TDS
Facility

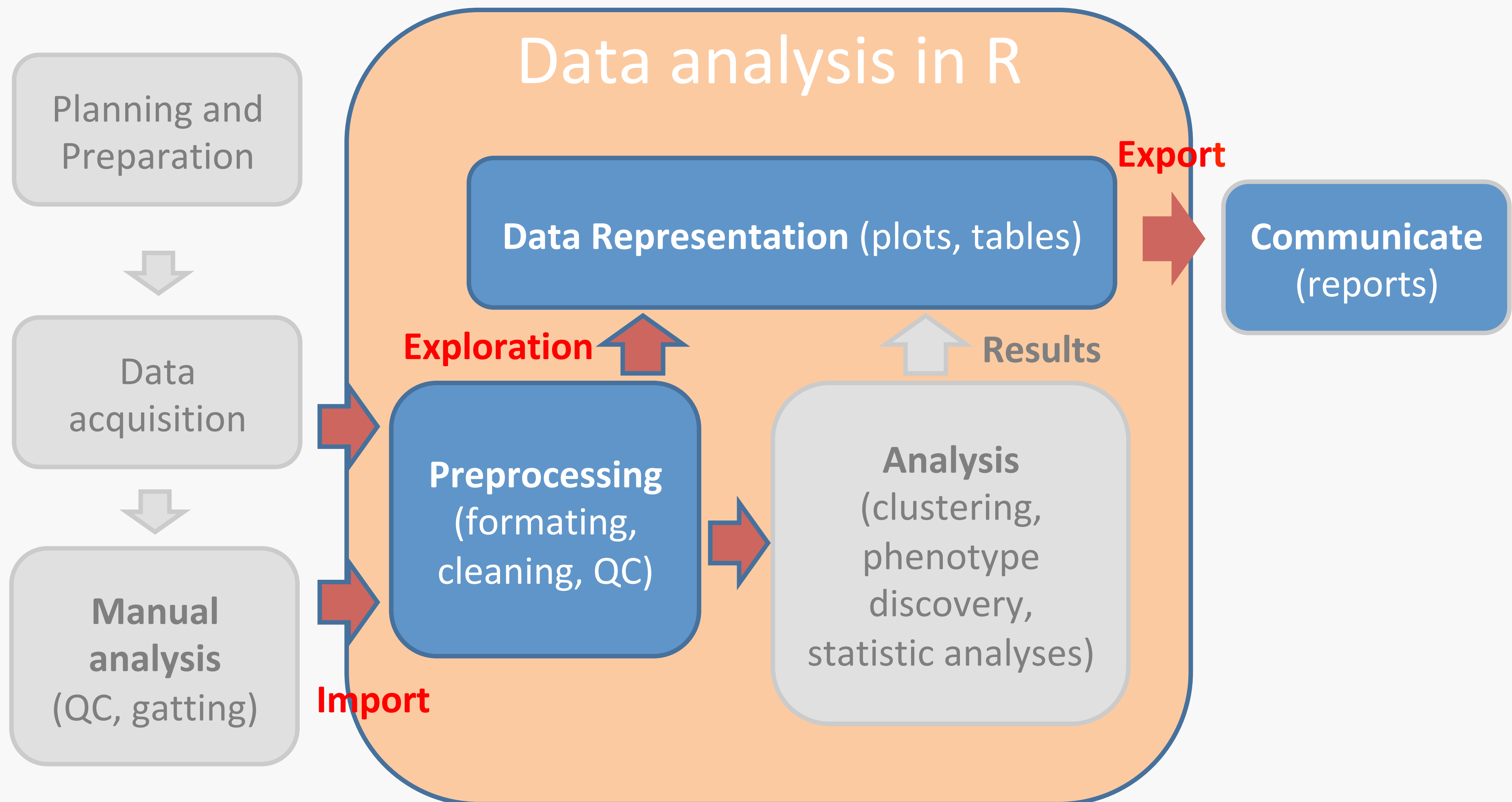
Introduction to R for flow cytometry data analysis Day 2

João Lourenço, Tania Wyss & Nadine Fournier

Translational Data Science – Facility

SIB Swiss Institute of Bioinformatics

Taking Advantage of R For Your Work



Outline

Day 2 (afternoon)

01 **Rmarkdown and report generation**

02 **Starting to work with flow cytometry data**

03

04

01 Rmarkdown and report generation

R Markdown

- Create documents with text *and to embed R code and display their outputs*
- Variation on **markdown** that *is specific to R* (**rmarkdown** package)
- R Markdown files have '.Rmd' extension.
- Needs two other packages:
  - **knitr** reads the code chunks, execute them, and 'knit' it back into the markdown document
  - **pandoc** converts the output into word/pdf/powerpoint etc

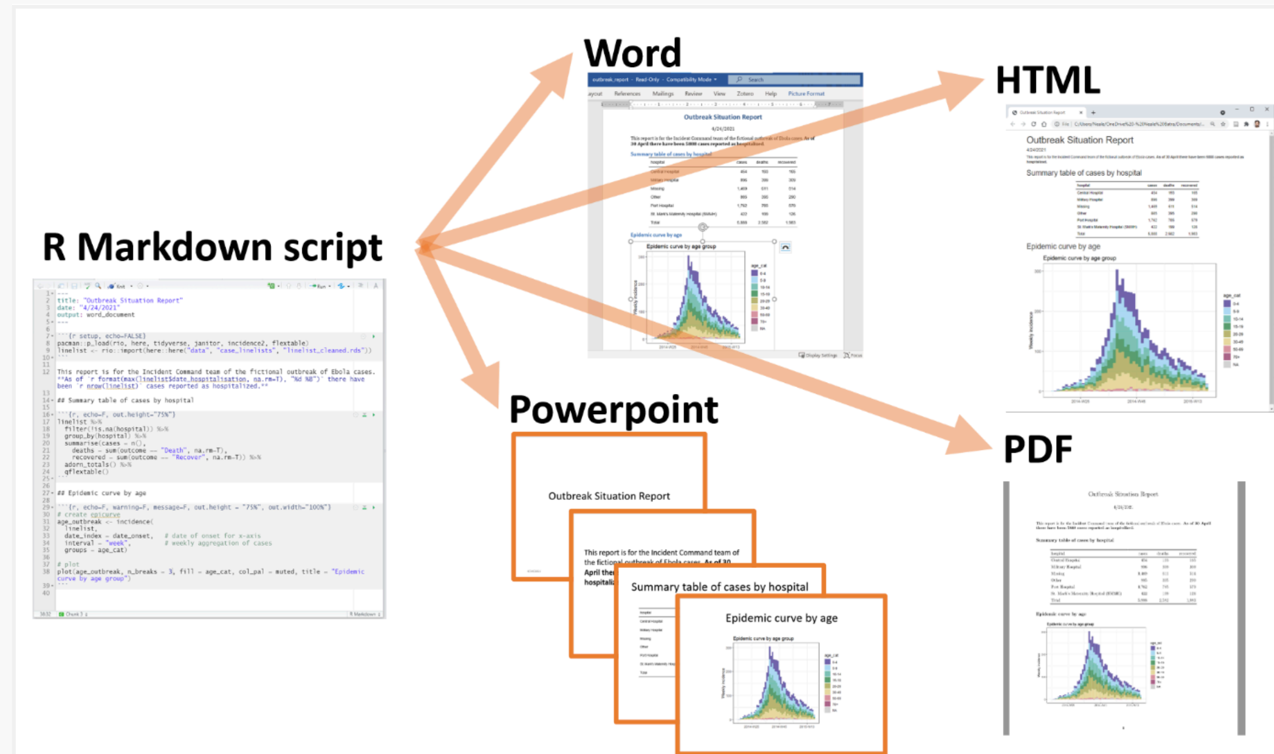
(source: https://rmarkdown.rstudio.com/authoring_quick_tour.html)

R Markdown

Why use R markdown?

