Vignette run cytof analysis

Anja Bråthen Kristoffersen

last updated

date()

## [1] "Thu Jul 14 12:34:45 2022"

# To get started

This explains the lines in the script “run.R”. This is only an explanation of what these lines does.

* First R has to be told which librarys to use. If you have not installed the packages, see “Example-install-packages-in-R.docx”.

library(ComplexHeatmap)  
library(gridExtra)  
library(ggplot2)  
library(grid)  
library(flowCore)  
library(FlowSOM)  
library(org)  
library(data.table)  
library(fs)  
library(betareg)

* make a folder where you want to save the result from you analysis. In this example I have called it

path0 <- fs::path("C:", "Cytof data", "test")   
path0

FALSE C:/Cytof data/test

* make a folder “script” inside this folder, and copy all files from GitHub in here
* then source the file "function.R

source(fs::path(path0, "script", "functions.R"))

* and run the next line which will make a vector of folder names that also are created in path0.

paths <- make\_project\_folders(path0)  
rm(path0)

The function make\_project\_folders() use the packages org form FHI. If you have problems running this, you could rather make the folder structure as below. This structure is needed for the rest of the script to run properly.

paths

## $raw\_data\_path  
## C:/Cytof data/test/raw\_data  
##   
## $R\_path  
## C:/Cytof data/test/R  
##   
## $script\_path  
## C:/Cytof data/test/script  
##   
## $clean\_up\_path  
## C:/Cytof data/test/clean\_up  
##   
## $clean\_data\_path  
## C:/Cytof data/test/clean\_up/clean\_data  
##   
## $clean\_data\_fig\_density\_path  
## C:/Cytof data/test/clean\_up/clean\_data\_fig\_density  
##   
## $clean\_data\_fig\_signal\_path  
## C:/Cytof data/test/clean\_up/clean\_data\_fig\_signal  
##   
## $clean\_data\_info\_path  
## C:/Cytof data/test/clean\_up/clean\_data\_info  
##   
## $clean\_data\_posNeg\_path  
## C:/Cytof data/test/clean\_up/clean\_data\_posNeg  
##   
## $marker\_gating\_path  
## C:/Cytof data/test/clean\_up/marker\_gating  
##   
## $marker\_gating\_fig\_density\_path  
## C:/Cytof data/test/clean\_up/marker\_gating/fig\_density  
##   
## $marker\_gating\_fig\_signal\_path  
## C:/Cytof data/test/clean\_up/marker\_gating/fig\_signal  
##   
## $marker\_gating\_results\_path  
## C:/Cytof data/test/clean\_up/marker\_gating/results  
##   
## $clean\_data\_flowSOM\_results\_path  
## C:/Cytof data/test/clean\_up/flowSOM\_results  
##   
## $meta\_data\_path  
## C:/Cytof data/test/meta\_data

* place all your raw data in the folder “raw data”.
* then before the analysis can be started the function plotSignal() has to be updated. Here I have made an example with 15 files (and one commented away with 66 files), change this function so that it takes as many files as your dataset have.

# CLEAN Up

You are now ready to start the clean up of your data. Run the lines below to get a first cleanup version.

When you are happy with your cleanup you only have to run it once, that is why I have written it within an if-sentence.

if(nrow(file.info(list.files(paths$clean\_data\_path))) == 0){  
 source(fs::path(paths$script\_path, "clean up gating all files.R"))  
}  
  
#

Check the gates produced in both density plots and time-signal-plots which you find in "…\_up\_data\_fig\_density" and "…\_up\_data\_fig\_signal“. If something has to be changed, and most likely it will, you have to update”clean up gating all files.R“. How to do this you fine in”Vignette clean up gating.docx". And rerun by writing

source(fs::path(paths$script\_path, "clean up gating all files.R"))

# Read clean data

You have now cleaned your data and the clean data can be found in the folder "…\_up\_data“. You have also made three files in”…\_up\_data\_info". “percent\_kept\_from\_full\_dataset.csv” tells you how many percent of the full dataset you have kept after each gating, “percent\_kept\_each\_gating.csv” how many percent of the gated dataset you have kept for each gating. The last file “marker\_names\_included\_manual\_shortnames.csv” are a table with marker names. The two first column must be kept as they are the last one can be changed. The last column are the names that will be used in plots and when you in the script refere to a marker. Please update the column “markers\_short\_name” with the names you want to use. Make sure that all markers in the list are markers that you want to use in your analysis. If there are too many markers you could either delete the rows manually or update the file “clean up gating all files.R” that make the script. If you make any update to the file “marker\_names\_included\_manual\_shortnames.csv” remember that this is produced again each time you run “marker\_names\_included\_manual\_shortnames.csv”, so you might also want to save a copy with your changes.

