Analyzing flow cytometry data in Bioconductor

Nishant Gopalakrishnan

Fred Hutchinson Cancer Research Center

30 July, 2010

Introduction

Data visualization and transformation

Sequential gating of Data

Outline

Introduction

Data visualization and transformation

Sequential gating of Data

Bioconductor packages

- flowCore
- flowStats
- flowViz
- flowQ
- flowClust
- flowMerge
- flowMeans
- flowUtils
- flowFP

Data structures for flow cytometry

flowFrame: flow data for a single sample

- exprs
- parameters
- description

Data structures for flow cytometry

flowSet: flow data for multiple samples along with meta data

Useful methods

- sampleNames
- colnames
- pData
- `"["`
- `"[["`

Read in flow data

- `read.FCS(filename, transformation )`
- `read.flowSet(files, path, phenoData, transformation)`

Excercise 1 : Get FCS files

1.
   ```
   library(flowCore)
   library(flowCytBioc)
   library(flowViz)
   system.file("extdata", package = "flowtrack")
   ```
2. Copy the files in the ”extdata” folder to your working directory

Excercise 2: Create a flowSet

1. Create a flowSet by reading in the supplied fcs files and phenoData information using the `read.flowSet` function.
2. Observe the phenoData information stored in the flowSet using the `pData` function.
3. Update the sampleNames of the flowSet with the ”PatientID” information from the phenoData information provided.
4. Observe the parameters information for the `flowData[[1]]` *flowFrame*.
5. Update the description field for each *flowFrame* in the *flowSet* with the stain names `c(NA, NA, "CD8", "CD69", "CD4", "CD3", "HLADr", NA)` using the `pData` and `parameters` update methods.

Outline

Introduction

Data visualization and transformation

Sequential gating of Data

Data visualization

```
> xyplot( y ~ x, data, xlab, ylab, main)
> densityplot( ~ x, data, xlab, ylab, main)
> splom(data)
```

Data transformation

- `asinh`
- `logicleTransform`
- `transformList(from, tfun)`
- `transform(data, ...)`

Excercise

- Create an object of the class *transformList* for transforming flourescence channels `FITC-A`, `PE-A`, `FL3-A`, `PE-CY7-A and APC-A` using the `asinh` transformation.
- Transform the `flowData` *flowSet* that we created using the the `transform` method.
- Create a scatter plot of the transformed `‘FITC-A‘` and `‘PE-A‘` channels using the `xyplot` function.
- Create a density plot of the `‘FL3-A‘` channel using the `densityplot` function.

Work flows

- `workFlow(data, name)`
- `add(wf, action)`
- `"[["`
- `Data`
- `undo`

Excercise

- Create a workflow for the transformed data `tData` called ”myWork” using the `workFlow` function.
- Create a rectangle gate using the `rectangeGate` to include parameters FSC-A and SSC-A between the values of 100 and 600
- Add the rectangle gate to the workflow using the `add` function.
- Create a scatter plot of FSC-A and SSC-A for the events included in the rectangle gate using the `xyplot` function. The events included in the gate can be accessed from the workFlow using the `Data` function and the `[[`.