- To **Create** efficient **reports** to summarize analyses
- To **Communicate** to an audience
- To ensure **reproducibility of results**

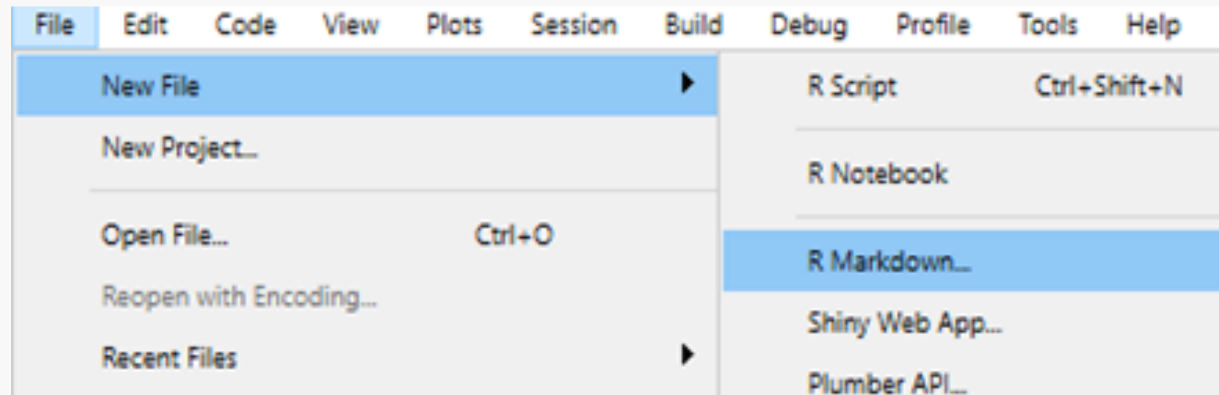
Different output file formats: HTML, PDF, …



(Source: https://epirhandbook.com/en/reports-with-r-markdown.html)

Create a new Rmd file

Start a new Rmd file: File > New File > R Markdown



Select some **options**:

- Output format (for example “HTML”)
- The title and the author names are not important.
- These options can be changed it in the script later

The working directory of a markdown file is wherever the Rmd file itself is saved

R Markdown elements

```
---
title: "R Markdown Example"
output: html_document
date: '2023-09-19'
---
```

Metadata (YAML)

```
```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```
```

Code chunk

```
## R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:
```

Text (markdown)

```
```{r cars}
summary(cars)
```
```

Code chunk

YAML metadata

```
1 ---
2 title: "R Markdown Example"
3 output: html_document
4 date: '2023-09-19'
5 ---
```

- This section must begin with a line containing just three dashes **---** and must close with a line containing just three dashes **---**
- Parameters come in **key:value pairs**
- The the key:value pairs are separated by **colons (not equals signs!**)
- The order of the parameters (if not indented) does not matter
- Basic parameters are pre-selected, but we can change them (for example, if we wish a ”**pdf_document**” instead of “**html_document**”)

Text

```
12
13  This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF,
    and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.
14
15  When you click the **Knit** button a document will be generated that includes both content as
    well as the output of any embedded R code chunks within the document. You can embed an R code
    chunk like this:
16
```

- Free text (like in a word document)
- **New lines**: enter two spaces*at the end of the previous line and then hit return.
- **Cases**:
  - Underscores (_text_) or single asterisk (*text*) to *italicise*
  - Double asterisks (**text**) for **bold text**
  - Back-ticks (`code`) to display text as code

Titles and headings

Different heading levels are established with different numbers of hash symbols at the start of a new line.

```
# First-level heading / title

## Second level heading

### Third-level heading
```

This is different than in a code chunk, in which hashes are used for commenting

Bullets and numering

Use asterisks (*) to created a bullets list.

Include a space between the asterisk and your bullet text

Finish the previous sentence, enter two spaces, hit return twice, and then start the bullets

```
Here are my bullets (there are two spaces after this colon):

* Bullet 1 (followed by two spaces and Enter/Return)
* Bullet 2 (followed by two spaces and Enter/Return)
  * Sub-bullet 1 (followed by two spaces and Enter/Return)
  * Sub-bullet 2 (followed by two spaces and Enter/Return)
```

Sub-bullets work the same way but are indented (tab)

After each bullet enter two spaces and then hit return

Numbers work the same way but instead of an asterisk, write 1), 2), etc.

Code chunks

- This is where you load packages, import data, and perform the actual data processing, analysis and visualisation
- There may be many code chunks

Each chunk is opened with a line that starts with three back-ticks, and curly brackets that contain parameters for the chunk ({ }).

```
16
17 ```{r cars}
18 summary(cars)
19 ```
20
```

The chunk ends with three more back-ticks

Create a new chunk:
- type the starting and ending lines yourself, or
- use the keyboard shortcut "Ctrl + Alt + i" (or Cmd + Shift + r in Mac), or
- click the green 'insert a new code chunk' icon

Code chunks options

- A chunk header must be written in *one line*
- Allways start with ‘r’
- After the ‘r’ you can **optionally** write a chunk “name” (**which has to be unique**).
- Other options are written as tag=value, such as:
    - eval = FALSE to not run the R code
    - echo = FALSE to not print the chunk’s R source code
    - warning = FALSE to not print warnings produced by the R code
    - message = FALSE to not print any messages produced by the R code
    - include = either TRUE/FALSE whether to include chunk outputs (e.g. plots) or not
    - fig.align = "center" adjust how a figure is aligned across the page

More about chunk options in https://yihui.org/knitr/options/

Global options

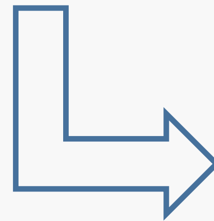
- Applied to all chunks in the script,
- Set up within the very first R code chunk
- For instance, so that only the outputs are shown for each code chunk and not the code itself, you can include this command in the R code chunk:

```
6
7 ```{r setup, include=FALSE}
8 knitr::opts_chunk$set(echo = TRUE)
9 ```
10
```

In-text R code

- Short R code commands within back-ticks
- Within the back-ticks, begin the code with “r” and a space, so RStudio knows to evaluate the code as R code
- Example:

```
16
17 ```{r cars}
18 summary(cars)
19 ```
20
21 The dimension of the cars dataset is `r dim(cars)`
```

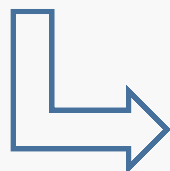


```
##      speed           dist       
##  Min.   : 4.0   Min.   :  2.00  
##  1st Qu.:12.0   1st Qu.: 26.00  
##  Median :15.0   Median : 36.00  
##  Mean   :15.4   Mean   : 42.98  
##  3rd Qu.:19.0   3rd Qu.: 56.00  
##  Max.   :25.0   Max.   :120.00  
```

The dimension of the cars dataset is 50, 2

Producing the document

- **Knitting** a file is how we generate the output file from the R markdown file
- RStudio will you show the progress within an ‘R Markdown’ tab near your R console
- The document will automatically open when complete
- **The document will be saved in the same folder as your R markdown script, and with the same file name**



R Markdown Example

2023-09-19

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
summary(cars)
```

```
##      speed           dist       
##  Min.   : 4.0   Min.   :  2.00  
##  1st Qu.:12.0   1st Qu.: 26.00  
##  Median :15.0   Median : 36.00  
##  Mean   :15.4   Mean   : 42.98  
##  3rd Qu.:19.0   3rd Qu.: 56.00  
##  Max.   :25.0   Max.   :120.00  
```

The dimension of the cars dataset is 50, 2

Let’s practice – 9

1) Create a new Rmd file with the following options
   - Title: «Let’s practice - 9»
   - Author: your name
   - Select the «use current date when rendering object» option
   - Default output format: HTML

2) We will repeat exercise 7, but this time by creating a report:
   - Import the data from the file clinical_data_mod.csv , convert gender and response to treatment to factors and compute the BMI of patients in a separate code chunk called “prepare_data”. **Change the chunk options so that code will not appear in the output.**
   - Create a code chunk for each plot. **Make sure the plot is centered**.
   - Add a headline (highest level) before each plot with some suggestive plot title

3) Save the Rmd file and produce the html document by «knitting» it.

