental homologues in single-cell. This feature has been shown to be valuable for scaffolding early build genome assemblies as well finding chimeric or misoriented contigs. Here we introduce a SaaRclust as an R based package that implements a novel latent variable model and a corresponding Expectation Maximization (EM) algorithm in order to reliably cluster long sequencing reads by chromosome. Briefly, our approach produces, for each long read, a posterior probability distribution over all chromosomes of origin and read directionalities. In this way, it allows to assess the amount of uncertainty inherent to sparse Strand-seq data on the level of individual reads.

## 2 Parameters

---

**inputfolder:** A folder name where minimap file(s) is stored.

**outputfolder:** A folder name to export the results.

**num.clusters:** Expected number of clusters. (for 22 human autosomes == 44 clusters). However overclustering (~ 54 clusters) is recommended such that smaller chromosomes are not missed

**EM.iter:** Number of iteration of EM algorithm.

**alpha:** Estimated level of background in Strand-seq reads. In other words expected noise in sequencing data caused either by library preparation or mapping to repetitive parts of the genome.

**minLib:** Minimal number of different Strand-seq libraries being represent per every long read.

**upperQ:** Filter out given percentage of long reads with the highest number of Strand-seq alignments.

**logL.th:** Set the difference between objective function from the current and the previous iteration for EM algorithm to converge.

**theta.constrain:** Recalibrate theta values to meet expected distribution of W and C strands across all Strand-seq libraries.

**store.counts:** Logical TRUE/FALSE if to store raw read counts aligned to each long read.

**store.bestAlign:** If set to TRUE (best) representative alignments will be stored in RData object

**numAlignments:** Required number of (best) representative alignments to be used in hard clustering.

**HC.only:** If set to TRUE only the hard clustering will be performed and the rest of the clustering pipeline will be skipped.

**HC.input:** A location to a filename where the hard clustering results are stored.

**verbose:** Set to TRUE if progress messages should be printed.

## 3 Quick Start

---

Download example data from the github repository

```
git clone https://github.com/daewoooo/SaaRclustExampleData
```

Set the location of the example data

```
inputfolder <- 'SaaRclust_exampleData'
```

### 3.1 Hard Clustering

In order to run only k-means based hard clustering on a example data.

```
# Hard clustering
# Remember to set HC.only=TRUE

runSaaRclust(inputfolder=inputfolder, outputfolder="SaaRclust_results", num.clusters=54,
EM.iter=100,alpha=0.01, minLib=10, upperQ=0.95, logL.th=1, theta.constrain=FALSE,
store.counts=FALSE, store.bestAlign=TRUE, numAlignments=3000, HC.only=TRUE, verbose=TRUE)
```

### 3.2 Soft Clustering

If RData object containing hard clustering results is already available you can run only soft clustering.

```
# Setting some variables

HC.input='SaaRclust_results/Clusters/'
minimap.file='SaaRclust_exampleData/...'
```

```
# Soft clustering
# If theta.param & pi.param are set to NULL SaaRclust will try to load them from HC.input.

SaaRclust(minimap.file=minimap.file, outputfolder='SaaRclust_results', num.clusters=47,
EM.iter=100, alpha=0.1, minLib=10, upperQ=0.95, theta.param=NULL, pi.param=NULL, logL.th=1,
theta.constrain=FALSE, store.counts=FALSE, HC.input=HC.input)
```

### 3.3 Hard & Soft Clustering

In order to run both, hard and soft clustering in a single command.

```
# Hard clustering
# Remember to set HC.only=FALSE

runSaaRclust(inputfolder=inputfolder, outputfolder="SaaRclust_results", num.clusters=54,
EM.iter=100,alpha=0.01, minLib=10, upperQ=0.95, logL.th=1, theta.constrain=FALSE,
store.counts=FALSE, store.bestAlign=TRUE, numAlignments=3000, HC.only=FALSE, verbose=TRUE)
```

### 3.4 Export clustered long reads

In order export soft clustered long sequencing reads use function below. Set required threshold for probability values and minimal number of libraries being represented per long read.

```
exportClusteredReads(inputfolder="SaaRclust_results", prob.th=0.5, minLib=5)
```

### 3.5 Plot clustering accuracy plots

NOTE: Working only for data from the original publication.

Rscript below plots clustering accuracy measures presented in the original paper (Fig4 b,d,c)

```
run from the command line:
Rscript /SaaRclust/utils/runPostProcessing.R /SaaRclust/utils/postProcessing.R <outputdirectory>
```

## 4 Session Info

---

```
devtools::session_info()
## Session info -----
## setting value
## version R version 3.3.3 (2017-03-06)
## system x86_64, linux-gnu
## ui X11
## language
## collate en_US.UTF-8
## tz Europe/Berlin
## date 2018-03-18
## Packages -----
## package * version date source
## backports 1.1.2 2017-12-13 CRAN (R 3.3.1)
## base * 3.3.3 2017-03-06 local
## BiocStyle * 2.2.1 2018-03-04 Bioconductor
## datasets * 3.3.3 2017-03-06 local
## devtools 1.13.4 2017-11-09 CRAN (R 3.1.1)
## digest 0.6.13 2017-12-14 CRAN (R 3.3.1)
## evaluate 0.10.1 2017-06-24 CRAN (R 3.3.1)
## graphics * 3.3.3 2017-03-06 local
## grDevices * 3.3.3 2017-03-06 local
## htmltools 0.3.6 2017-04-28 CRAN (R 3.3.1)
## knitr 1.18 2017-12-27 CRAN (R 3.3.1)
## magrittr 1.5 2014-11-22 CRAN (R 3.3.1)
## memoise 1.1.0 2017-04-21 CRAN (R 3.1.1)
## methods * 3.3.3 2017-03-06 local
## Rcpp 0.12.15 2018-01-20 cran (@0.12.15)
## rmarkdown 1.9 2018-03-01 CRAN (R 3.3.3)
## rprojroot 1.2 2017-01-16 CRAN (R 3.1.1)
## stats * 3.3.3 2017-03-06 local
## stringi 1.1.6 2017-11-17 CRAN (R 3.3.1)
## stringr 1.2.0 2017-02-18 CRAN (R 3.3.1)
## tools 3.3.3 2017-03-06 local
## utils * 3.3.3 2017-03-06 local
## withr 2.1.0 2017-11-01 CRAN (R 3.1.1)
## yaml 2.1.16 2017-12-12 CRAN (R 3.3.1)
```

Report any issues [here](#):