How to Work with Flow Cytometry Data – Example using Automated Gating

TERMINOLOGY

We use the term **samples** to denote units such as patients, persons, study participants, animals, plants, cells, etc. In this tutorial, the term **variables** denote a set of features that encapsulates important properties of each sample. A variable can for example be the cell count or mean fluorescence intensity (MFI) of a cell population of interest. The term **data set** is used to describe measurement data, and is normally represented by a matrix where the columns corresponds to samples and the rows to variables.

INTRODUCTION

This is an accompanying document to "How to Work with Flow Cytometry Data", with a self-contained example based on automated gating. The automated gating is performed using R (https://www.r-project.org) and Bioconductor (https://bioconductor.org). Make sure you have at least version 3.3.1 of R. To run R, we recommend using RStudio (https://www.rstudio.com).

Automated gating enables extraction of features from flow cytometry data in a non-subjective way. This is especially important for high-dimensional data sets where it can be hard to decide upon gating strategies. The Probability Binning algorithm (Roederer, et. al. 2001, Rogers et al. 2008) is a well-established and simple tool (Perfetto et al. 2004, O'Neill et al. 2013) that can be used to obtain a signature or fingerprint for each flow cytometry sample in a data set. The algorithm automatically finds data dependent rectangle gates (bins) and constructs a signature from the number of events in each gate/bin.

Note that any automated gating algorithm that generates cell counts for synchronized populations across samples can be used instead. For a review of automated methods in flow cytometry, we refer to (O'Neill et al. 2013).

This example shows how you can try out visualization and statistics tools in Qlucore Omics Explorer on flow cytometry data for yourself.

*A comprehensive description of all functions can be found in the **Reference Manual** that is supplied in the **Help Menu** of Qlucore Omics Explorer. On www.qlucore.com you can find more documentation and watch instruction videos. Please register and log in to get access to the documents provided there.*

EXAMPLE DATA SET

In this document we use the same data set as in "How to Work with Flow Cytometry Data", but we extract features in a different way — with an automated gating procedure called probability binning. The data set has leukemic and normal samples, stained with five colors (Aghaeepour et al. 2013). The data is publicly available from FlowRepository (Spidlen J et al. 2012). Each sample has been analyzed with eight tubes, including isotope control. Here we look at data from tube 6.

R SCRIPTS

Start by downloading the R scripts *AMLdownload.R, AMLfingerprint.R* and *writeGedata2.R* from the Qlucore support pages — search for "flow cytometry" or ID 111 and "Convert from R to Qlucore data file format" or ID 60.

DATA DOWNLOAD

To download the first 100 samples from tube 6, open the script *AMLdownload.R* in your R session. On every line containing the keyword "CHANGE", you need to customize the input according to your preferences. Then run the entire script.

If you prefer you can download the data directly from FlowRepository (https://flowrepository.org/id/FR-FCM-ZZYA). Then make sure to select only samples corresponding to tube 6 (i.e. 0006.FCS, 0014.FCS, 0022.FCS, …).

FINGERPRINT GENERATION WITH AUTOMATED GATING

You can run the entire fingerprinting procedure through the script *AMLfingerprint.R* at once if you customize all lines with the keyword "CHANGE". This generates a file called *fingerprintAML_n100.gedata*. Another option is to follow the step-by-step explanation in the appendix of how to do the automated gating.

The features (variables) that are obtained are cell counts in 128 different bins.

LOAD DATA TO QLUCORE OMICS EXPLORER

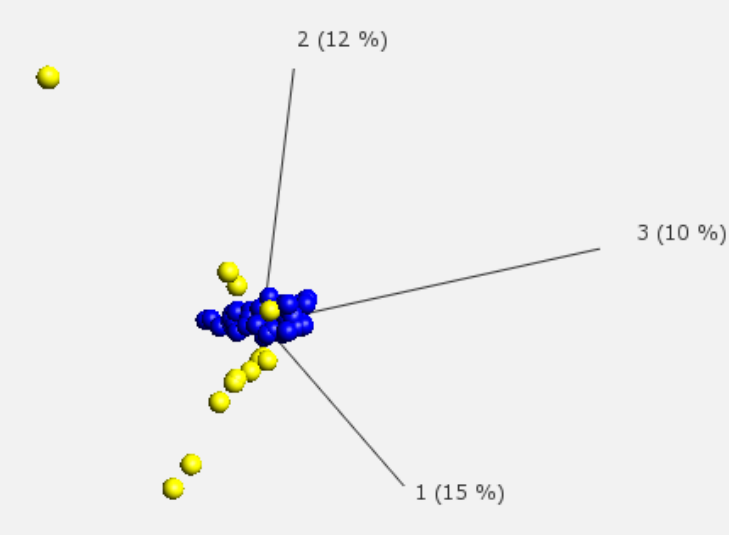
We can open the file fingerprintAML_n100.gedata directly in Qlucore **(File > Open).**

