Multiparametric Flow Cytometry First Steps

Carine P. Beatrici, Fabricio A. B. Silva

September 26, 2018

1 Introduction

The multiparametric-flow-cytometry application is the program used to generate the figures in the paper "A high-throughput computational tool to analyze phenotypic modulations in multiparametric flow cytometry data". In this document the main functionalities of this program and associated scripts are presented.

The dependencies to compile and execute the multiparametric-flow-cytometry application are:

1. A Linux/MacOS system;
2. make
3. gawk
4. The gfortran compiler
5. The gnuplot application;
6. The R compiler;
7. The Bioconductor package flowcore;

2 Compiling

The first step is to download all the content from the github repository (https://github.com/CombinatorialCytometry/multiparametric-flow-cytometry) and make sure the computer has all the dependencies installed and working properly. To compile the code, open a Linux/MacOS terminal in the same folder where the files were saved. Then compile the code using "make":

```
> make
```

If the compilation goes well and there are no compilation errors, you should see a file named "multiparametric-flow-cytometry". To execute the program just type the following command in the linux terminal:

```
> ./multiparametric-flow-cytometry
```

To get the graphics with the colored dot-plots, we need to provide an input configuration file ("in.dat"). In the github repository we provide examples of this configuration file.

3 The Configuration File

To build the color dot plots, it is necessary to provide a configuration file. Here we show an example configuration file and explain how to edit it in order to perform different analysis on the data.

The input file must be named as "in.dat". One available example is shown below. This example corresponds to the “in.dat” file available in the github repository:

```
1   Sample_39.fcs
2   10
3   6
4   1 5 6 7 9 10 11 12 14 15
5   525 625 475 675 725 425
6   10.09  17.35  10000.0  46.76  44.37  13.47
7   100.00    1.40    0.00    0.00    0.00     0.00
8    18.22  100.00   11.49    0.00    0.00     0.00
9     6.07   28.50  100.00    0.00    0.00     0.00
10    2.80   15.42   20.11  100.00    0.00     0.00
11    0.00    0.41    0.00    6.86    100.00  14.49
12    0.00    0.00    0.00    0.00    6.54   100.00
13  0 0 0 0 0 0
```

This file is composed of 13 lines. It is important to keep the correct order of the input lines. We can group the input lines in eight items, organized by their function in the process.

```
1) name of the FCS file.........................Sample_39.fcs
2) total number of columns in the fcs file.......10
3) number of fluorescence channels...............6
4) columns to be copied into memory...................1 5 6 7 9 10 11 12 13 14 15
first the fsc, ssc and optional data
after the fluorescence channels
5) wavelength values for the channel colors......525 625 475 675 725 425
6) Background values............................10.09  17.35  10000.0  46.76  44.37  13.47
7) Compensation values (the full matrix).....................................
...100.00    1.40    0.00    0.00    0.00     0.00
....18.22  100.00   11.49    0.00    0.00     0.00
.....6.07   28.50  100.00    0.00    0.00     0.00
.....2.80   15.42   20.11  100.00    0.00     0.00
.....0.00    0.41    0.00    6.86    100.00  14.49
.....0.00    0.00    0.00    0.00    6.54   100.00
8) Fluorescence channels used in the color resultant calculation..0 0 0 0 0 0 0
```

The in.dat must provide these information in order, without any other text between lines.

1) The first line is the complete name of the Flow Cytometry Standard (FCS) file. The program will read this FCS file and analyze it.

2) The second line is the number of columns that the program will copy from the FCS file.

3) Number of fluorescence signals. For example, if you want to analyze seven fluorescence signals and five other parameters (time, etc.), the second line must have 12 and the third line must have 7.

4) The columns to be copied. Here you need to know which columns of the FCS file have the information you want to display in the color dotplot. Please list first general data columns (like event ID, time, morphology, etc) and then the fluorescence signal columns.

If you do not know which columns to choose, you can open the FCS file and read the first line or execute the multiparametric-flow-cytometry application. The program will show you the first line of the file with all the names of the columns for you to choose. Column counting starts at 1.

5) the wavelength values can be related or not to real filters values. The multiparametric-flow-cytometry application automatically distributes colors evenly along the hue circle, ordered according to the wavelength values informed.

