

Seminar III: R/Bioconductor

August-December 2009

Leonardo Collado Torres

Bachelor in Genomic Sciences (LCG),
UNAM, Cuernavaca, Mexico

lcollado@lcg.unam.mx

<http://www.lcg.unam.mx/~lcollado/>

September 2, 2009

Assistants: Alejandro Reyes areyes@lcg.unam.mx, José Reyes jreyes@lcg.unam.mx
and Víctor Moreno jmoreno@lcg.unam.mx

Note: Questions through the forum please. Those who are not from the sixth LCG
generation send us an email so we can register you on the forum.

Abstract

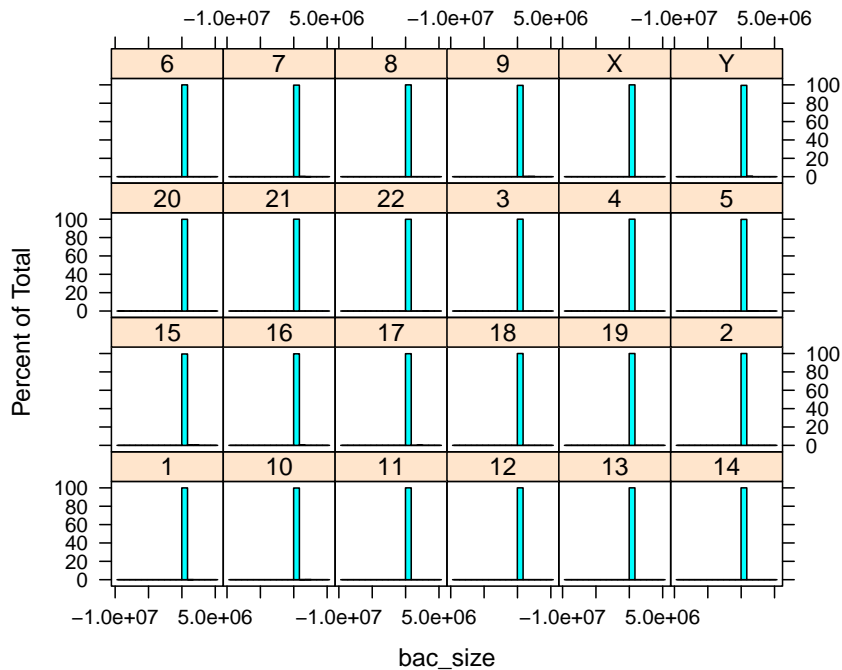
With the following exercises you'll practice creating some advanced plots.
You'll have to explain every plot.

1 lattice

```
> t1 <- read.csv("http://www.lcg.unam.mx/~lcollado/B/data/SuppTable01_kogenome6.d  
+   header = T)  
> t2 <- read.csv("http://www.lcg.unam.mx/~lcollado/B/data/SuppTable06_nsSnp_AK1.c  
+   header = T)  
> library(lattice)  
> library(plotrix)
```

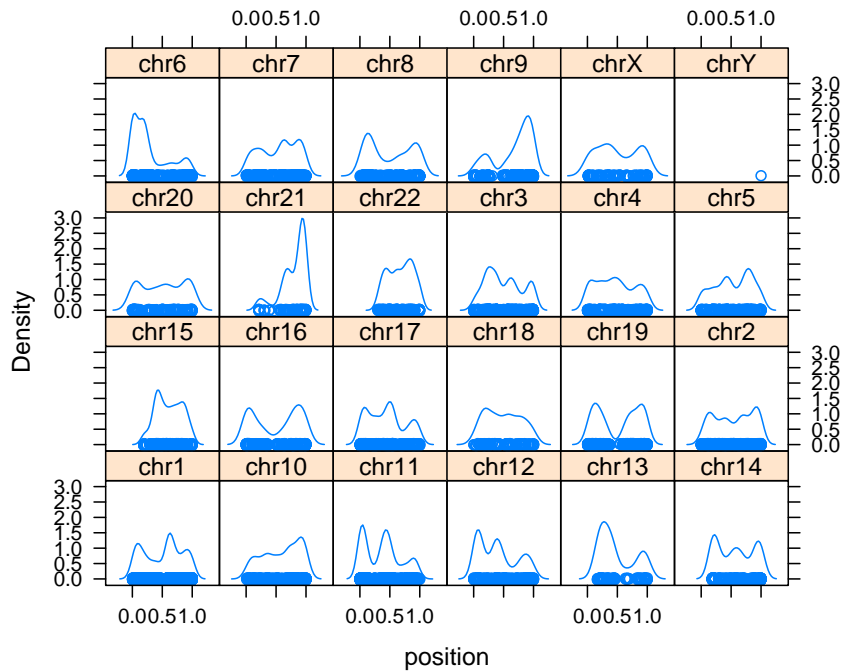
- Using our object `t1`, create a histogram of the BAC sizes with one panel per chromosome. With this plot we notice that mostly all the chromosomes have a tight BAC size distribution.

```
> print(histogram(~bac_size | chrNo, data = t1))
```



