Flow Cytometry Analysis

What do I need to make sure I illustrate:

1. In tomato leaves there is a slight change from 2C to 4C, but not much beyond that.
2. There is no difference between tip and base in plants.

Read in data and establish dependencies.

```
library(gdata)
```

```
## gdata: read.xls support for 'XLS' (Excel 97-2004) files ENABLED.

##

## gdata: read.xls support for 'XLSX' (Excel 2007+) files ENABLED.

##
## Attaching package: 'gdata'

## The following object is masked from 'package:stats':
##
##     nobs

## The following object is masked from 'package:utils':
##
##     object.size

## The following object is masked from 'package:base':
##
##     startsWith
```

```
library(plyr)
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.3.2

## Warning: failed to assign NativeSymbolInfo for env since env is already
## defined in the 'lazyeval' namespace
```

```
library(reshape2)

flowC = read.xls("../data/flowCytometry.xlsx", sheet = 1, header = TRUE)
names(flowC)
```

```
##  [1] "leafSample"  "genotype"    "plant"       "leaf"        "section"
##  [6] "area"        "area2c"      "area4c"      "area8C"      "percent2C"
## [11] "percent4C"   "percent8C"   "germinated"  "measurement" "age"
```

```
str(flowC)
```

```
## 'data.frame':    57 obs. of  15 variables:
##  $ leafSample : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ genotype   : Factor w/ 2 levels "e2","wt": 1 1 2 1 2 2 1 1 2 1 ...
##  $ plant      : Factor w/ 22 levels "0","11","12",..: 22 22 22 22 22 22 22 22 4 6 ...
##  $ leaf       : Factor w/ 7 levels "cot","L1","l2",..: 1 1 4 4 4 4 4 4 5 5 ...
##  $ section    : Factor w/ 6 levels "base","mid","na",..: 3 3 6 6 6 6 6 6 6 6 ...
##  $ area       : num  14.2 22.8 43 53.8 57 ...
##  $ area2c     : num  114 1845 128 2175 2388 ...
##  $ area4c     : num  169 728 181 418 223 ...
##  $ area8C     : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ percent2C  : num  0.404 0.717 0.413 0.839 0.915 ...
##  $ percent4C  : num  0.5956 0.2829 0.587 0.1612 0.0854 ...
##  $ percent8C  : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ germinated : Factor w/ 2 levels "26-Jan","9-Jan": 2 2 2 2 2 2 2 2 1 1 ...
##  $ measurement: Factor w/ 5 levels "10-Apr","13-Feb",..: 5 5 5 5 5 5 5 5 2 2 ...
##  $ age        : int  8 8 8 8 8 8 8 8 8 8 ...
```

Reshape data.

```
#subset only what I need
names(flowC)
```

```
##  [1] "leafSample"  "genotype"    "plant"       "leaf"        "section"
##  [6] "area"        "area2c"      "area4c"      "area8C"      "percent2C"
## [11] "percent4C"   "percent8C"   "germinated"  "measurement" "age"
```

```
levels(flowC$section)
```

```
## [1] "base"  "mid"   "na"    "tip"   "tip "  "whole"
```

```
#fix typo
flowC$section <- gsub("tip ", "tip", flowC$section)

# Clean up
flowCSub1 <- flowC[,c(1:6, 10:12, 15)]
names(flowCSub1)
```

```
##  [1] "leafSample" "genotype"   "plant"      "leaf"       "section"
##  [6] "area"       "percent2C"  "percent4C"  "percent8C"  "age"
```

```
str(flowCSub1)
```

```
## 'data.frame':    57 obs. of  10 variables:
##  $ leafSample: int  1 2 3 4 5 6 7 8 9 10 ...
##  $ genotype  : Factor w/ 2 levels "e2","wt": 1 1 2 1 2 2 1 1 2 1 ...
##  $ plant     : Factor w/ 22 levels "0","11","12",..: 22 22 22 22 22 22 22 22 4 6 ...
##  $ leaf      : Factor w/ 7 levels "cot","L1","l2",..: 1 1 4 4 4 4 4 4 5 5 ...
##  $ section   : chr  "na" "na" "whole" "whole" ...
```

```
##  $ area      : num  14.2 22.8 43 53.8 57 ...
##  $ percent2C : num  0.404 0.717 0.413 0.839 0.915 ...
##  $ percent4C : num  0.5956 0.2829 0.587 0.1612 0.0854 ...
##  $ percent8C : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ age       : int  8 8 8 8 8 8 8 8 8 8 ...
```

```
flowCSub1[is.na(flowCSub1)] <- 0
flowCMelt <- melt(flowCSub1, id.vars = c("leafSample", "genotype", "plant", "leaf", "section", "area",
  variable.name = "peaks",
  value.name = "flowValue")
```