6)The background labeling corresponds to the maximum fluorescence value of negative events. Any positive event for a given channel must have superior fluorescence intensity, when compared with the background reference value. This value must be in agreement with the data.

7) The conventional compensation table. The values must be between [0,100] and the value 100 is required when the same channel is considered for both dimensions.

8) To calculate the color tendency for all the channels, set this value as zeros (one zero for each color channel) or complete with the sequence according to the number of channels (e.g 1 2 3 4 5 6). By doing this, all the channels are taken into account to define the resultant color.

To calculate the resultant color only for positive events for a subset of channels, type the corresponding channel numbers followed by zeros, e.g. 2 4 0 0 0 0. In this case, only the positives events for both the second and fourth channels will be considered and will receive a color, all others are going to stay black. In this example, for resultant color calculation, the fluorescences of the second and forth channel will not be considered.

To calculate the resultant color for negative events in a set of channels, just put the number of channels as negative numbers, e.g 2 -4 0 0 0 0. Using this configuration, the events are going to receive a color if and only if they are positive for the second channel and negative for the fourth channel (single positive events). In this example, for resultant color calculation, the fluorescences of the second and forth channel will not be considered.

4 Example

When you run the multiparametric-flow-cytometry application, you obtain two output files: "result_file.eps", with the color dot plot, and "result_filebw.eps", with the trust values. Results for the Sample_39.fcs file shown in figure 1

Please make sure you rename output files before running new tests, otherwise the output files will be overwritten.

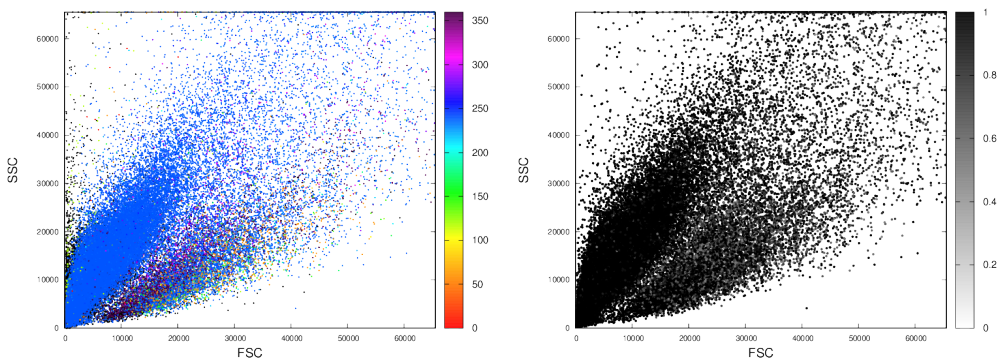

Figure 1: In the left, the color dot plot. In the right, the trust values (for 6 colors).

4.1 Selection of Positive Filters

For the flow cytometry analysis, it may be interesting to see not all the colors in the tendency plot at the same time. In some situations, when showing all the colors, the results may have low trust or even hide populations due to the high number of other cell types in the same region. To select only events with a positive signal for the channel 1 (FITC in this example) edit the file "in.dat". Change the line number 13 from "0 0 0 0 0 0" or "1 2 3 4 5 6" to "1 0 0 0 0 0". The results for this selection are presented in figure 2.

4.2 Selection of Negative Filters

For the flow cytometry analysis it may be interesting to see not all the colors in the tendency because the results may have low trust or even hide populations due to the high number of other cell types in the same region. To select only events with negative signal for the channel 1 (FITC in this example) edit the file "in.dat". Change the line number 13 from "0 0 0 0 0 0" or "1 2 3 4 5 6" to "-1 0 0 0 0 0". The results for this selection are presented in figure 3.

Comparing figures 2 and 3 we can see that the colored regions are different. In fact, all the dots colored in the figure 2 are now black. The only black dots in both figures are the negative events for all the fluorescence channels.

