````
---
title: "QA FCM data in R"
author: "R. Burke Squires, Radina Droumeva"
date: "7/11/2017"
output: ioslides_presentation
---

QA FCM data in R
================


```{r setup, include=FALSE}
require("knitr")
knitr::opts_chunk$set(echo = FALSE)
opts_knit$set(root.dir = "./") #~/gits/project/")
```


# Start fresh: remove all loaded variables:

```{r}
rm(list=ls())
graphics.off() # close all graphics, for efficiency
```

Ensure flowCore library is loaded:

```{r}
library(flowCore)
#setwd('./data')
```

To see a list of files in the 'fullFCS' folder, use dir(), it returns a vector of
file names

```{r}
files <- dir('./data/fullFCS/')
files
```

To select the first file in the directory, subset the vector on its first index:

```{r}
firstFile <- paste('./data/fullFCS/', files[1], sep = "")
firstFile
f <- read.FCS(firstFile)
f
```

##############################################################################

# Quality Assurance


## Load helper package and one support function from Radina
````

```{r}
library(flowQ)
source("./code/supportCode/qaProcess.GenericNumber.R")
```

Define directory where QA results will be saved:

```{r}
save.dir <- "./data/QA/"
```

This removes the current contents of the QA folder. Do this for space efficiency and to ensure correct results:
(You may get an error the first time, as there is no "QA" folder yet. If later you set this up on your own computer and don't use Ubuntu OS, you can just delete the folder yourself in the file browser)

```{r}
system(paste('rm -r ', save.dir, "*", collapse="", sep=""))
```

First, check the raw cell counts (use 'fs')

See ?qaProcess.cellnumber for explanation

```{r}
load('./data/fs.RData') # Make sure we get the raw counts -- these should all be 20,000 for us!
qa.raw.count <- qaProcess.cellnumber(fs, outdir=save.dir, cFactor=Inf)
```

Next, examine non-debris cell counts from 'clean.fs'

```{r}
qa.nonDebris.count <- qaProcess.cellnumber(set = clean.fs, outdir=save.dir)
```

Use Radina's "pretty" non-debris count and set threshold at 10000 cells:

```{r}
numbers <- as.vector(fsApply(clean.fs, nrow))
frameIDs <- as.vector(sampleNames(fs))
qa.nonDebris.pretty <- qaProcess.GenericNumber(numbers=numbers,
frameIDs=frameIDs, outdir=save.dir, cutoff=10000, name="Cell count")
```

flowQ provides an easy html report function:

```{r}
url <- writeQAReport(fs, list(qa.raw.count, qa.nonDebris.count,
qa.nonDebris.pretty), outdir=save.dir)
browseURL(url)
```

You can do quality checks without relying on flowQ.

Can you say anything about the sample quality based on the following plot?

```{r}
graphics.off() # closes all current plots, good for computer speed
plot(density(exprs(trans.fs[[1]])[, "FSC-A"]), xlim=c(35000, 125000), ylim =
c(0,0.00005), lwd=2, main = "FSC Density", sub="", xlab="FSC-A")
for (i in 2:10){
  lines(density(exprs(trans.fs[[i]])[, "FSC-A"]), col=i, lwd=2)
}
```

Save the transformed flowSet object for tomorrow!

```{r}
save(trans.fs, file='/home/rguru/Documents/Workshop/data/trans.fs.RData')
```

##

Radina has provided a set of functions to process typical data sets:
#graphics.off()
#source("../code/supportCode/flowPrep.R")
#source("../code/supportCode/support_functions.R")
#fs.prep <- flowPrep(fs, apply.comp=FALSE, plot.preproc=TRUE,
plot.for.lympho=TRUE)