

flowClean

Christopher Fletez-Brant, Pratip Chattopadhyay

Modified: April 1, 2014. Compiled: September 19, 2017

Introduction

This package contains the `flowCore` method for performing quality control on flow cytometry datasets. This method is described in [1].

```
> library(flowClean)
> library(flowViz)
> library(grid)
> library(gridExtra)
```

Data

Example data is a real FCS file in which we intentionally perturbed the fluorescent intensity (FI) of a subset of cells along the V705 channel ('<V705-A>').

```
> data(synPerturbed)
> synPerturbed
```

```
flowFrame object '9301d9e4-a160-477f-a5fb-ee7d785d5655'
with 76466 cells and 17 observables:
```

	name	desc	range	minRange	maxRange
\$P1	FSC-A	<NA>	262144	0.00000	262144
\$P2	FSC-H	<NA>	262144	0.00000	262144
\$P3	SSC-A	<NA>	261589	0.00000	261589
\$P4	Time	<NA>	2048	0.00000	2048
\$P5	<B515-A>	CD27 FITC	260954	-26.88460	260954
\$P6	<V705-A>	CD57 QD705	261947	-111.00000	261947
\$P7	<G560-A>	CD95 PE	261196	-111.00000	261196
\$P8	<G660-A>	CD28 Cy5 PE	261507	-111.00000	261507
\$P9	<G710-A>	CD244 Cy55 PE	261763	-111.00000	261763
\$P10	<G780-A>	CD31 Cy7 PE	261402	-111.00000	261402

```

$P11 <R710-A>      CCR7 Ax680 261602 -111.00000 261602
$P12 <R780-A>      CD3 H7 APC 261301 -111.00000 261301
$P13 <V450-A>      CD127 BV421 260964 -35.98385 260964
$P14 <V545-A>      Aq Blu 260949 -22.20724 260949
$P15 <V585-A>      CD8 QD585 261965 -111.00000 261965
$P16 <R660-A>      CD45RA APC 261023 -96.50925 261023
$P17 <V605-A>      CD4 BV605 261131 -111.00000 261131
212 keywords are stored in the 'description' slot

```

Quality Control

The full details are available in [1]. The motivating idea for this methodology is that populations in a flow experiment should be collected nearly uniformly with respect to time of collection. The primary actor in flowClean is the `clean`, which tests for deviations from uniformity of collection. Specifically, the collection time is discretized into l periods, each of which can be considered a N -part composition

$$D_{j=1..l} = [P_1, P_2, \dots, P_N]$$

with each P_i the frequency of a population defined as +/- with respect to some threshold; the default is the median FI of a flow parameter. By default $l = 100$.

Each D_j then undergoes the centered log ratio (CLR) transformation [2]:

$$CLR(D_j) = \left[\ln \frac{P_1}{g(D_j)}; \dots; \ln \frac{P_N}{g(D_j)} \right]$$

where

$$g(D_j) = \sqrt[N]{P_1 P_2 \dots P_N}$$

To avoid `-Inf` values, substitution of zeroes is performed using the 'modified Aitchison' of [3].

The L_p norm of the subset $CLR(D_j) > 0$, denoted $L_p = \|CLR(D_j)\|^+$, where $p = |CLR(D_j) > 0|$, is then calculated for each D_j and changepoint analysis is performed on the set of all $\|CLR(D_j)\|^+$. If there are no changes then the FCS is assumed to contain no errors. Otherwise, the means of the periods are compared relative to the mean of the longest period between changepoints and thresholded according to some k , which empirically works well with $k = 1.3$.

Actually calling `clean` requires only specifying a `flowFrame`, which markers are to be analyzed (generally without the 'scatter' parameters), the name to be given to the output (directory structure can be included) and the file extension:

```

> synPerturbed.c <- clean(synPerturbed, vectMarkers=c(5:16),
+                          filePrefixWithDir="sample_out", ext="fcs", diagnostic=TRUE)

```

```
[1] "flowClean has identified problems in synPerturbed.FCS with 24, 25, 26, 27, 28, 29, 30, 31,
```

```
> synPerturbed.c
```

```
flowFrame object '9301d9e4-a160-477f-a5fb-ee7d785d5655'
with 76466 cells and 18 observables:
```

	name	desc	range	minRange	maxRange
\$P1	FSC-A	<NA>	262144	0.00000	262144
\$P2	FSC-H	<NA>	262144	0.00000	262144
\$P3	SSC-A	<NA>	261589	0.00000	261589
\$P4	Time	<NA>	2048	0.00000	2048
\$P5	<B515-A>	CD27 FITC	260954	-26.88460	260954
\$P6	<V705-A>	CD57 QD705	261947	-111.00000	261947
\$P7	<G560-A>	CD95 PE	261196	-111.00000	261196
\$P8	<G660-A>	CD28 Cy5 PE	261507	-111.00000	261507
\$P9	<G710-A>	CD244 Cy55 PE	261763	-111.00000	261763
\$P10	<G780-A>	CD31 Cy7 PE	261402	-111.00000	261402
\$P11	<R710-A>	CCR7 Ax680	261602	-111.00000	261602
\$P12	<R780-A>	CD3 H7 APC	261301	-111.00000	261301
\$P13	<V450-A>	CD127 BV421	260964	-35.98385	260964
\$P14	<V545-A>	Aq Blu	260949	-22.20724	260949
\$P15	<V585-A>	CD8 QD585	261965	-111.00000	261965
\$P16	<R660-A>	CD45RA APC	261023	-96.50925	261023
\$P17	<V605-A>	CD4 BV605	261131	-111.00000	261131
18	GoodVsBad	GoodVsBad	262144	0.00000	262143