Genotype

We are not using the entire-2 data. Only wildtype There are two data sets 1. flowCMelt and 2.flowCSub1, make for wt only.

```
flowCMelt <- subset(flowCMelt, genotype == "wt")
flowCSub1 <- subset(flowCSub1, genotype == "wt")
```

Age

Since age of (30 and 31) and (90 and 91) are essentially the same, I am going to collapse them.

```
flowCMelt$age <- as.factor(flowCMelt$age) #treat age as factor
flowCMelt$age <- sub("31", "30", flowCMelt$age)
flowCMelt$age <- sub("91", "60", flowCMelt$age)

flowCMelt.summary <- ddply(flowCMelt, c("genotype", "peaks","age"), summarise,
            N  = length(flowValue),
            flowCyt = mean(flowValue),
            sd = sd(flowValue),
            se = sd / sqrt(N) )

flowCMelt.summary
```

```
##    genotype     peaks age N   flowCyt         sd         se
## 1        wt percent2C  18 8 0.7666582 0.18551839 0.06559066
## 2        wt percent2C  30 8 0.7394346 0.15221015 0.05381441
## 3        wt percent2C  60 5 0.6737317 0.09956442 0.04452656
## 4        wt percent2C   8 4 0.6523421 0.27876819 0.13938410
## 5        wt percent2C  90 4 0.5972060 0.06555129 0.03277565
## 6        wt percent4C  18 8 0.2333418 0.18551839 0.06559066
## 7        wt percent4C  30 8 0.2566654 0.14765031 0.05220227
## 8        wt percent4C  60 5 0.3110837 0.06880662 0.03077126
## 9        wt percent4C   8 4 0.3476579 0.27876819 0.13938410
## 10       wt percent4C  90 4 0.4027940 0.06555129 0.03277565
## 11       wt percent8C  18 8 0.0000000 0.00000000 0.00000000
## 12       wt percent8C  30 8 0.0037500 0.01060660 0.00375000
```

```
## 13       wt percent8C  60 5 0.0160000 0.03577709 0.01600000
## 14       wt percent8C   8 4 0.0000000 0.00000000 0.00000000
## 15       wt percent8C  90 4 0.0000000 0.00000000 0.00000000
```

```
#Only WT

wtOnly <- subset(flowCMelt.summary, genotype == "wt")
```

Visualize 1

First visualize across age.