4.3 Selection of Filter Combinations

It is possible to select any combination of channels for positive or negative events. To select a combination of positive and negative values just edit the input file "in.dat" in line 13. For

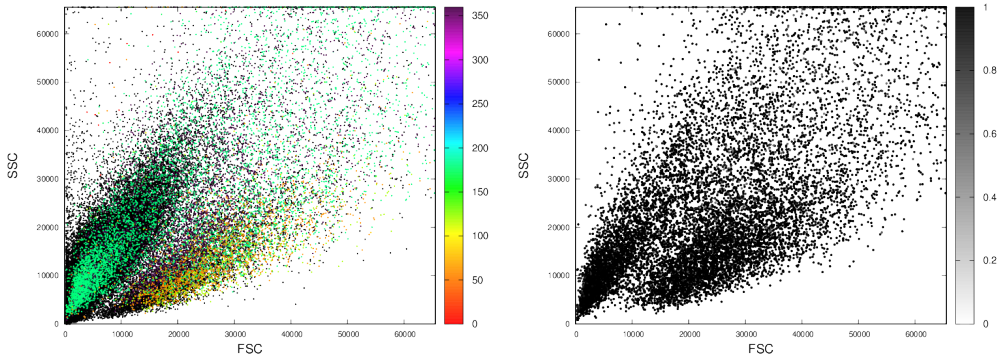

Figure 2: In the left, the color dot plot. In the right, the trust values. In this case we selected only the positive dots for the channel number 1 who corresponds to the FITC marker. Note that, in the trust dot plot, the white region suggests that those dots are not well defined by its color and they should not be taken into account.

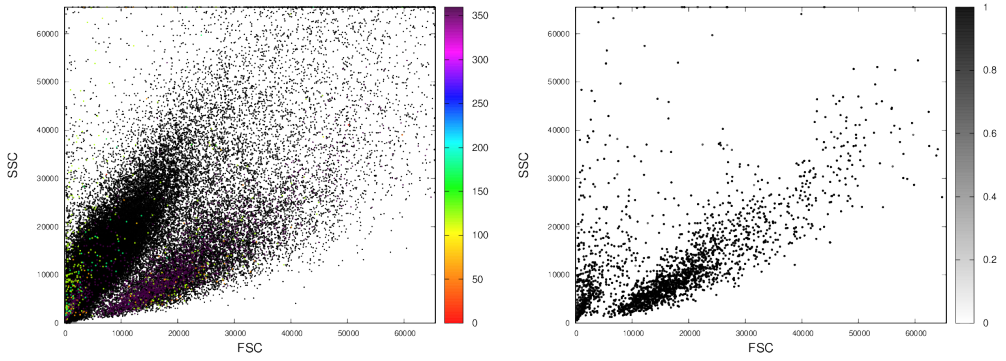

Figure 3: In the left the color dot plot, as described in section 4.2. In the right the trust values. In this case we selected only the negative dots for the channel number 1, which correspond to the FITC marker. Note the trust plot, the white region suggests that those dots are not well defined by its color and they should not be taken into account.

example, if you want to see the color tendency for all events with the second filter (fluorescence) positive and the fourth filter (fluorescence) negative, the line 13 must be like: "2 -4 0 0 0 0".

5 Gnuplot Scripts

The multiparametric-flow-cytometry application creates two temporary files (fluo.dat and nofl.dat) and images are generated via gnuplot by the two scripts provided (*script-gnu-color* and *script-gnu-bw*). The multiparametric-flow-cytometry application calls gnuplot automatically. The gnuplot scripts provided in the github repository were parametrized specifically for the example configuration file provided (in.dat). However, it is up to the user to edit the gnuplot scripts for new data.

To create the color dot plot, the script below reads the temporary files and plot them into the final eps images (see the *script_ gnu_ color* file).

```
1....set termoption enhanced
2....set title font ", 20"
3....set xlabel 'FSC'
4....set xlabel font ", 18"
5....set xtics font ", 11"
6....set ylabel font ", 18"
7....set ytics font ", 11"
8....set ylabel 'SSC' enhanced
9....set nokey
```

```
10...set cbrange [0:360]
11...set palette defined ( 0 '#ff0000',\
.....................1 '#ff7d00',\
.....................2 '#ffff00',\
.....................3 '#00ff00',\
.....................4 '#00ffff',\
.....................5 '#0000ff',\
.....................6 '#ff00ff',\
.....................7 '#440044')
12...plot [0:65533][0:65533] "nofl.dat" using 2:4:5 with points
     pt 7 ps 0.3 lc rgb "black"
13...set term postscript eps enhanced color
14...set output "result_file.eps"
15...replot "fluo.dat" using 2:4:5 with points palette pt 7 ps 0.3
```