222 keywords are stored in the 'description' slot

The result is an FCS file identical to the input file with a new parameter, 'GoodVsBad', in which 'Good' cells all are given $FI < 10000$ and 'Bad' cells are given $FI \geq 10000$, which allows for easy programmatic gating out of 'Bad' cells from multiple FCS files. This parameter can also be used in plots as any other flow parameter as well.

```
> lgcl <- estimateLogicle(synPerturbed.c, unname(parameters(synPerturbed.c)$name[5:16]))
> synPerturbed.cl <- transform(synPerturbed.c, lgcl)
> p1 <- xyplot(`<V705-A>` ~ `Time`, data=synPerturbed.cl,
+             abs=TRUE, smooth=FALSE, alpha=0.5, xlim=c(0, 100))
> p2 <- xyplot(`GoodVsBad` ~ `Time`, data=synPerturbed.cl,
+             abs=TRUE, smooth=FALSE, alpha=0.5, xlim=c(0, 100), ylim=c(0, 20000))
> rg <- rectangleGate(filterId="gvb", list("GoodVsBad"=c(0, 9999)))
> idx <- filter(synPerturbed.cl, rg)
> synPerturbed.clean <- Subset(synPerturbed.cl, idx)
> p3 <- xyplot(`<V705-A>` ~ `Time`, data=synPerturbed.clean,
+             abs=TRUE, smooth=FALSE, alpha=0.5, xlim=c(0, 100))
> grid.arrange(p1, p2, p3, ncol=3)
```

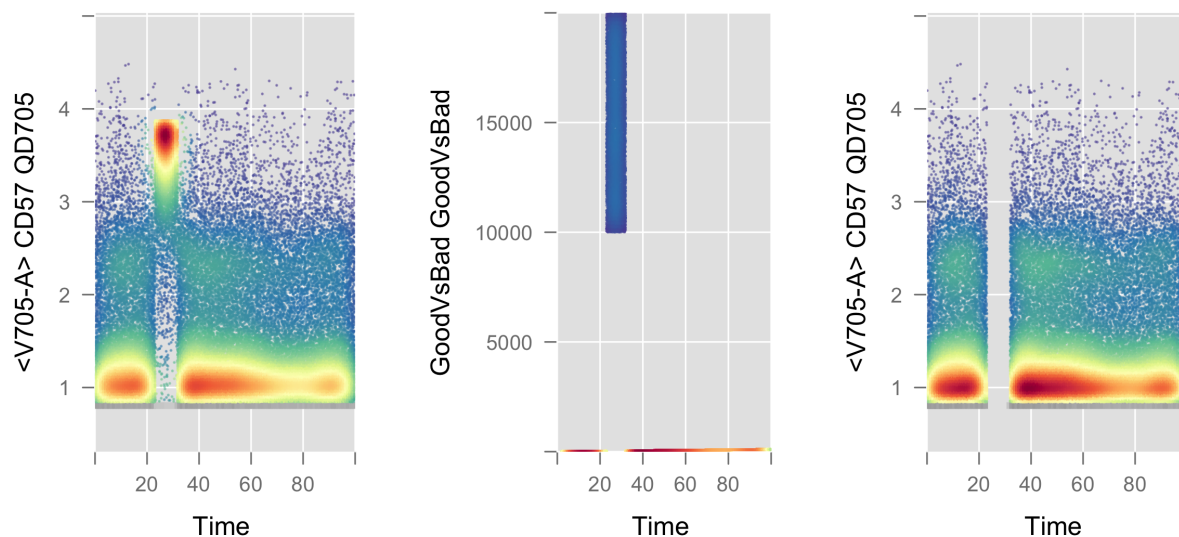


Figure 1: Left) FCS before flowClean. Center) New 'GoodVsBad' parameter. Right) FCS after flowClean and filtering.

SessionInfo

- R version 3.4.1 (2017-06-30), x86_64-apple-darwin15.6.0
- Locale: C/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
- Running under: OS X El Capitan 10.11.6
- Matrix products: default
- BLAS:
/Library/Frameworks/R.framework/Versions/3.4/Resources/lib/libRblas.0.dylib
- LAPACK:
/Library/Frameworks/R.framework/Versions/3.4/Resources/lib/libRlapack.dylib
- Base packages: base, datasets, grDevices, graphics, grid, methods, stats, utils
- Other packages: flowClean 1.16.0, flowCore 1.42.3, flowViz 1.40.0, gridExtra 2.3, lattice 0.20-35
- Loaded via a namespace (and not attached): Biobase 2.36.2, BiocGenerics 0.22.0, DEoptimR 1.0-8, IDPmisc 1.1.17, KernSmooth 2.23-15, MASS 7.3-47, RColorBrewer 1.1-2, Rcpp 0.12.12, bit 1.1-12, changepoint 2.2.2, cluster 2.0.6, compiler 3.4.1, corpcor 1.6.9, graph 1.54.0, gtable 0.2.0, hexbin 1.27.1,

latticeExtra 0.6-28, matrixStats 0.52.2, mvtnorm 1.0-6, parallel 3.4.1, pcaPP 1.9-72, robustbase 0.92-7, rrcov 1.4-3, sfsmisc 1.1-1, stats4 3.4.1, tools 3.4.1, zoo 1.8-0

References

- [1] Fletez-Brant C, Spidlen J, Brinkman R, Roederer M, Chattopadhyay P. Quality Control of flow cytometry data through compositional data analysis. In preparation.
- [2] Aitchison J. A concise guide to compositional data analysis. Compositional Data Analysis Workshop; Girona, Italy.
- [3] Fry J, Fry T, McLaren K. Compositional data analysis and zeros in micro data. CoPS/IMPACT Working Paper Number G-120.