- With the object `t2` from class, first normalize the position variable per chromosome (use `tapply` to find out the max values per chromosome). Then create density plots for your normalized position variable. Every chromosome has to have its own panel. Suprisingly, chromosome Y only has one data point, and some are quite skewed such as chromosome 17.

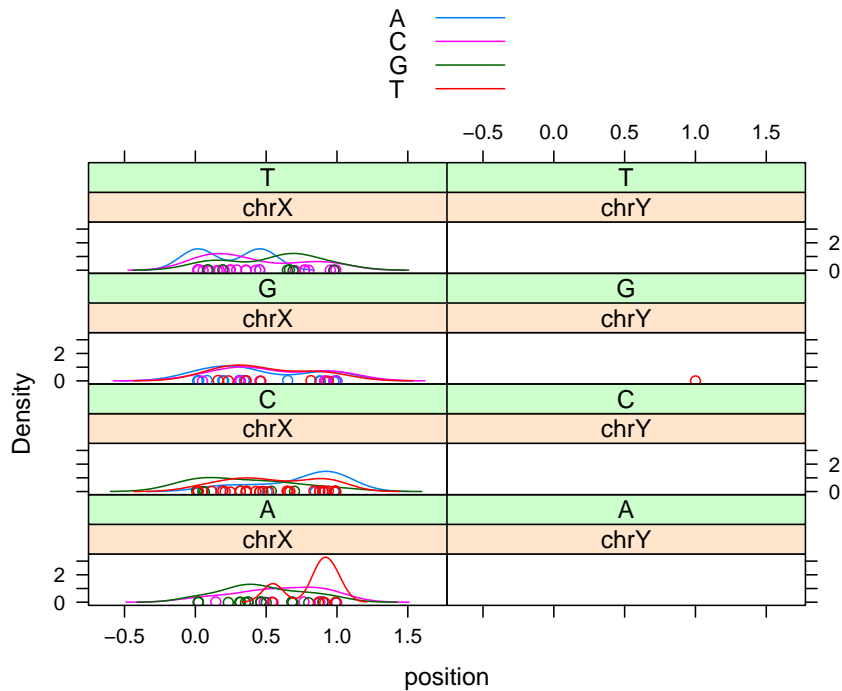
```
> max <- tapply(t2$position, t2$chr, max)
> t2b <- t2
> for (i in 1:length(max)) {
+   pos <- which(t2b$chr %in% names(max[i]))
+   t2b$position[pos] <- t2b$position[pos]/max[i]
+ }
> print(densityplot(~position | chr, data = t2b))
```



3. For chromosomes X and Y, and using the normalized position variable, make a densityplot grouping the information by the reference allele. For each chromosome, separate the data by the `AK1.allele` variable¹. Your resulting plot should have 8 panels. With this graph we notice things like for chr X where most cases where the `AK1.allele` variable is a C, the reference allele variable is a T; and you find them on most of the chromosome except on some gaps visible with the data points.

```
> pos <- which(t2b$chr %in% c("chrX", "chrY"))
> print(densityplot(~position | chr * AK1.allele, groups = reference.allele,
+   data = t2b, subset = pos, auto.key = T))
```