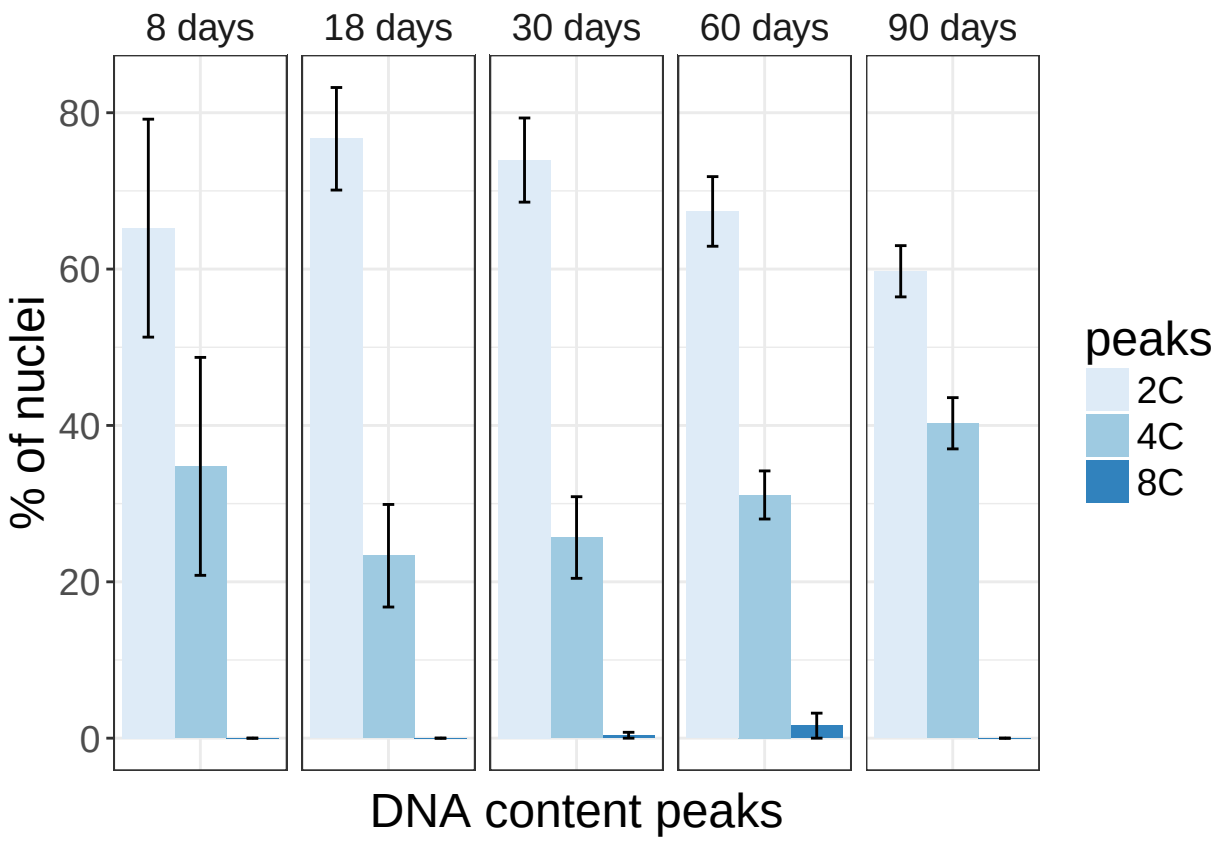
```
## Change Data for Plot
wtOnly$peaks <- gsub("percent", "", wtOnly$peaks)
wtOnly$age <- paste(wtOnly$age,"days", sep=" ")

## change order of age, so facetting is correct

wtOnly$age <- factor(wtOnly$age,
         levels = c("8 days", "18 days", "30 days", "60 days", "90 days"))

## Change decimal places
wtOnly[,c(5:7)] <- wtOnly[,c(5:7)] * 100

# Plot with error bars
ggplot(wtOnly, aes(genotype, flowCyt, fill = peaks)) +
   geom_bar(stat = "identity", position = "dodge") +
  geom_errorbar(aes(ymin=flowCyt - se, ymax=flowCyt + se),
                width=.2,
                colour="black",
                position = position_dodge(.9)) +
  theme_bw() +
  facet_grid(.~age) +
  scale_fill_brewer() +
  xlab("DNA content peaks") +
  ylab("% of nuclei") +
  theme(strip.background = element_rect(color = "000000", fill="#FFFFFF"),
        text = element_text(size = 18),
        axis.text.x=element_blank(),
        axis.ticks.x=element_blank())
```



Visualize 2

Look at differences between tip and base.

```
#I only want to use one sample per leaf
#remove cotolydons
flowCMelt2 <- subset(flowCMelt, section == "tip" | section == "base")

flowCMelt.summary2 <- ddply(flowCMelt2, c("genotype", "age", "section", "peaks"), summarise,
                    N  = length(flowValue),
                    flowCyt = mean(flowValue),
                    sd = sd(flowValue),
                    se = sd / sqrt(N) )

## Only wildtype
flowCMelt.summary2.wt <- subset(flowCMelt.summary2, genotype == "wt")

# Remove 30 days

flowCMelt.summary2.wt <- subset(flowCMelt.summary2.wt, age != 30)

# change wording
flowCMelt.summary2.wt$peaks <- gsub("percent", "", flowCMelt.summary2.wt$peaks)
flowCMelt.summary2.wt$age <- paste(flowCMelt.summary2.wt$age,"days", sep=" ")

## Change decimal places
```

```
colnames(flowCMelt.summary2.wt)
```

```
## [1] "genotype" "age"      "section"  "peaks"    "N"        "flowCyt"
## [7] "sd"       "se"
```

```
flowCMelt.summary2.wt[,c(6:8)] <- flowCMelt.summary2.wt[,c(6:8)] * 100

## Plot

ggplot(flowCMelt.summary2.wt, aes(peaks, flowCyt, fill = section)) +
   geom_bar(stat = "identity", position = "dodge") +
  geom_errorbar(aes(ymin=flowCyt - se, ymax=flowCyt + se),
                width=.2,
                colour="black",
                position = position_dodge(.9)) +
  theme_bw() +
  facet_grid(.~age) +
  xlab("DNA content peaks") +
  ylab("percent of nuclei") +
  scale_fill_manual(values=c("#548B54","mediumpurple1")) +
  theme(strip.background = element_rect(color = "000000", fill="#FFFFFF"),
        strip.text.x = element_text(),
        text = element_text(size = 20))
```