After loading the data, go to the **Method** tab and choose **Mean=0** under **Normalization**. For a discussion of which normalization to use and how to use log transforms, we refer to ”How to Work with Flow Cytometry Data A”.

The variables are cell counts in 128 different bins. If you go to the **Variables** dock window **(View > Dock Windows > Variables)** you can click on a variable in the ”Variable IDs” section to see relative ranges of the bins for each of the five fluorescence parameters and the scatter parameters.

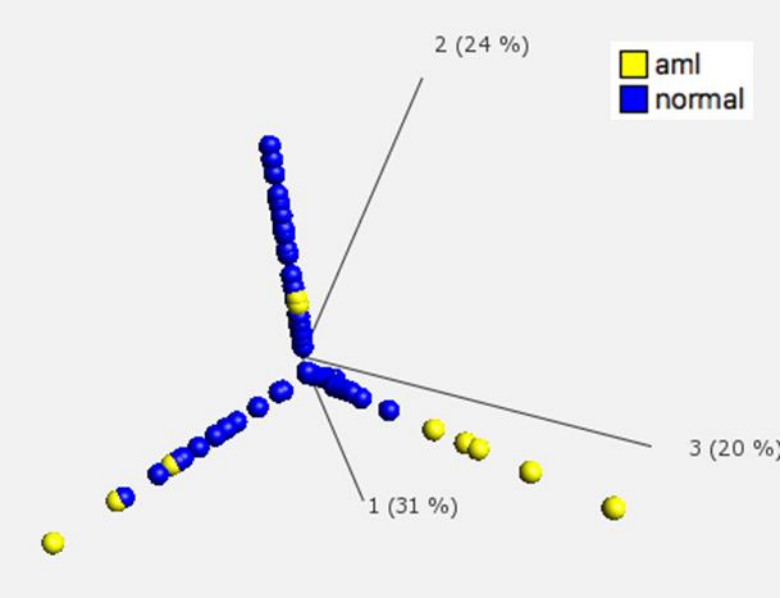
PRINCIPAL COMPONENT ANALYSIS (PCA)

In the **Method** tab choose **PCA** under **Plot Type**. In the Samples dock window, select the ”Condition” annotation and click on Color samples.