files\_to\_open <- "all"

This will include all files in the clean data folder. If you want to do the analysis with only some files this can be changed to a vector of filenames to use.

rownames(file.info(list.files(paths$clean\_data\_path)))

give you a list of all files in the clean data folder, only .fcs files are read

* to read the data and the updated file with marker short names write:

fcs\_data\_with\_info <- read\_data\_from\_folder(data\_path = paths$clean\_data\_path,   
 files\_to\_open = files\_to\_open)  
fcs\_data <- fcs\_data\_with\_info$fcs\_data  
file\_names <- fcs\_data\_with\_info$file\_names  
rm(fcs\_data\_with\_info)  
  
params\_fcs <- get\_params\_fcs\_data(fcs\_data[[1]])  
  
marker\_info <- read.csv2(fs::path(paths$clean\_data\_info\_path, "marker\_names\_included\_manual\_shortnames.csv"), stringsAsFactors = FALSE)

The variable fcs\_data now contains a list of matrixes with your data, while file\_names are a vector of names in the same order as the data are stored in fcs\_data.

# Marker gating

This section can be skipped if manual gates for the markers are not of interest, but it is needed for some markers if selected events should be used in further analysis. Once the manual gating is as you like it, it is not necessary to rerun. Hence the if sentence. If you still want to rerun just run the line within the if sentence.

The example only include gating for CD3, CD4, CD8 and CD45, all gates should be adjusted inside the file “marker gating all files.R” see “Vignette marker gating.docx” for explanation on how to make changes in the file. You might also change the gates to be equal for all files.

if(nrow(file.info(list.files(paths$marker\_gating\_results\_path)) ) == 0){  
 source(fs::path(paths$script\_path, "marker gating all files.R"))  
}

This produce a list of matrices saved in "…\_up\_gating.rds" that are in the same size as fcs\_data but which contains 0 (negative), 1 (positive or low positive) and 2 (high positive) for each event and each marker. The same folder also contains “mean\_gates.csv” which is a table of the mean gates for those markers gated, the other markers have gotten the gate NA.

# Run FlowSOM

run\_flowSOM(fcs\_data = fcs\_data, # clean data, not transformed  
 file\_names = file\_names, # file\_names in fcs\_data in same order as the data  
 included\_files = file\_names, # could be changed to only some files  
 n\_per\_file = 15000, # number of events to include per file  
 included\_markers = marker\_info$markers\_name, # use all markers, can be manually changed with a list of markers to use in analysis.  
 transformation = "arc\_sinh", # by now only option.   
 scaling\_flowSOM = TRUE, # scaling is recommanded, but will mean that all markers count as much regardless of size  
 k\_s = c(10, 15, 20), # number of clusters, k\_s has to be a number or a vector of numbers.  
 xdim = 10, # xdim \* ydim gives the number of nodes that FlowSOM will combined to k\_s clusters.  
 ydim = 10,   
 resultpath = paths$clean\_data\_flowSOM\_results\_path, # can be changed if new folder is wanted a new for different analysis, the folder have to exists before running function. recommand one folder per selected event analysis.  
 seed = 1234, # the seed ensure that the same events are chosen, and hence the same result are uptained next time the exactly same analysis are done  
 selectedEvents = "all", # if not "all" has to be the same name as a coloum-name in the matrices in posNeg  
 posNeg = NULL, # has to have same structure as fcs\_data, creat by....  
 make\_heatmap = TRUE, # need library ComplexHeatmap to make a heatmap.   
 heatmap\_cluster\_column = FALSE # if FALSE the column are not clusters and will appear in the same order.  
 )

this should be repeated for different seeds, and if wanted different selected events (e.g. only CD3, Cd45, CD8 positive)

# Run FlowSOM for selected events

To do this marker gating has to be performed and a posNeg.rds produced

posNeg <- readRDS( fs::path(paths$marker\_gating\_results\_path, "posNeg.rds"))  
  
posNeg <- extra\_column\_posneg(posNeg = posNeg, column\_name ="CD45CD3CD8", markers = c("CD45", "CD3", "CD8"), markers\_values = c(1,1,12))   
  