02

Starting to work with flow cytometry data

Example of fow cytometry dataset

frontiers
in Immunology

METHODS
published: 19 November 2021
doi: 10.3389/fimmu.2021.768113

How to Prepare Spectral Flow Cytometry Datasets for High Dimensional Data Analysis: A Practical Workflow

Hannah den Braanker[1,2,3†], Margot Bongenaar[1,2†] and Erik Lubberts[1,2]*

[1] Department of Rheumatology, Erasmus University Medical Center, Rotterdam, Netherlands, [2] Department of Immunology, Erasmus University Medical Center, Rotterdam, Netherlands, [3] Department of Clinical Immunology and Rheumatology, Maasstad Hospital, Rotterdam, Netherlands

Example of fow cytometry dataset

- Publicly available through the **FlowRepository database** at https://flowrepository.org/, using repository ID FR-FCM-Z4KT
- Data from **31-color spectral flow cytometry** on peripheral blood mononuclear cells (**PBMCs**) from healthy controls
- Data were acquired and unmixed using SpectroFlo® v2.2.0.3 software (Cytek Biosciences, Fremont, California, USA)
- **Resulting unmixed fcs files were analyzed using manual gating** in FlowJo v10.7 software (BD Biosciences, San Jose, California, USA) according

Flow Cytometry Standard (FCS) files

- Data standard for the reading and writing of data from flow cytometry experiments
- File exported from the cytometer’s acquisition software
- Versions: FCS1.o (1984), FCS 2.0 (1990), FCS 3.0 (1997), FCS 3.1 (2010),
- File Format (main segments):
  - HEADER segment (ASCII text): version, …
  - TEXT segment (ASCII text): keywords and values which describe the data format and encoding
  - DATA segment (binary): contains the actual measurements
  - Others …

Data structure

- Array (matrix) with fluorescence and scatter channels represented in columns and individual «events» (cells…) forming the rows

Channels

Events

	FSC-A	FSC-H	SSC-A	B515-A	R780-A	R710-A	R660-A	V800-A	V655-A
[1,]	27700.75	27291.75	177.52585	1984.485	625.0796	1232.1008	748.5101	1553.0295	1350.2565
[2,]	41264.25	39764.25	320.12296	3639.620	539.7032	1433.3112	1470.2659	2217.6750	2305.3516
[3,]	65054.75	57606.25	203.01607	2191.861	198.6541	726.9798	766.2198	802.2521	809.9579
[4,]	30584.00	31664.50	130.68690	1873.409	1304.0895	2528.7083	784.6980	1702.3671	1185.8608
[5,]	39505.75	39626.00	203.25166	2540.620	323.2625	857.1525	715.0004	1117.4775	1746.5798
[6,]	33171.50	34794.00	333.64246	2192.864	1408.8563	2573.5095	1604.2236	2128.1748	1727.5891
[7,]	63711.00	54475.50	1122.48340	3879.044	1730.8085	3573.5652	1691.8744	5106.0596	3578.0332
[8,]	40000.75	40213.50	236.54262	2545.858	1081.6753	2313.5962	1411.0983	2989.7524	1920.4047
[9,]	49286.00	49182.50	78.61845	1601.092	123.2834	493.6364	242.0255	633.3533	759.2227
[10,]	32209.75	33368.25	203.29897	2387.361	1056.0723	1769.4005	939.7758	1693.8635	1579.7000
[11,]	35937.25	36212.50	220.66580	2901.591	1218.1395	3202.3853	1059.7604	2443.0205	2253.0146
[12,]	32905.50	33897.50	233.98033	2726.240	1952.0721	3405.7139	2726.1091	2988.6882	2011.0159
[13,]	36028.50	35845.50	219.18674	3221.668	2542.3389	3895.0371	2283.0444	3331.8298	2479.6580
[14,]	38616.00	38775.00	218.46669	3218.305	582.6801	1022.7971	1255.5858	2150.4185	1993.2681
[15,]	45282.25	42223.25	1173.74487	6941.545	705.4651	1649.9570	1615.0811	4287.2036	3778.2302
[16,]	36246.25	36207.75	189.15569	3049.417	1736.7826	2823.7266	1031.0308	2824.6582	2053.6843
[17,]	29282.75	29884.00	209.64102	1836.197	612.2673	1149.7164	870.3303	1720.2170	1525.6914
[18,]	57757.25	54448.25	1999.17517	12972.877	4364.5908	11298.7070	6745.5039	20934.3457	17057.1934
[19,]	33301.00	33093.50	208.47151	2146.622	429.5022	855.5981	845.9418	1207.8969	1297.2683
[20,]	34478.25	35390.75	211.26921	3060.585	2016.3651	3442.5408	1348.4852	2673.9729	2259.8494
[21,]	29406.25	28219.50	231.55798	3008.380	997.8875	2319.5779	1514.2091	1757.2463	1675.9983
[22,]	49978.50	48517.75	537.04224	3122.343	981.1232	2252.1189	1861.3472	2518.4731	2230.5327
[23,]	39872.50	37620.75	198.75706	2719.222	1657.0939	2945.5713	1025.1293	2203.0527	1670.1367
[24,]	33395.00	35331.75	220.46056	2664.632	690.1926	1483.0898	1736.9537	1397.0316	1982.9124
[25,]	46976.00	47355.25	231.33037	2530.461	537.1376	1194.0681	1072.7083	1531.7494	1766.5841
[26,]	56663.75	51458.25	223.06416	3217.866	398.6222	1279.4880	1207.4561	1268.9905	1553.6884
[27,]	50818.75	48556.25	305.77182	3714.351	577.0732	1364.4095	1064.0983	1633.2513	2077.0466
[28,]	36225.25	36196.75	180.30524	2636.466	946.7570	2138.4143	1695.0502	1807.8429	2057.7292
[29,]	28509.25	30715.50	230.27397	1072.201	1867.2009	1643.1423	882.4811	1201.5806	688.1475
[30,]	37198.75	36200.50	237.67776	3046.719	1376.3452	2580.9287	1326.2197	2599.6101	2196.7258

flowCore R Package

- https://bioconductor.org/packages/release/bioc/html/flowCore.html
- Provides data structures and basic functions to deal with flow cytometry data in R
- Installation:

```
if (!require("BiocManager", quietly = TRUE))
    install.packages("BiocManager")

BiocManager::install("flowCore")
```

- Vignette https://bioconductor.org/packages/release/bioc/vignettes/flowCore/inst/doc/HowTo-flowCore.pdf

Reading an FCS file into a *flowFrame*

- *A* ***flowFrame*** is the basic unit of manipulation
- Corresponds to a single FCS file

The function **read.FCS()** allows to read a single FCS file into R. Example:

```
> FCS_file <- read.FCS(
             filename ="T_cells_REU270_alive_T_cells.fcs",
             transformation=FALSE,
            truncate_max_range = FALSE)
```

- Important arguments:
  - **filename** is the file path
  - **transformation** specifies the type of transformation to be applied. When set to FALSE, no transformation is applied.
  - **truncate_max_range**. Set to FALSE to avoid truncating the extreme positive value to the instrument measurement range.