¹Remember that you can use more than 1 factor

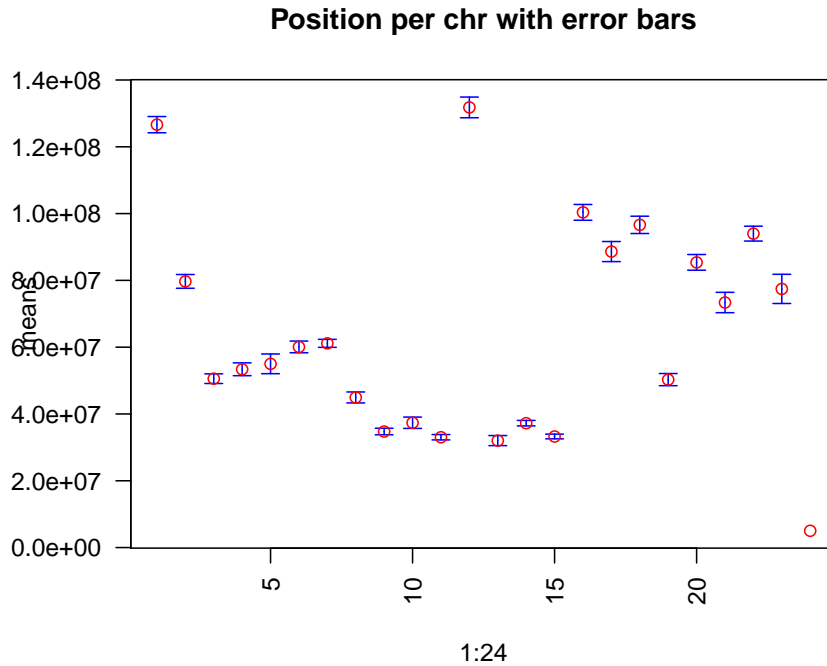


- (Optional) Check out the `latticeExtra` package and make a plot with one of its functions.

2 plotrix

- Using the original `t2` object, plot for every chromosome the mean position with error bars. We did something very similar with `t1` on class. In general, the error bars are small except for chromosomes such as the 23rd one we plot.

```
> means <- tapply(t2$position, t2$chr, mean)
> err <- tapply(t2$position, t2$chr, std.error)
> plotCI(1:24, means, err, col = "red", scol = "blue", las = 2,
+       main = "Position per chr with error bars")
```

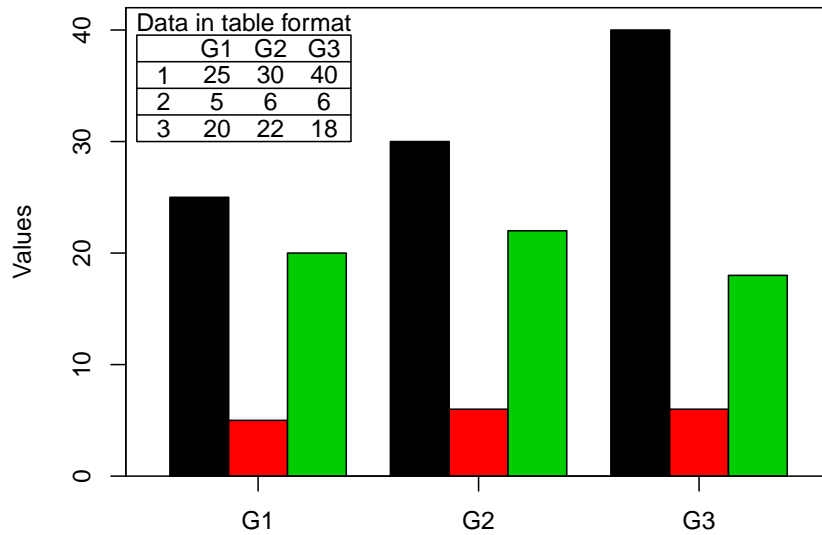


2. Create a bar plot with the table information for the following data. With this plot, well, we can see the data values and the bars on one plot :)

```
> df <- data.frame(G1 = c(25, 5, 20), G2 = c(30, 6, 22), G3 = c(40,
+   6, 18))
> df
```

```
  G1 G2 G3
1 25 30 40
2  5  6  6
3 20 22 18
```

```
> barp(df, ylab = "Values", names.arg = colnames(df), col = 1:3)
> addtable2plot(0.45, 30, df, bty = "o", display.rownames = TRUE,
+   hlines = TRUE, title = "Data in table format")
```



3. (Optional) With whichever data you want, create an interesting plot using hierobarp. Don't use the default examples.