  
run\_flowSOM(fcs\_data = fcs\_data, # clean data, not transformed  
 file\_names = file\_names, # file\_names in fcs\_data in same order as the data  
 included\_files = file\_names, # could be changed to only some files  
 n\_per\_file = 5000, # number of events to include per file  
 included\_markers = marker\_info$markers\_name, # use all markers, can be manually changed with a list of markers to use in analysis.  
 transformation = "arc\_sinh", # by now only option.   
 scaling\_flowSOM = TRUE, # scaling is recommanded, but will mean that all markers count as much regardless of size  
 k\_s = c(10, 15, 20), # number of clusters, k\_s has to be a number or a vector of numbers.  
 xdim = 10, # xdim \* ydim gives the number of nodes that FlowSOM will combined to k\_s clusters.  
 ydim = 10,   
 resultpath = fs::path(paths$clean\_data\_flowSOM\_results\_path, "CD45CD3CD8"), # this folder must be generated before analysis  
 seed = 2134, # the seed ensure that the same events are chosen, and hence the same result are uptained next time the exactly same analysis are done  
 selectedEvents = "CD45CD3CD8", # if not "all" has to be the same name as a coloum-name in the matrices in posNeg  
 posNeg = posNeg, # has to have same structure as fcs\_data, creat by marker gating....  
 make\_heatmap = TRUE, # need library ComplexHeatmap to make a heatmap.   
 heatmap\_cluster\_column = FALSE # if FALSE the column are not clusters and will appear in the same order.  
)

# Markerplot

resultpath <- paths$clean\_data\_flowSOM\_results\_path # path to where the q5, q10, ... files from run\_flowSOM is saved.  
# if gates included in markerplots the csv file has to be made, colnames "X" "low" "high"  
# X has to be equal column marker\_short\_name in ...clean\_up\clean\_data\_info\marker\_names\_included\_manual\_shortnames.csv  
# low is first gate, high second gate, both can be NA  
# gates <- read.csv2(fs::path(paths$clean\_data\_posNeg\_path, "mean\_gates.csv"), stringsAsFactors = FALSE)  
  
k <- 10  
seed <- 1234  
selectedEvents <- "all"  
highlight\_cluster <- 4  
order\_marker\_shortname <- marker\_info$markers\_short\_name  
gates <- NULL  
  
tiff(fs::path(resultpath, paste0("markerplot\_k\_", k,"\_seed", seed, selectedEvents, ".tiff")), width = 1150, height = 900)  
marker\_plot(path = resultpath,   
 k = k,   
 seed = seed,   
 selectedEvents = selectedEvents,   
 highlight\_cluster = highlight\_cluster,   
 gates = gates,   
 order\_marker\_shortname = order\_marker\_shortname)  
dev.off()  
  
  
  
  
k <- 15  
seed <- 1234  
selectedEvents <- "all"  
highlight\_cluster <- NA  
order\_marker\_shortname <- marker\_info$markers\_short\_name  
gates <- NULL  
tiff(fs::path(resultpath, paste0("markerplot\_k\_", k,"\_seed", seed, selectedEvents, ".tiff")), width = 1150, height = 900)  
marker\_plot(path = resultpath,   
 k = k,   
 seed = seed,   
 selectedEvents = selectedEvents,   
 highlight\_cluster = highlight\_cluster,   
 gates = gates,   
 order\_marker\_shortname = order\_marker\_shortname)  
dev.off()  
  
  
  
k <- 20  
seed <- 1234  
selectedEvents <- "all"  
highlight\_cluster <- 1:20  
order\_marker\_shortname <- marker\_info$markers\_short\_name  
gates <- NULL  
tiff(fs::path(resultpath, paste0("markerplot\_k\_", k,"\_seed", seed, selectedEvents, ".tiff")), width = 1150, height = 900)  
marker\_plot(path = resultpath,   
 k = k,   
 seed = seed,   
 selectedEvents = selectedEvents,   
 highlight\_cluster = highlight\_cluster,   
 gates = gates,   
 order\_marker\_shortname = rker\_info$markers\_short\_name)  
dev.off()  
  
  
#  
k <- 10  
seed <- 2134  
selectedEvents <- "CD45CD3CD8"  
highlight\_cluster <- 4  
order\_marker\_shortname <- marker\_info$markers\_short\_name  
gates <- read.csv2(fs::path(paths$marker\_gating\_results\_path, "mean\_gates.csv"))  
resultpath <- fs::path(paths$clean\_data\_flowSOM\_results\_path, "CD45CD3CD8")  
  
tiff(fs::path(resultpath, paste0("markerplot\_k\_", k,"\_seed", seed, selectedEvents, ".tiff")), width = 1150, height = 900)  
marker\_plot(path = resultpath,   
 k = k,   
 seed = seed,   
 selectedEvents = selectedEvents,   
 highlight\_cluster = highlight\_cluster,   
 gates = gates,   
 order\_marker\_shortname = order\_marker\_shortname)  
dev.off()