What is a flowFrame object?

```
> help(flowFrame)
```

flowFrame-class {flowCore} R Documentation

'flowFrame': a class for storing observed quantitative properties for a population of cells from a FACS run

Description

This class represents the data contained in a FCS file or similar data structure. There are three parts of the data:

1. a numeric matrix of the raw measurement values with `rows=events` and `columns=parameters`
2. annotation for the parameters (e.g., the measurement channels, stains, dynamic range)
3. additional annotation provided through keywords in the FCS file

Details

Objects of class `flowFrame` can be used to hold arbitrary data of cell populations, acquired in flow-cytometry.

What is a flowFrame object ?

- In R, objects such as flowFrames are **collections of data (variables)** and **methods (functions).**
- They belong to a given **class (**a blueprint for that object)
- Member variables in R objects are called **slots**. There are three slots in a flowFrame: *exprs*, *parameters* and *description*

Slots

`exprs`

Object of class `matrix` containing the measured intensities. Rows correspond to cells, columns to the different measurement channels. The `colnames` attribute of the matrix is supposed to hold the names or identifiers for the channels. The `rownames` attribute would usually not be set.

`parameters`

An `AnnotatedDataFrame` containing information about each column of the `flowFrame`. This will generally be filled in by `read.FCS` or similar functions using data from the FCS keywords describing the parameters.

`description`

A list containing the meta data included in the FCS file.

Summarize a *flowFrame*

```
> FCS_file
```

```
flowFrame object 'T_cells_REU270_alive_T cells.fcs'
with 315735 cells and 39 observables:
                      name       desc    range minRange maxRange
$P1                  FSC-A         NA  4194304        0  4194303
$P2                  FSC-H         NA  4194304        0  4194303
$P3                  SSC-A         NA  4194304        0  4194303
$P4                SSC-B-A         NA  4194304        0  4194303
$P5                SSC-B-H         NA  4194304        0  4194303
...                    ...        ...      ...      ...      ...
$P35 FJComp-PerCP-eFluor ..     CD127   100000     -111    99999
$P36 FJComp-Spark Blue 55..       CD3   100000        0    99999
$P37     FJComp-Zombie UV-A Zombie UV   100000     -111    99999
$P38    FJComp-eFluor 660-A    CTLA-4   100000     -111    99999
$P39                   Time        NA      166        0      165
278 keywords are stored in the 'description' slot
```

```
> summarize(FCS_file)
```

```
            FSC-A     FSC-H    SSC-A   SSC-B-A   SSC-B-H    SSC-H FJComp-AF-A
Min.     248252.4  200055.0 104527.1  54734.91   43496.0  71728.0  -85312.109
1st Qu.  694239.2  544927.0 371822.5 216510.48  163184.0 290444.0  -11530.027
Median   794500.6  628644.0 453873.5 263345.84  195338.0 348003.0   -7938.739
Mean     809134.7  639853.2 452449.1 263406.04  195808.8 347746.4   -8392.375
3rd Qu.  908499.2  722281.5 527944.0 307300.50  226672.5 402455.0   -4571.343
Max.    1608623.4 1358182.0 946566.2 608151.94  449530.0 791039.0   37643.547
```

…

```
     Time
  0.00000
 39.13176
 81.05503
 81.00381
121.98465
162.51257
```

Access data elements in a *flowFrame*

- To access data: use the @ operator or a method (function)
- Matrix of expression values (as a matrix)

```
> FCS_file@exprs
```
or
```
> exprs(FCS_file)
```

```
        FSC-A  FSC-H    SSC-A    SSC-B-A SSC-B-H  SSC-H FJComp-AF-A FJComp-APC-A
[1,] 708579.4 593958 331966.4 195681.8  161726 273584  -12322.742   -4990.1958
[2,] 587231.9 489906 323881.8 209247.5  165442 265458  -10672.745   -5642.0508
[3,] 828618.7 662813 487978.5 289251.3  215334 379895   -1366.873   -3940.6289
[4,] 733458.1 606898 447868.5 242895.0  188230 357695   -2092.956    -998.5401
[5,] 576551.5 461784 428876.1 238000.4  175819 326038   -6251.983   -5225.1035
[6,] 762848.1 606807 583804.5 344976.0  251346 444231  -10864.361   -4390.9263
```

- Metadata (as a data frame)

```
> pData(FCS_file@parameters)
```
or
```
> pData(parameters(FCS_file))
```

```
                         name        desc   range minRange maxRange
$P1                     FSC-A          NA 4194304        0  4194303
$P2                     FSC-H          NA 4194304        0  4194303
...                       ...         ...     ...      ...      ...
$P35 FJComp-PerCP-eFluor ..         CD127  100000     -111    99999
$P36 FJComp-Spark Blue 55..           CD3  100000        0    99999
$P37       FJComp-Zombie UV-A Zombie UV  100000     -111    99999
$P38      FJComp-eFluor 660-A    CTLA-4  100000     -111    99999
$P39                     Time          NA     166        0      165
```

How to replace the channel names by the antigen names in the expression matrix

- Copy the metadata to a data frame

```
> panel <- pData(FCS_file@parameters)
```

- Copy the names to a new column

```
> pData(FCS_file@parameters)$channel <- panel$name
```

- Replace the names by the antigens

```
> colnames(FCS_file)[!is.na(panel$desc)] <- panel$desc[!
  is.na(panel$desc)]
```

```
> head(exprs(FCS_file)[,10:15])
```

```
           CD27       LAG-3     CD25     CD49b        CD8        CD4
[1,] 34010.4844  -726.74323 2337.454  622.7443 -1674.8558 54145.7031
[2,] 26705.5781  -447.67514 3196.554 1380.7151 -1855.1270 64054.7617
[3,]   846.9209   246.76016 1000.591  581.6843  -977.6837  -219.4745
[4,]  1110.7271  -507.84625 1215.742 1224.3079  1438.7726 -2148.8167
[5,]  3685.3149 -1773.50989 4398.276  818.2174 -1662.8772 88996.8984
[6,]   505.4961   -21.17858 2401.317 -358.6945   451.0627 -2950.0181
```

Reading a list of FCS files into a *flowSet*

- *A **flowSet*** is a collection of *flowFrame*
- Convenient way to apply methods to all *flowFrame* simultaneously

The function **read.flowSet()** allows to read several FCS files in a given directory.

Example:

```
> fcs_data <- read.flowSet(path=“./”, pattern="*.fcs",
                           transformation = FALSE,
                           truncate_max_range = FALSE)
```

- Important arguments:
  - **path** is the path to the directory containing the FCS files
  - **pattern** sets which files to read (* is a wildcard replacing the file names)

You can coerce a list of *flowFrames* into a *FlowSet*, but is less convenient

Slots in a *flowSet*

> help(flowSet)

flowSet-class {flowCore} R Documentation

'flowSet': a class for storing flow cytometry raw data from quantitative cell-based assays

Description

This class is a container for a set of `flowFrame` objects

Slots

`frames`

An `environment` containing one or more `flowFrame` objects.