Outline

Introduction

Data visualization and transformation

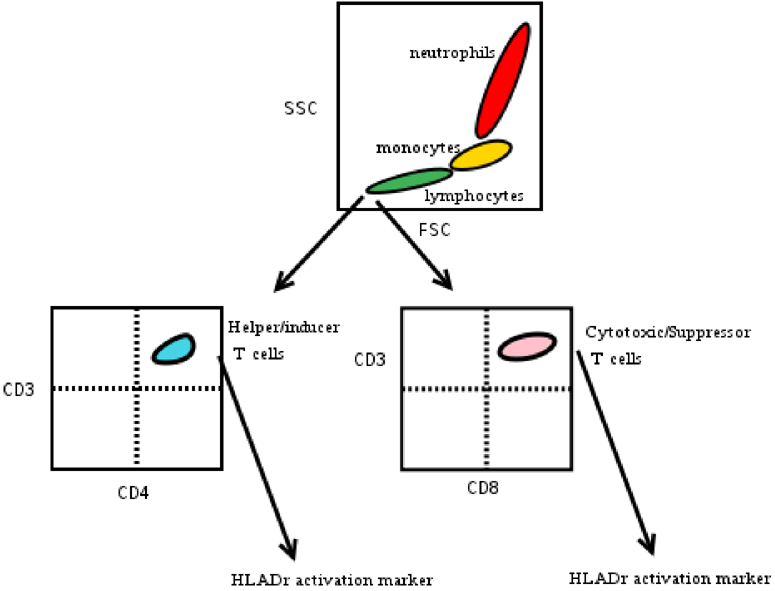
Sequential gating of Data

Goals

For the two groups that received treatment with drug A or B, compare the

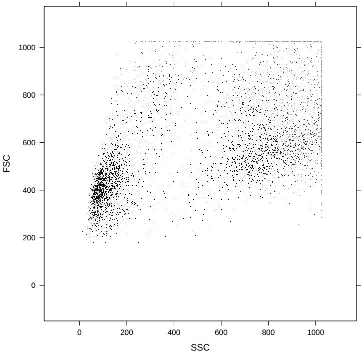
- T helper cells that exhibit HLADr activation marker.
- T cytotoxic cells that exhibit the HLADr activation marker.

Sequential gating strategy



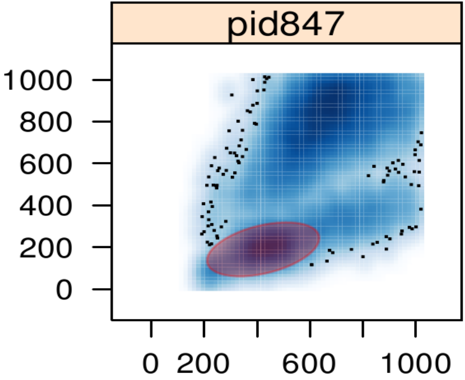
Sequential gating using workflows

- Create a new work flow
- Transform the data using asinh transform
- Remove boundary events for FSC-A SSC-A channels