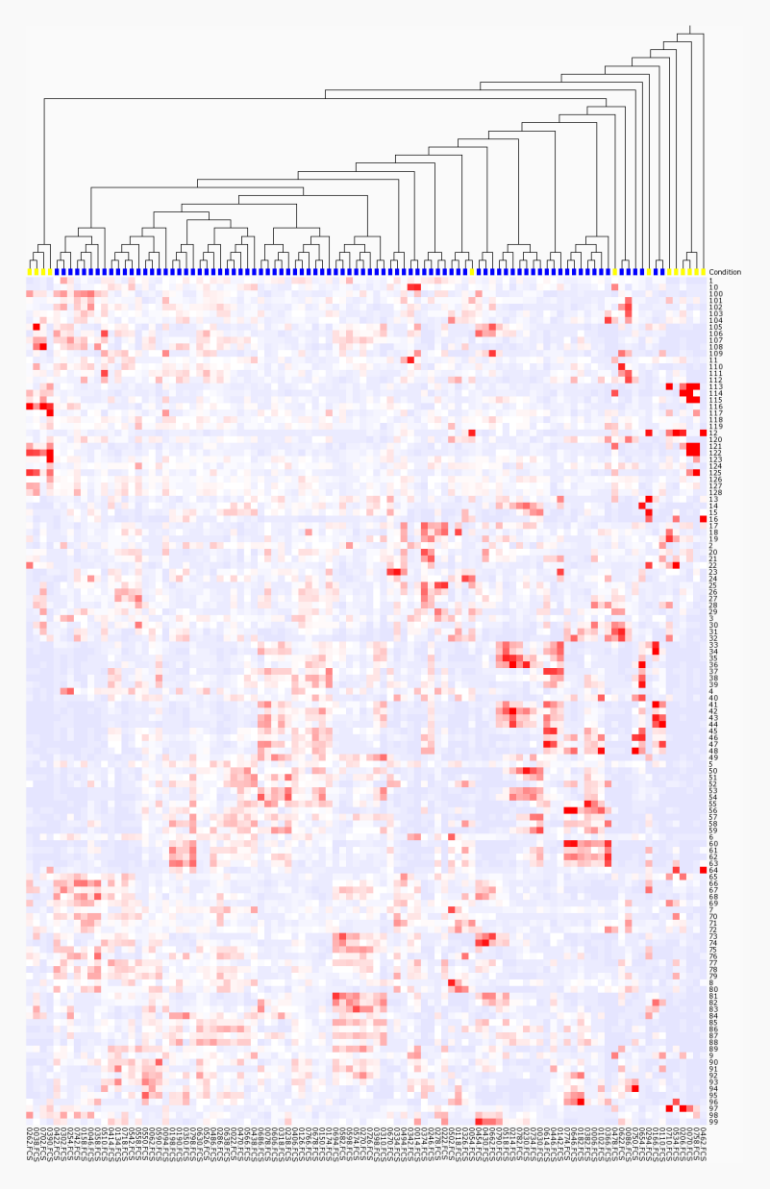
ISOMAP

Go to the Options tab. Under **Isomap**, click on **Set**. Isomap uses local distances between samples in the high-dimensional feature space to create a three-dimensional visualization.



HEAT MAP AND HIERARCHICAL CLUSTERING

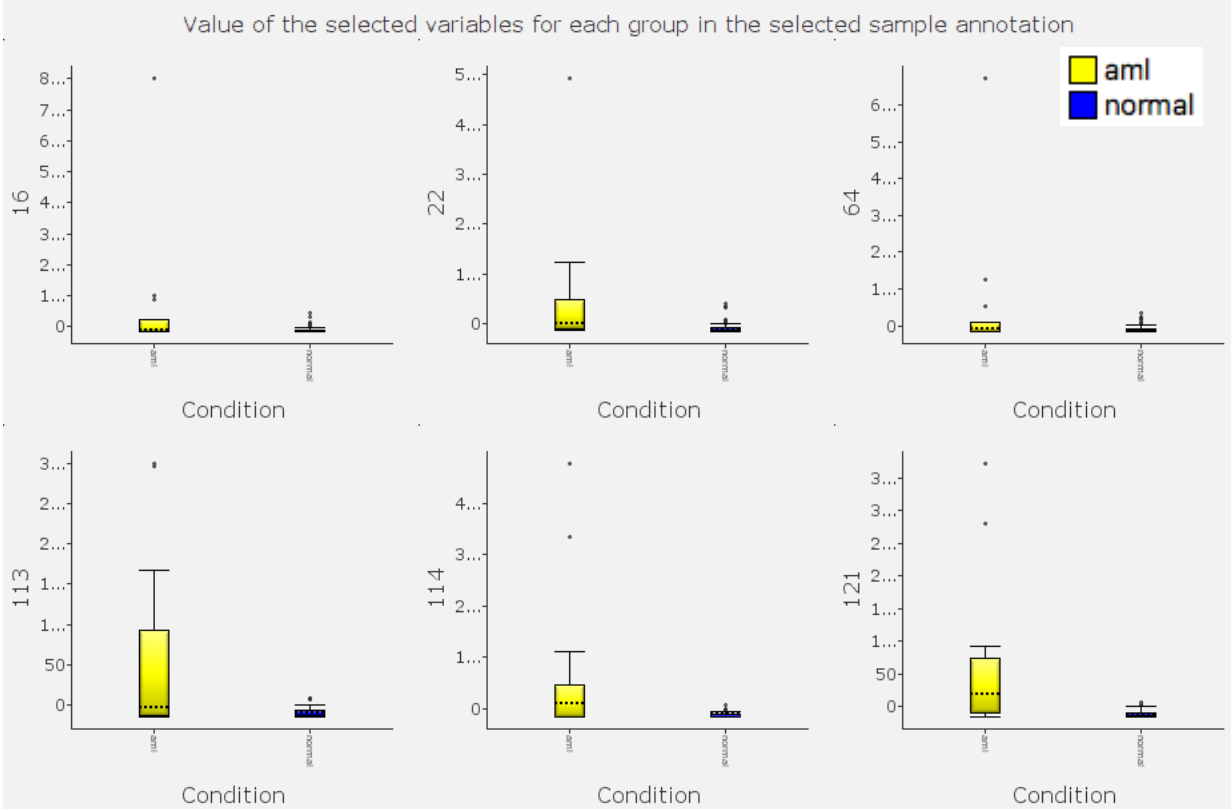
We can also visualize the difference between normal and AML samples in a heat map with hierarchical clustering. In the **Method** tab select **Heat** under Plot Type. Go to the **View** tab. Under **Order** set **Sample Order** to **Hierarchical Clustering**. To see which samples are annotated as AML samples, in the Color box set **Sample Color** to **By the annotation** — ”Condition”. The hierarchical clustering is visualized with a dendrogram shown above the heat map. Note that in each split in the dendrogram the ordering is arbitrary, i.e. which cluster that is to the left and which cluster that is to the right. You can swap this by selecting **Flip** under **Toolbox** and click on the junctions in the dendrogram.