`phenoData`

An `AnnotatedDataFrame` containing the phenotypic data for the whole data set. Each row corresponds to one of the `flowFrame`s in the `frames` slot. The `sampleNames` of `phenoData` (see below) must match the names of the `flowFrame` in the `frames` environment.

List sample names

```
> sampleNames(fcs_data)
```

```
 [1] "T_cells_REU267_alive_T_cells.fcs"           "T_cells_REU268_alive_T_cells.fcs"
 [3] "T_cells_REU269_alive_T_cells.fcs"           "T_cells_REU270_alive_T_cells.fcs"
 [5] "T_cells_REU271_12_july_alive_T_cells.fcs"   "T_cells_REU271_13_april_alive_T_cells.fcs"
 [7] "T_cells_REU271_14_april_alive_T_cells.fcs" "T_cells_REU271_7_apr_alive_T_cells.fcs"
 [9] "T_cells_REU271_9_april_alive_T_cells.fcs"   "T_cells_REU271_alive_T_cells.fcs"
[11] "T_cells_REU272_12_july_alive_T_cells.fcs"   "T_cells_REU272_13_april_alive_T_cells.fcs"
[13] "T_cells_REU272_14_april_alive_T_cells.fcs" "T_cells_REU272_7_apr_alive_T_cells.fcs"
[15] "T_cells_REU272_9_april_alive_T_cells.fcs"   "T_cells_REU272_alive_T_cells.fcs"
```

Phenotypic data

- Extract / replace the data frame (or columns thereof) containing actual phenotypic information from the phenoData slot

```
> pData(fcs_data)
```

```
                                                                         name
T_cells_REU267_alive_T_cells.fcs                     T_cells_REU267_alive_T_cells.fcs
T_cells_REU268_alive_T_cells.fcs                     T_cells_REU268_alive_T_cells.fcs
T_cells_REU269_alive_T_cells.fcs                     T_cells_REU269_alive_T_cells.fcs
T_cells_REU270_alive_T_cells.fcs                     T_cells_REU270_alive_T_cells.fcs
T_cells_REU271_12_july_alive_T_cells.fcs     T_cells_REU271_12_july_alive_T_cells.fcs
T_cells_REU271_13_april_alive_T_cells.fcs T_cells_REU271_13_april_alive_T_cells.fcs
T_cells_REU271_14_april_alive_T_cells.fcs T_cells_REU271_14_april_alive_T_cells.fcs
T_cells_REU271_7_apr_alive_T_cells.fcs         T_cells_REU271_7_apr_alive_T_cells.fcs
T_cells_REU271_9_april_alive_T_cells.fcs     T_cells_REU271_9_april_alive_T_cells.fcs
T_cells_REU271_alive_T_cells.fcs                     T_cells_REU271_alive_T_cells.fcs
T_cells_REU272_12_july_alive_T_cells.fcs     T_cells_REU272_12_july_alive_T_cells.fcs
T_cells_REU272_13_april_alive_T_cells.fcs T_cells_REU272_13_april_alive_T_cells.fcs
T_cells_REU272_14_april_alive_T_cells.fcs T_cells_REU272_14_april_alive_T_cells.fcs
T_cells_REU272_7_apr_alive_T_cells.fcs         T_cells_REU272_7_apr_alive_T_cells.fcs
T_cells_REU272_9_april_alive_T_cells.fcs     T_cells_REU272_9_april_alive_T_cells.fcs
T_cells_REU272_alive_T_cells.fcs                     T_cells_REU272_alive_T_cells.fcs
```

Add a new column to the phenotypic data

```
> pData(fcs_data)$gender <- c(rep("male",8), rep("female",8))
> pData(fcs_data)
```

```
                                                                                    name gender
T_cells_REU267_alive_T_cells.fcs                         T_cells_REU267_alive_T_cells.fcs   male
T_cells_REU268_alive_T_cells.fcs                         T_cells_REU268_alive_T_cells.fcs   male
T_cells_REU269_alive_T_cells.fcs                         T_cells_REU269_alive_T_cells.fcs   male
T_cells_REU270_alive_T_cells.fcs                         T_cells_REU270_alive_T_cells.fcs   male
T_cells_REU271_12_july_alive_T_cells.fcs         T_cells_REU271_12_july_alive_T_cells.fcs   male
T_cells_REU271_13_april_alive_T_cells.fcs T_cells_REU271_13_april_alive_T_cells.fcs   male
T_cells_REU271_14_april_alive_T_cells.fcs T_cells_REU271_14_april_alive_T_cells.fcs   male
T_cells_REU271_7_apr_alive_T_cells.fcs             T_cells_REU271_7_apr_alive_T_cells.fcs   male
T_cells_REU271_9_april_alive_T_cells.fcs         T_cells_REU271_9_april_alive_T_cells.fcs female
T_cells_REU271_alive_T_cells.fcs                         T_cells_REU271_alive_T_cells.fcs female
T_cells_REU272_12_july_alive_T_cells.fcs         T_cells_REU272_12_july_alive_T_cells.fcs female
T_cells_REU272_13_april_alive_T_cells.fcs T_cells_REU272_13_april_alive_T_cells.fcs female
T_cells_REU272_14_april_alive_T_cells.fcs T_cells_REU272_14_april_alive_T_cells.fcs female
T_cells_REU272_7_apr_alive_T_cells.fcs             T_cells_REU272_7_apr_alive_T_cells.fcs female
T_cells_REU272_9_april_alive_T_cells.fcs         T_cells_REU272_9_april_alive_T_cells.fcs female
T_cells_REU272_alive_T_cells.fcs                         T_cells_REU272_alive_T_cells.fcs female
```

Manipulating a *FlowSet*

- Extract a *flowFrame* from a flowSet object using the [[operator

```
> fcs_data[[1]]
```

```
flowFrame object 'T_cells_REU267_alive_T_cells.fcs'
with 265857 cells and 39 observables:
                  name   desc     range  minRange  maxRange
$P1              FSC-A     NA   4194304         0   4194303
$P2              FSC-H     NA   4194304         0   4194303
$P3              SSC-A     NA   4194304         0   4194303
$P4            SSC-B-A     NA   4194304         0   4194303
$P5            SSC-B-H     NA   4194304         0   4194303
```

- Create a new *flowSet* object by subsetting with the [operator

```
> fcs_data[1:5]
```

```
A flowSet with 5 experiments.

column names(39): FSC-A FSC-H ... FJComp-eFluor 660-A Time
```

Manipulating a *FlowSet*

- Subset the *flowSet* based on a condition

```
> fcs_data_males <- fcs_data[pData(fcs_data)$gender=="male"]
> fcs_data_females <- subset(fcs_data, pData(fcs_data)$gender==”female")
```

- Split the *flowSet* based on a condition

```
> fcs_data_split <- split(fcs_data, pData(fcs_data)$gender)
> names(fcs_data_split)
[1] "female" "male"
```