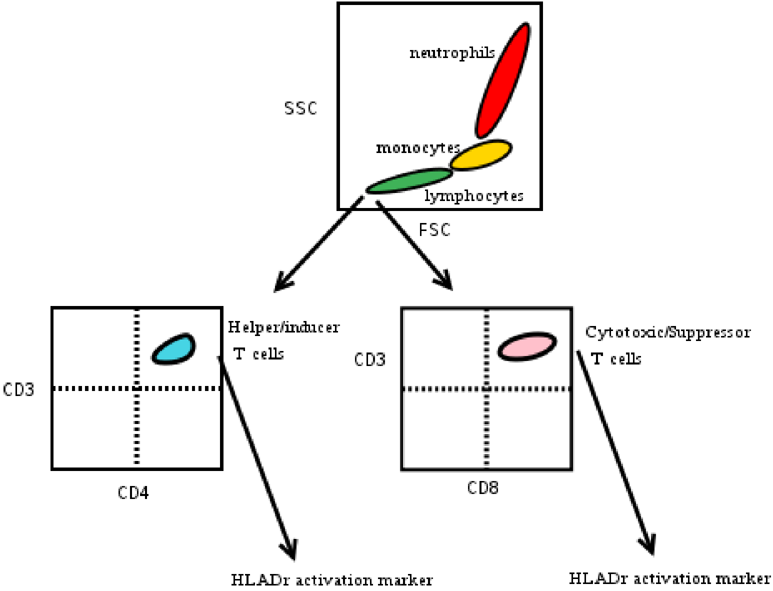


Identify T cells using: `lymphGate`

- create a lymphGate using CD3 preselection and FSC SSC channels

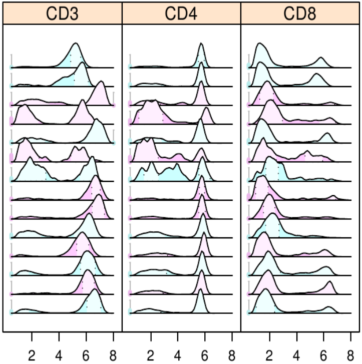


Sequential gating strategy



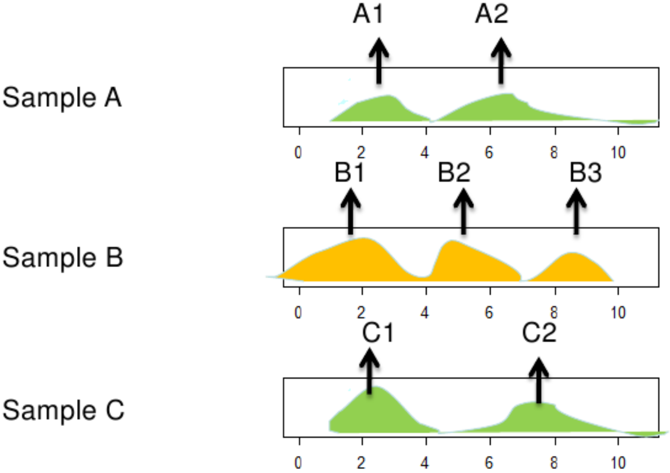
Need for normalization

Quadrant qate



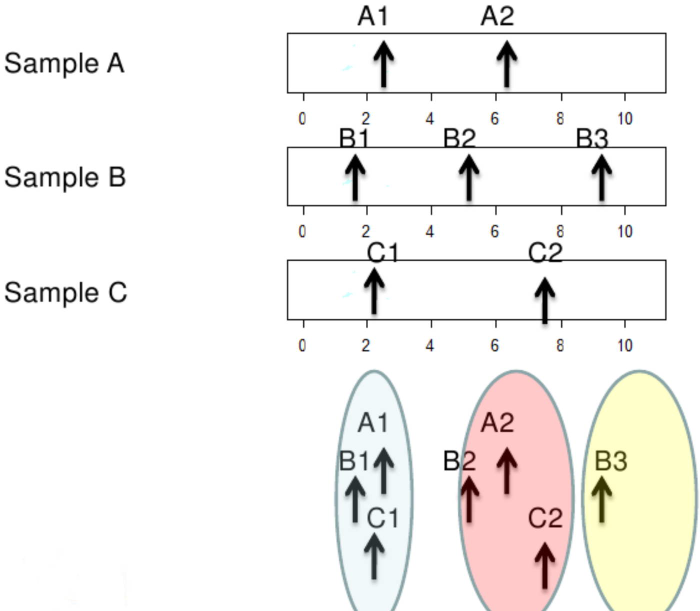
Data Normalization

Feature identifcation



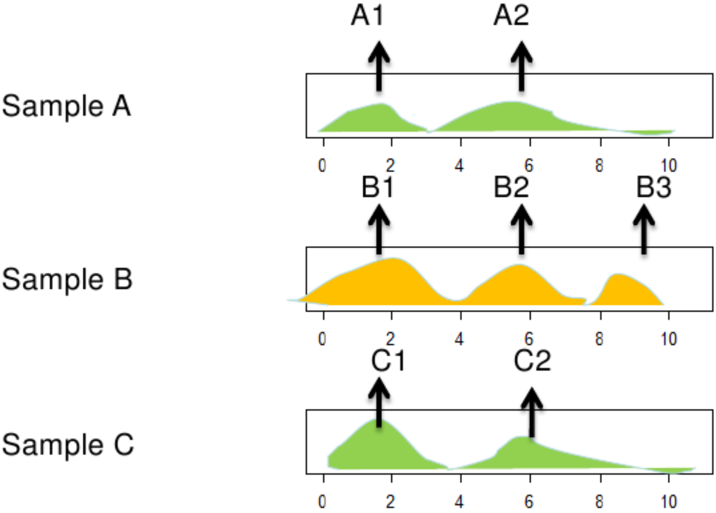
Data Normalization

Feature classification



Data Normalization

Feature alignment



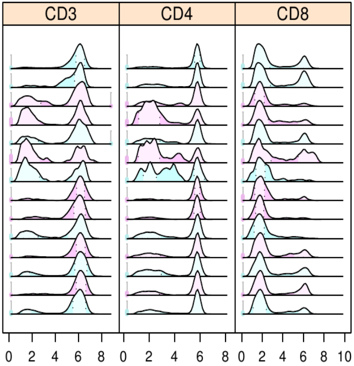
Data Normalization

Assumptions

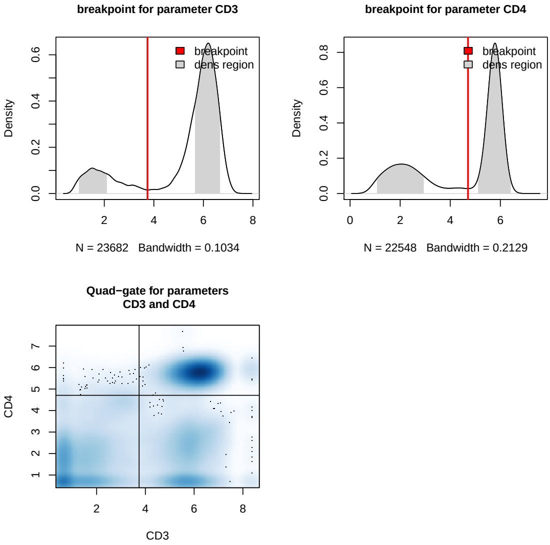
- Fluorescence channels have been compensated
- Median fluorescence intensity
- High density areas represent relevant populations