BOX PLOTS

To further analyze how the counts vary between groups we can do box plots. In the **Method** tab, select **Box** under **Plot Type**. Under **Axis Data Selection** select ”Condition” for the X axis. Then choose ”Variables selected by the Y axis tool” for the Y axis and click on the file name *fingerprintAML_n100.gedata* in the **Variables** dock window **(View > Dock Windows > Variables).** Since we have more than 64 variables, the box plots are not directly shown, so we need to filter variables using the **Statistics** dock window **(View > Dock Windows > Statistics).** If we use the option ”Filter by Variance” we show only those variables that have the highest

variance across samples. Setting the slider at 0.5, we get six variables which have a variance which is has at least 0.5 times the variance of the most varying variable.



The Y axis legend shows the identification number of the bin whose count is shown. In the Variables dock window you can select an identification number to see the relative ranges of the fluorescence and scatter parameters for that bin.

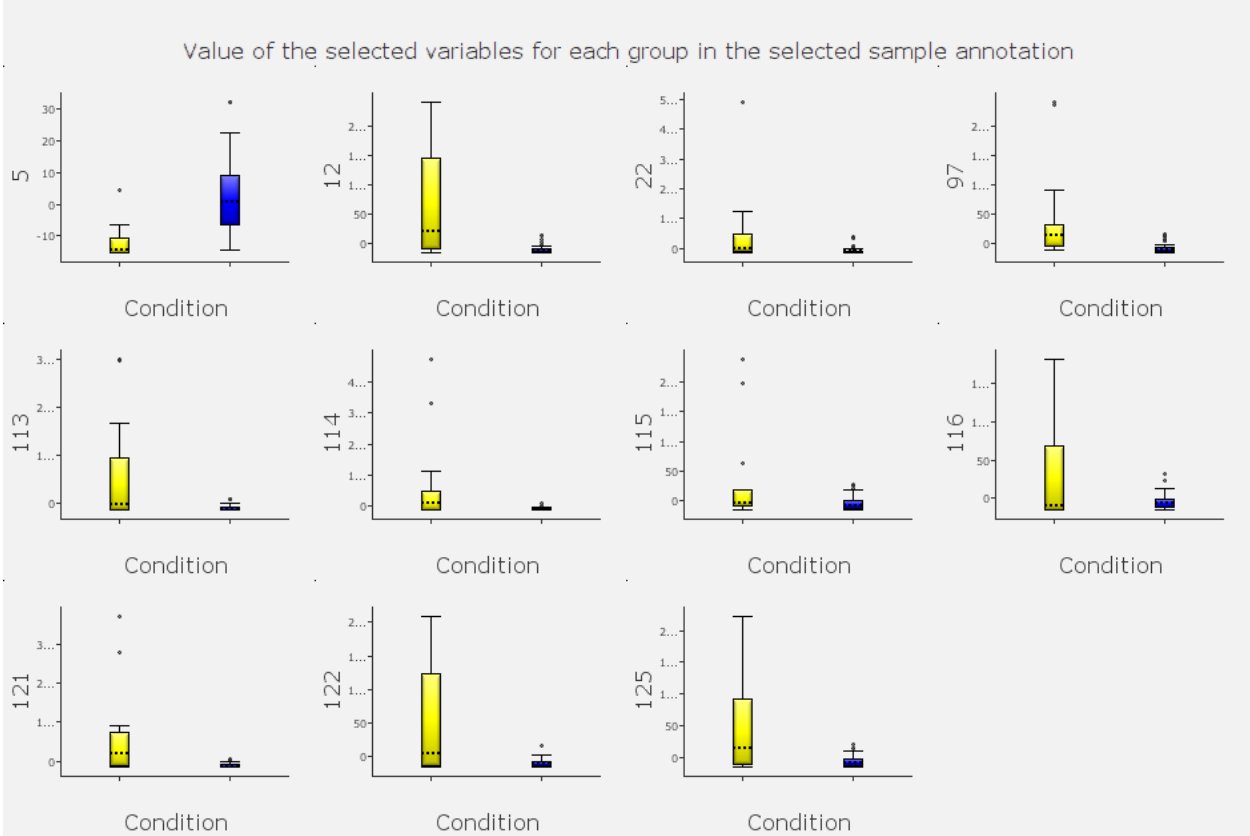
STATISTICAL TESTS

To only show variables that show statistically significant variation between the two groups, we can use the **Statistics** dock window to do a t-test. This can be combined with any of the plots described above.

Open the **Statistics** dock window through **View > Dock Windows > Statistics**. Select **Filter by Two Group Comparison.** To specify the annotation that distinguishes groups, for the AML data set we select ”Condition”. We choose significance level 0.0001 by setting **p = 0.0001**, meaning that all variables with a p-value below 0.0001 are displayed. At this significance level we have ten significant variables, this is shown below for the box plots. We can also drag the slide to continuously change the p-value bound.

Note: The p-value is not corrected for multiple testing. If you have a large number of features it makes sense to use the q-value instead since it compensates for multiple tests based on the False Discovery rate.

A comprehensive description of the statistical tests available in Qlucore Omics Explorer can be found in the **Reference Manual** in the **Help menu.**



APPENDIX

PACKAGE INSTALLATION

• You need the R Bioconductor packages flowCore and flowFP. To install them, open R and run the following lines of code:

```
source("http://bioconductor.org/biocLite.R")

biocLite('flowCore')
```

```
biocLite('flowFP')
```

LOADING PACKAGES AND FUNCTIONS

Load flowCore and flowFP by:

```
library(flowCore)

library(flowFP)
```

Set your working directory to the directory which contains the file writeGedata2.R and load functions from this script:

```
setwd("<<path to directory with writeGedata2.R>>")

source('writeGedata2.R')
```

LOAD DATA TO R

With flowCore we can read a .fcs file into a *flowFrame* and then combine flow frames into a *flowSet*. If you have a large data set, with many events per sample, and/or many samples, you might need to subsample the data if the computations takes too much time, this means that you randomly select a number of events per sample.

In R, first run:

```
dirpath <- "<<path to directory with the .fcs files>>"

fcs.file.names <- list.files(dirpath, pattern='.*\\.(FCS|fcs)')
```

To see which .fcs files you will load and how many files it is, run:

```
fcs.file.names

length(fcs.file.names)
```

Then we will load the data, either without subsampling:

```
frames <- lapply(file.path(dirpath, fcs.file.names), read.FCS,

                                    dataset = 1)
```

or with subsampling:

```
subsamp.size <- 2000

min.number.of.events <- 3000

frames <- lapply(file.path(dirpath, fcs.file.names), read.FCS,
```

```
                dataset = 1,

                    which.lines = sample(1:min.number.of.events,
        subsamp.size))
```

Set the parameter *min.number.of.events* to the smallest number of events in any of the .fcs files.