Manipulating a *FlowSet*

- Combine *flowSets* (or *flowSets* and *flowFrames*)

```
> fcs_data_combined <- rbind2(fcs_data_split$female,fcs_data_split$male)
> fcs_data_combined
```

```
                                                                              name gender  split
T_cells_REU271_9_april_alive_T_cells.fcs   T_cells_REU271_9_april_alive_T_cells.fcs female female
T_cells_REU271_alive_T_cells.fcs                   T_cells_REU271_alive_T_cells.fcs female female
T_cells_REU272_12_july_alive_T_cells.fcs   T_cells_REU272_12_july_alive_T_cells.fcs female female
T_cells_REU272_13_april_alive_T_cells.fcs T_cells_REU272_13_april_alive_T_cells.fcs female female
T_cells_REU272_14_april_alive_T_cells.fcs T_cells_REU272_14_april_alive_T_cells.fcs female female
T_cells_REU272_7_apr_alive_T_cells.fcs       T_cells_REU272_7_apr_alive_T_cells.fcs female female
T_cells_REU272_9_april_alive_T_cells.fcs   T_cells_REU272_9_april_alive_T_cells.fcs female female
T_cells_REU272_alive_T_cells.fcs                   T_cells_REU272_alive_T_cells.fcs female female
T_cells_REU267_alive_T_cells.fcs                   T_cells_REU267_alive_T_cells.fcs   male   male
T_cells_REU268_alive_T_cells.fcs                   T_cells_REU268_alive_T_cells.fcs   male   male
T_cells_REU269_alive_T_cells.fcs                   T_cells_REU269_alive_T_cells.fcs   male   male
T_cells_REU270_alive_T_cells.fcs                   T_cells_REU270_alive_T_cells.fcs   male   male
T_cells_REU271_12_july_alive_T_cells.fcs   T_cells_REU271_12_july_alive_T_cells.fcs   male   male
T_cells_REU271_13_april_alive_T_cells.fcs T_cells_REU271_13_april_alive_T_cells.fcs   male   male
T_cells_REU271_14_april_alive_T_cells.fcs T_cells_REU271_14_april_alive_T_cells.fcs   male   male
T_cells_REU271_7_apr_alive_T_cells.fcs       T_cells_REU271_7_apr_alive_T_cells.fcs   male   male
```

Vizualizing Cytometry Data with *ggcyto* R Package

- https://www.bioconductor.org/packages/release/bioc/html/ggcyto.html
- Interface to the ggplot2 graphics system
- Installation:

```
if (!require("BiocManager", quietly = TRUE))
    install.packages("BiocManager")

BiocManager::install("ggcyto")
```

- Vignettes

https://www.bioconductor.org/packages/release/bioc/vignettes/ggcyto/inst/doc/Top_features_of_ggcyto.html

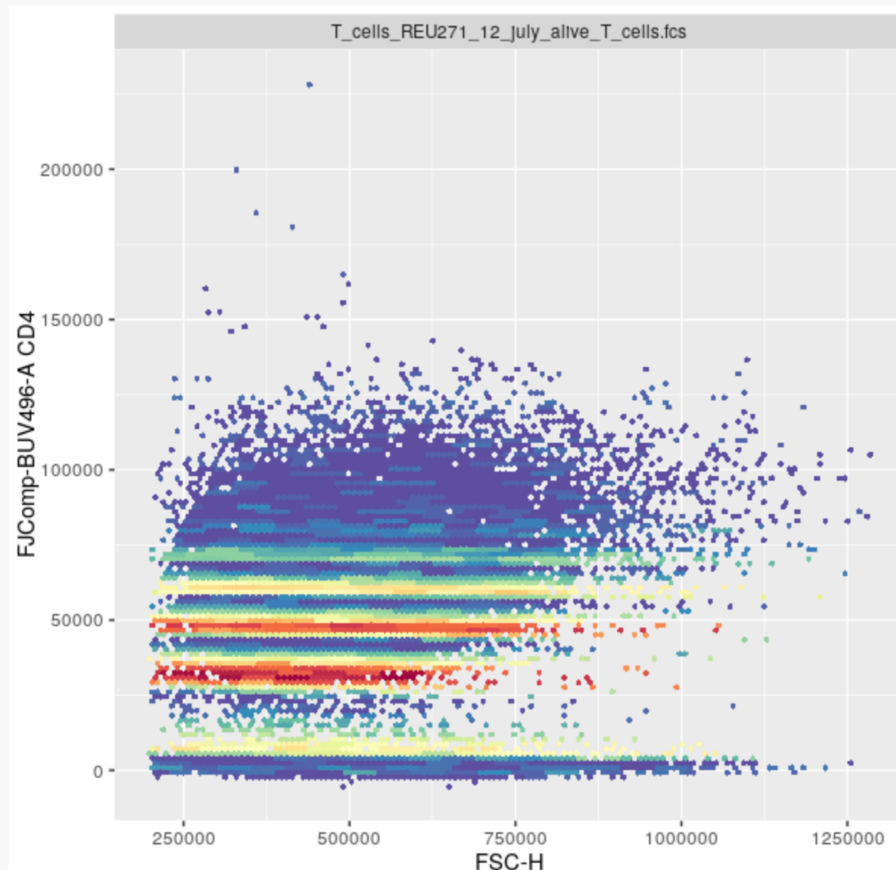
https://www.bioconductor.org/packages/release/bioc/vignettes/ggcyto/inst/doc/ggcyto.flowSet.html

Vizualizing a single *flowFrame*

The function **autoplot()** can be used to create a **bivariate density plot**.

Example:

```
> autoplot(object = fcs_data[[5]], x=”FSC-H", y="FJComp-BUV496-A",
  bins = 2^7)
```