Data normalized using: `normalization, warpSet`

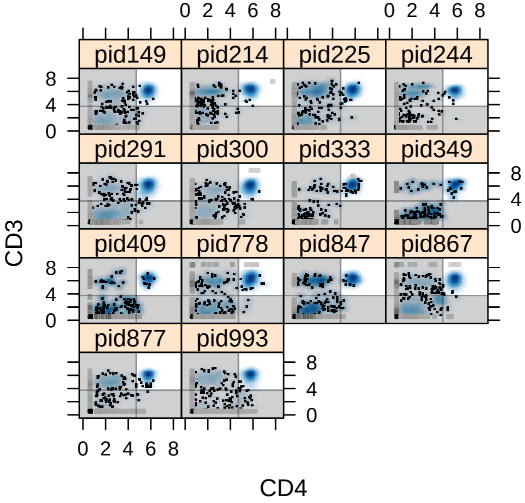
Quadrant qate



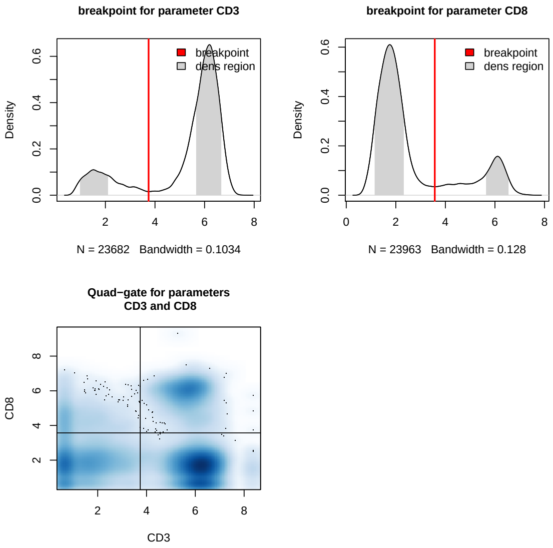
CD3CD4 Quadrant gate calculation



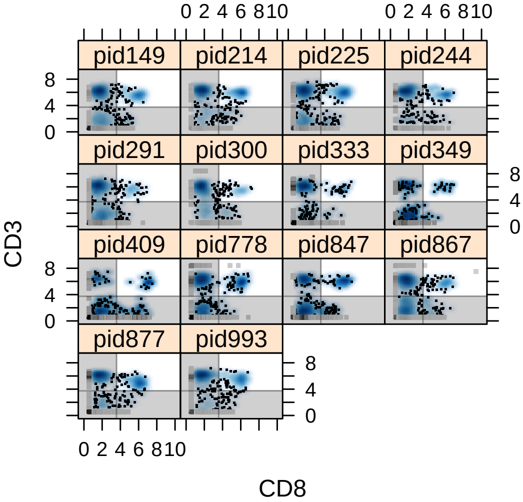
Quadrant gate CD3+CD4+ cells



CD3CD8 Quadrant gate calculation

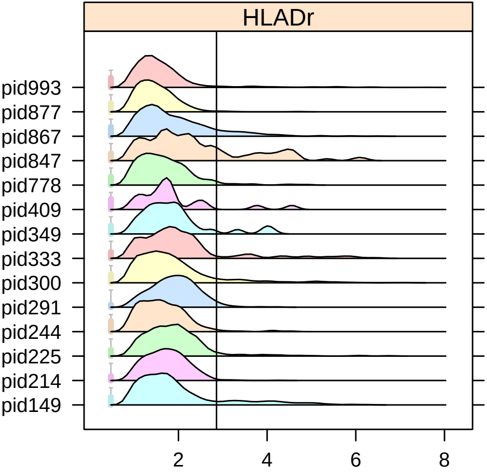


Quadrant gate CD3+CD8+ cells

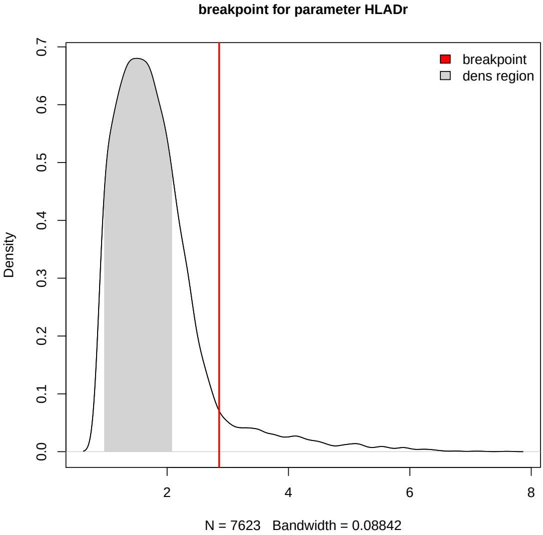


CD3+CD4+ cells with HLADr activation

```
rangeGate
```