1... Enable the color dot mode in gnuplot. Do not change this line.
2... Set the title font size.
3... Set the x axis label, the default is FSC.
4... Set the x axis label font size.
5... Set the tics label font size of the x axis.
6... Set the y axis label font size.
7... Set the tics label font size of the y axis.
8... Set the y axis label, the default is SSC.
11.. Define the palette, i.e. the colors of the graphic bar. The default values can be a visual approximation of the wavelength values.
12...The values inside the braces are the region plotted. For example, [0:1024][0:1024] will plot only the values between 0 and 1024 in both axes. The "nofl.dat" temporary file corresponds to events not associated with any color and are plotted in black.

After the word using there are tree numbers (2:4:5 in this example). These parameters are very important, since they define the data sources for the dot-plot. The first number is the column where the FSC information is located. The second number is where the SSC information is located. The third number is where the color is located.

To know where the data you want to plot are located in temporary files you have to check the in.dat configuration file. In this file, when you set the number of total columns and the number of fluorescence channels, temporary files store data accordingly. Therefore, in this example, in line 2, the total number of data columns equals 10 and, in line 3, we have 6 fluorescence channels, so there are 4 columns of other parameters (such as event ID, FSC, SSC). By doing this, resultant color results will be stored in the fifth column of both temporary files.

When you choose the columns in the configuration file (line number 4 of the in.dat file), you choose the order of the temporary files columns. If you keep the second column as the FSC and the fourth column as the SSC, there is no need to edit this script. However, it will depend on how you selected columns in the in.dat configuration file.

13 and 14 ..The lines 13 and 14 set the output format and file.
15... the same edition made in line 12 is required here. The "fluo.dat" temporary file corresponds to events associated with a color.

6 Resultant Filter Script - Hue Values

This script is used to filter resultant colors on the color dot plot generated by the multiparametric-flow-cytometry application based on hue values (0.0-360.0). In order to filter resultant colors, the user should run the *script_hue_filter* script in the same directory where the fluo.dat file, generated by the multiparametric-flow-cytometry application, is located. The user should inform the upper and lower limits of the filter, for instance:

Lower limit (0.0-360.0): 300
Upper limit (0.0-360.0): 340

The script generates an encapsulated postscript file (result_filter.eps) containing the new color plot generated when the filter is applied. See Figure 4 for an example of the application of the resultant filter in the same data used to generate Figure 1.

Since the *script_hue_filter* script is based on the script described in section 5, most remarks of that section are also valid for *script_hue_filter*.