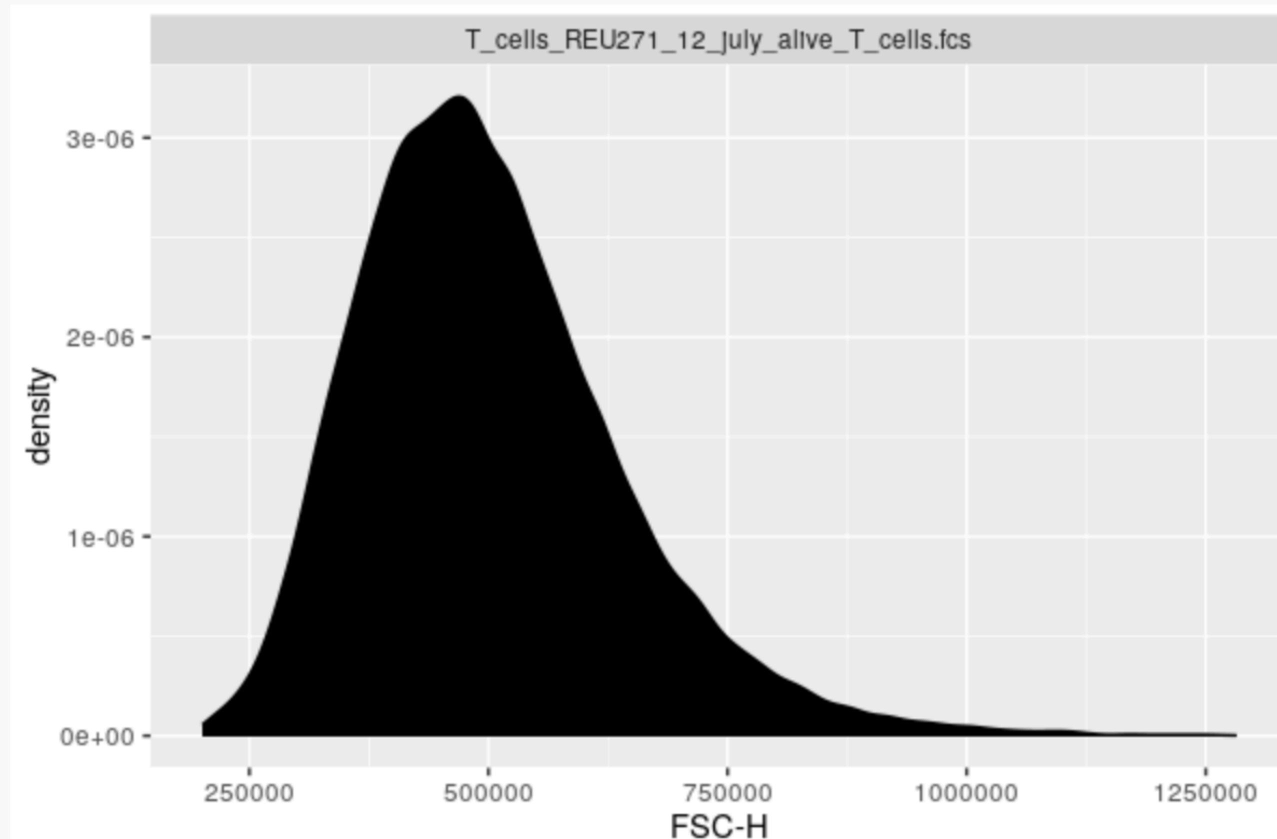


- **bins** sets the granularity of the plot. The higher the number of bins, the finer the granularity

Vizualizing a single *flowFrame*

Similarly, to get a **univariate densityplot**:

```
> autoplot(object = fcs_data[[5]], x="FSC-H")
```



In-line transformation

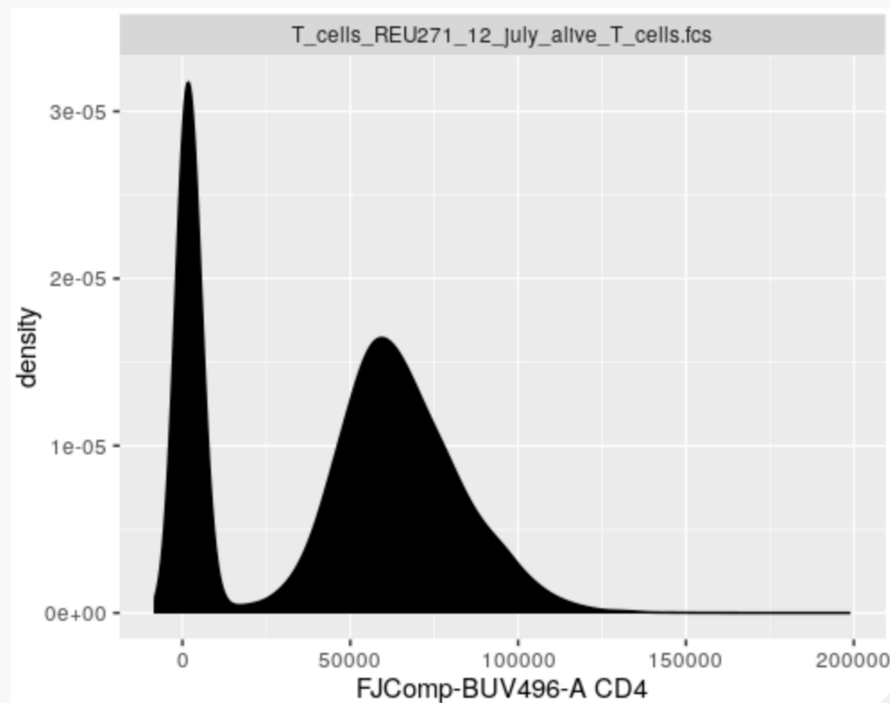
IUse a different scalefo the data

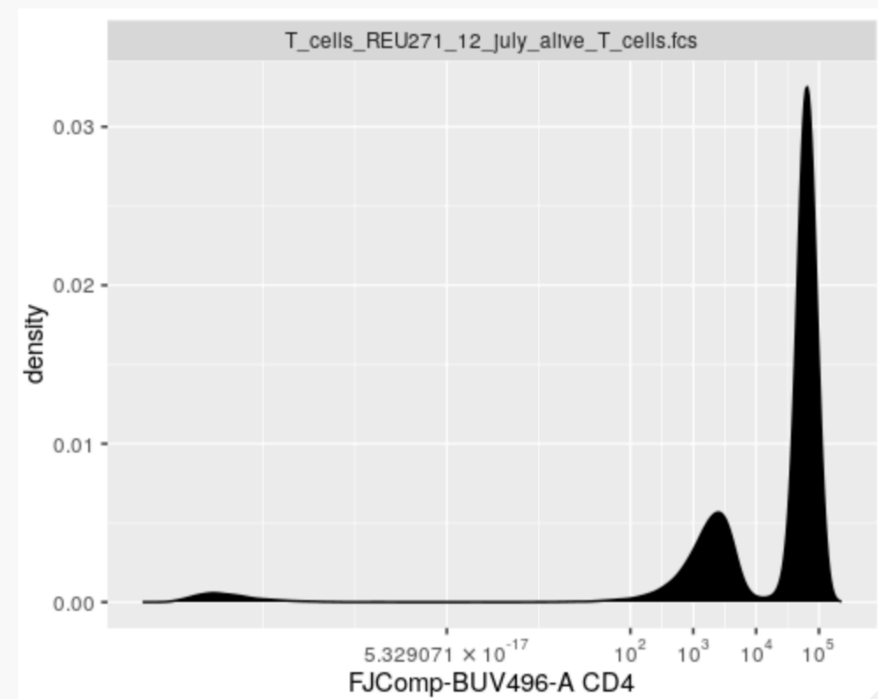
```
> autoplot(fcs_data[[5]],
           x="FJComp-BUV496-A")
```

```
> autoplot(fcs_data[[5]],
  x="FJComp-BUV496-A") +
  scale_x_flowjo_biexp()
```