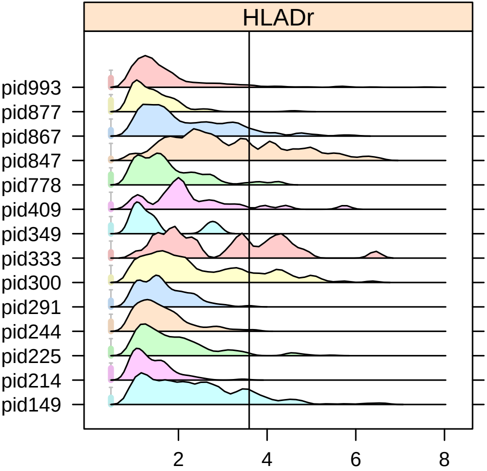


Range Gate: CD3+CD4+ HLADr activation

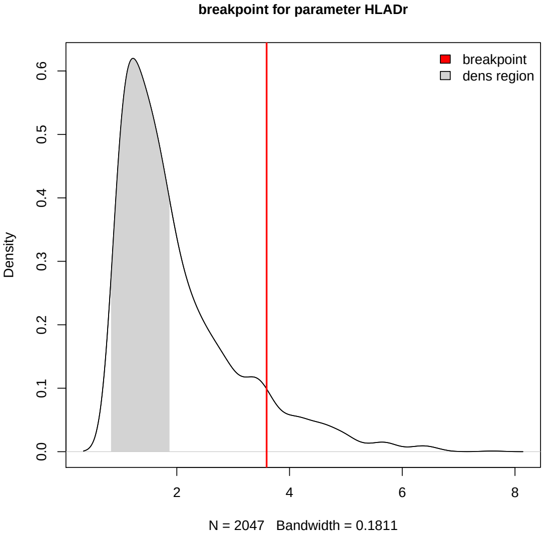


CD3+CD8+ cells with HLADr activation

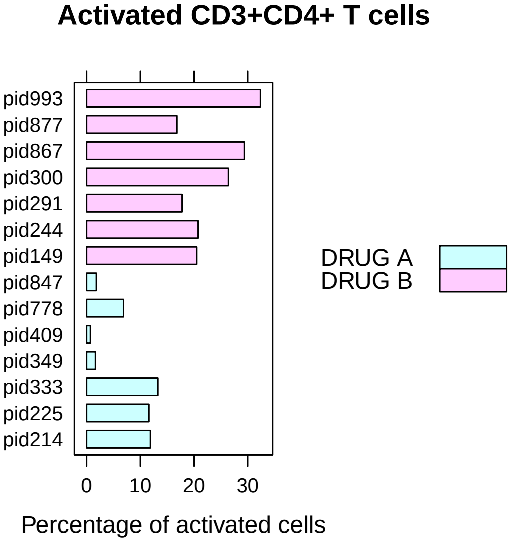
rangeGate



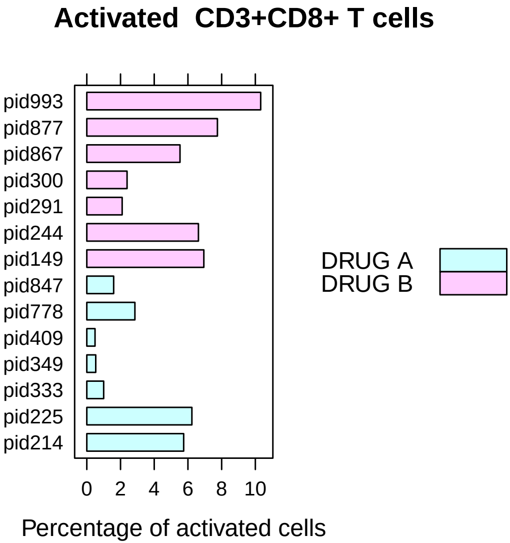
Range Gate: CD3+CD8+ HLADr activation



CD3+CD4+ summary



CD3+CD8+ summary


Activated CD3+CD8+ T cells
pid993
pid877
pid867
pid300
pid291
pid244
pid149
pid847
pid778
pid409
pid349
pid333
pid225
pid214
Percentage of activated cells
DRUG A
DRUG B