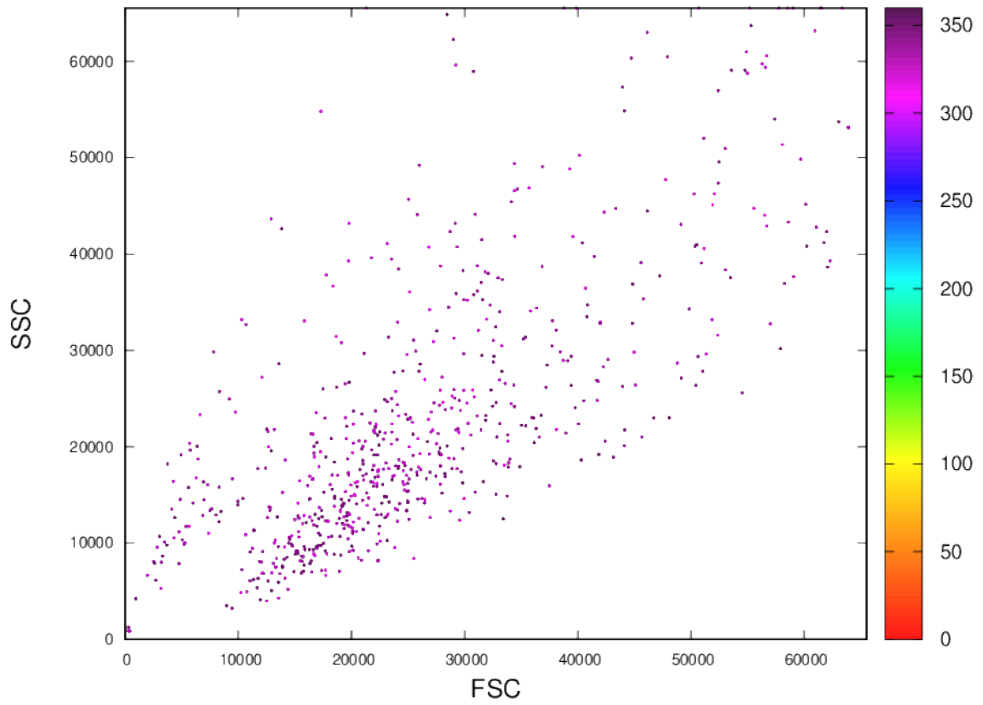

Figure 4: Color dot plot when the 300-340 resultant color filter is applied.

7 Resultant Filter Script - Trust Values

This script is used to filter resultant colors on the color dot plot generated by the multiparametric-flow-cytometry application based on trust values (0.0-1.0). In order to filter resultant colors, the user should run the *script_trust_filter* script in the same directory where the fluo.dat file, generated by the multiparametric-flow-cytometry application, is located. The user should inform the upper and lower limits of the filter, for instance:

Lower limit (0.0-1.0): 0.8
Upper limit (0.0-1.0): 1

The script generates an encapsulated postscript file (result_trust.eps) containing the new color plot generated when the filter is applied. See Figure 5 for an example of the application of the resultant filter in the same data used to generate Figure 1.

Since the *script_trust_filter* script is based on the script described in section 5, most remarks of that section are also valid for *script_trust_filter*.

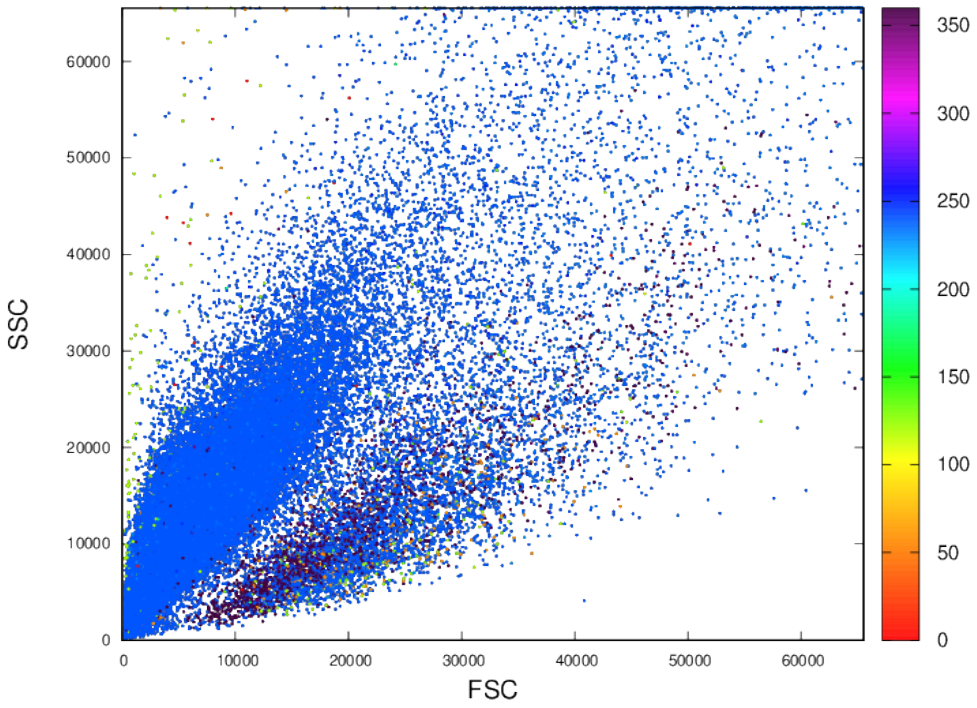

Figure 5: Color dot plot when the 0.8-1.0 trust value filter is applied.