Original scale

flowJo inverse hyperbolic sine



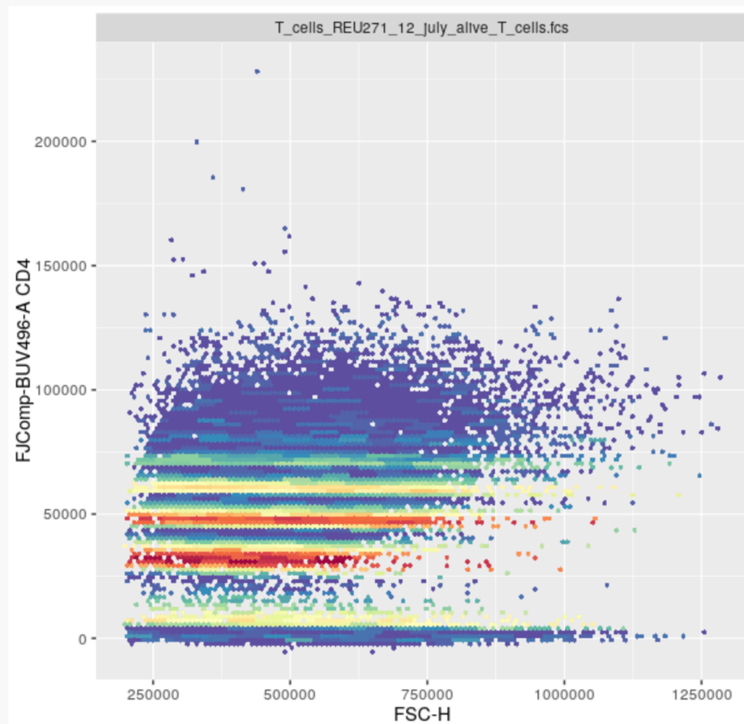


In-line transformation

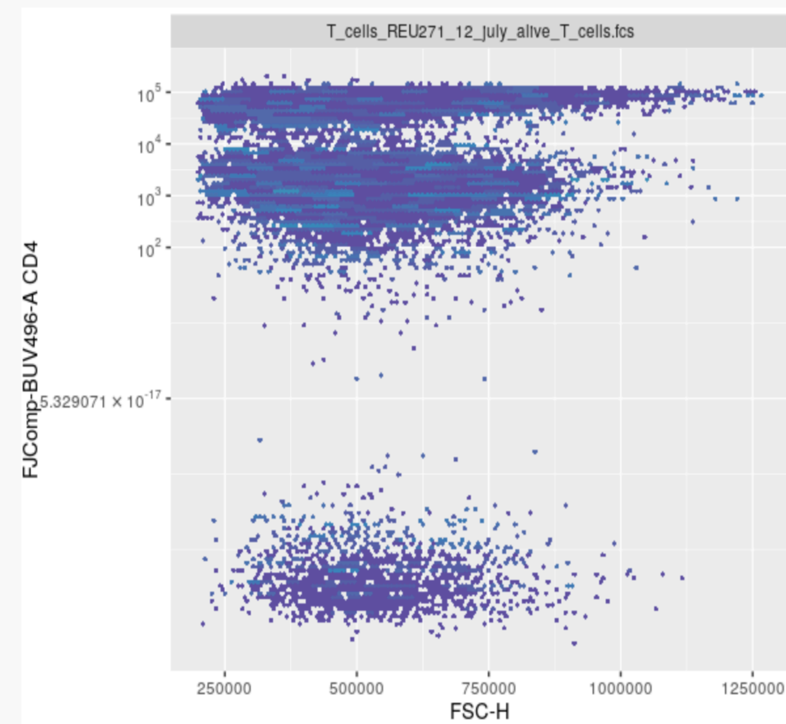
Example in a **bivariate density plot**.

```
> autoplot(object = fcs_data[[5]], x="FSC-H", y="FJComp-BUV496-A",
  bins = 2^7) + scale_y_flowjo_fasinh()
```

Original scale



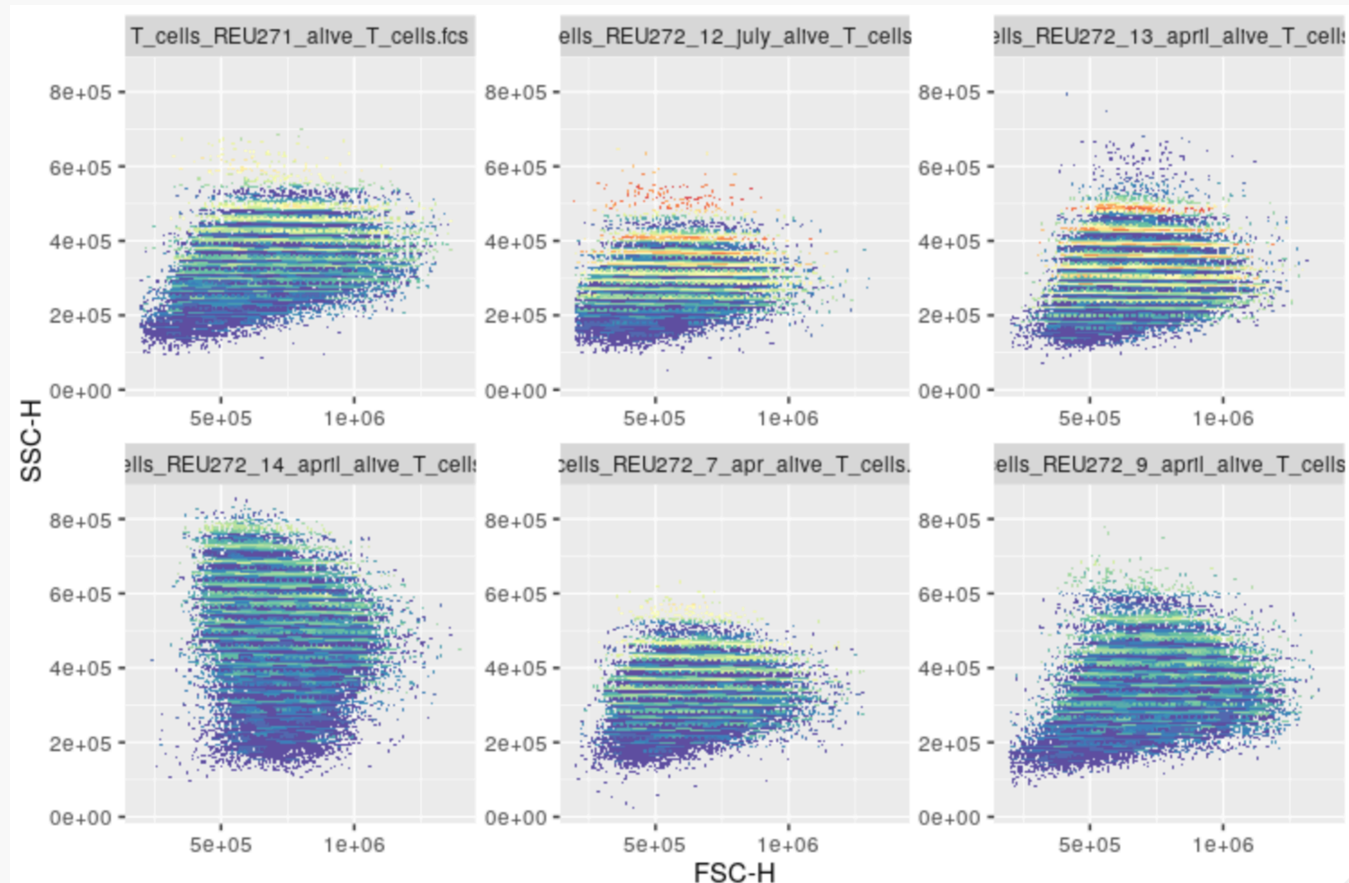
flowJo inverse hyperbolic sine



Vizualizing a *flowSet*

The syntax is basically the same for *flowSet* objects, with the output now being a grid of plots corresponding to each *flowFrame*. Example:

```
> autoplot(object = fcs_data[10:15], x="FSC-H", y="SSC-H", bins = 2^7)
```



Let's practice – 10

In this exercise we will use a 36-color spectral flow cytometry dataset from a study performed in the context of Covid-19 research. Only a subset from 5 healthy donors will be used. For each healthy donor, there are three time points, as indicated in FCS file names. Data was downloaded through the Flow Repository database (FR-FCM-Z3WR) at https://flowrepository.org/id/FR-FCM-Z3WR. FCS files were pre-gated on live CD3+CD19-T cells in FlowJo.

Create a new report in which you will

1) Import the FCS files (course_datasets/FR_FCM_Z3WR/). Do not transform or truncate the values
2) Create a data frame with the list of channels and corresponding antigens, and plot it . **Hint:** get the antigens from the parameters of one of the flowFrame in the set
3) Convert the channel names in the expression matrices to the corresponding antigen names (where applicable)
4) Add a new column to the phenotypic data with the time point of the sample. Plot the phenotypic data
5) Create a bivariate density plot showing «FSC-H» against «HLA-DR» for all samples from day 0. Apply a flowJo inverse hyperbolic sine scale to the y axis («HLA-DR»)