If your .fcs files contain more than one data set, for example if it contains both uncompensated and compensated data, you might want to change the dataset option. To see which markers the data set you have loaded have, run

```
frames[[1]]@parameters@data$desc
```

Now *frames* is a list where each entry has type *flowFrame*. We will combine the flow frames into a flow set, which can be used by the probability binning algorithm. To do this, run:

```
fs <- as(frames, "flowSet")
```

PROBABILITY BINNING

Now we have loaded the data to a flow set called *fs*. Applying the probability binning algorithm to generate a fingerprint is simple, load the *flowFP* package and run the algorithm by:

```
ffp <- flowFP(fs)

fingerprint <- counts(ffp)
```

There are additional options available for the *flowFP* command, you can use the command *help(flowFP)* to find information about these.

Now the variable *fingerprint* contains counts in each of the generated bins/gates for each sample. To find the boundaries for each bin (128 bins by default), we can use the command *binBoundary*:

```
binBoundary(ffp)
```

We want to store the bin boundaries in a data frame so that we can use them as annotations in Qlucore Omics Explorer. This can be done by running:

```
bin.annotations <- data.frame(id = 1:dim(fingerprint)[2])

for (p in parameters(ffp)) {

            lower.left <- sapply(
```

```
                binBoundary(ffp), function(boundary) {boundary@ll[p]})
        upper.right <- sapply(
                binBoundary(ffp), function(boundary) {boundary@ur[p]})
        levels_ <- sort(unique(c(lower.left, upper.right)))
        bin.annotations[p] <- paste0(
                as.numeric(factor(lower.left, levels = levels_)), '-',
                as.numeric(factor(upper.right, levels = levels_)), ' (',
                length(levels_), ')')
}
```

SAMPLE ANNOTATIONS

For the example data set there are sample annotations in a separate file AML.csv. However, it has annotations for all tubes and all samples. We only used the 100 first samples of tube 6, so we need to select the appropriate samples.

If you use another data set you can use the .fcs file names as annotation in R and read the rest separately from Qlucore Omics Explorer. To do this, run:

```
sample.annotations <- data.frame(file.name = sapply(
  frames, function(frame) frame@description$GUID))
```

For the AML example data set, we can instead load all meta data from the file 'AML.csv' by:

```
meta.data <- read.csv(file.path(dirpath, 'AML.csv'),
                stringsAsFactor = FALSE)
fcs.names <- sapply(frames, function(frame) frame@description$GUID)
sample.annotations <- meta.data[match(fcs.names, meta.data$FCS.file),]
```

EXPORT TO QLUCORE OMICS EXPLORER

From R we can export data as well as sample and variable annotations to a .gedata file which can be imported into Qlucore Omics Explorer. For this the function writeGedata in the script writeGedata2.R. To do this, run

```
source('writeGedata2.R')
```

```
resdir <- "<<path to directory where to put .gedata file>>"

writeGedata2(t(fingerprint), bin.annotations, t(sample.annotations),

        varAnnotsHeaders = names(bin.annotations),

        sampleAnnotsHeaders = names(sample.annotations),

        file.path(resdir, paste0("fingerprintAML_n",

        length(fs),".gedata")))
```

DISCLAIMER

The contents of this document are subject to revision without notice due to continuous progress in methodology, design, and manufacturing.

Qlucore shall have no liability for any error or damages of any kind resulting from the use of this document.

Qlucore Omics Explorer is only intended for research purposes.

REFERENCES

N. Aghaeepour et al. (2013). Critical assessment of automated flow cytometry data analysis techniques, Nat Methods 10, 228-238.

K. O'Neill et al. (2013). Flow Cytometry Bioinformatics. PLOS Computational Biology 9 (12), e1003365.

S. Perfetto et al. (2004). Seventeen-colour flow cytometry: unravelling the immune system. Nature Reviews Immunology 4, 648-655 .

M. Roederer et al. (2001). Probability Binning Comparison: A Metric for Quantitating Multivariate Distribution Differences, Cytometry 45, 47-55.

W. Rogers et al. (2008). Cytometric Fingerprinting: Quantitative Characterization of Multivariate Distributions, Cytometry Part A 73, 430-441.

J. Spidlen et al. (2012). FlowRepository - A Resource of Annotated Flow Cytometry Datasets Associated with Peer-reviewed Publications. Cytometry Part A 8, 727-731.