PDCB BioC for HTS topic
Reviewing R: Answers

LCG Leonardo Collado Torres
lcollado@wintergenomics.com – lcollado@ibt.unam.mx

August 25th, 2010

Abstract

Set of answers for the first set of exercises :)

1 Review

1. Why does the following expression show a warning? This is part of what rule?

```
> c(2, 3) + c(4, 5, 7)

> "Because the 2nd vector's length is not a multiple of the first one"

[1] "Because the 2nd vector's length is not a multiple of the first one"

> "and viceversa. Its due to the recycling rule."

[1] "and viceversa. Its due to the recycling rule."
```

2. For all the prime numbers between 1 and 10, calculate its square root. What is the sum, median and mean?

```
> prime <- c(2, 3, 5, 7)
> sqrt(prime)

[1] 1.414214 1.732051 2.236068 2.645751

> sum(prime)
```

```
[1] 17

> sum(sqrt(prime))

[1] 8.028084

> median(prime)

[1] 4

> median(sqrt(prime))

[1] 1.984059

> mean(prime)

[1] 4.25

> mean(sqrt(prime))

[1] 2.007021
```

2 Plots

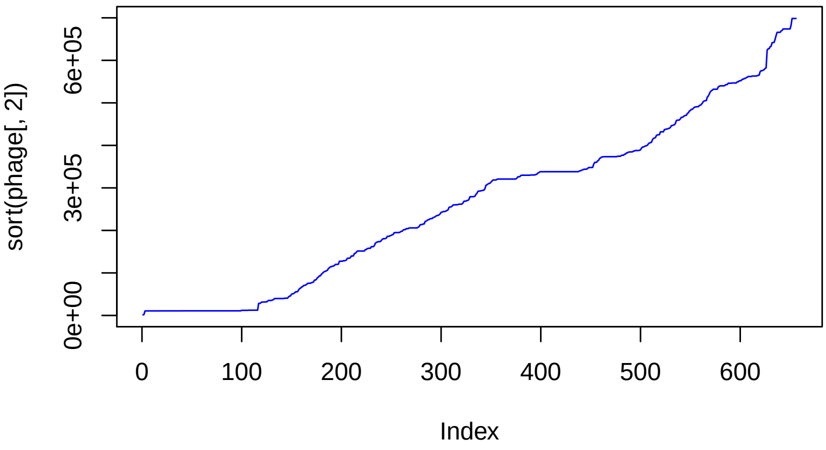
- Read the following file[1] into R: `ftp://ftp.ebi.ac.uk/pub/databases/genome_reviews/gr2species_phage.txt` and make the following plots. Check whether using a log10 scale on the y axis helps.

  ```
  > phage <- read.delim(file.path("ftp://ftp.ebi.ac.uk/pub/databases/genome_re
  +     header = F)
  ```

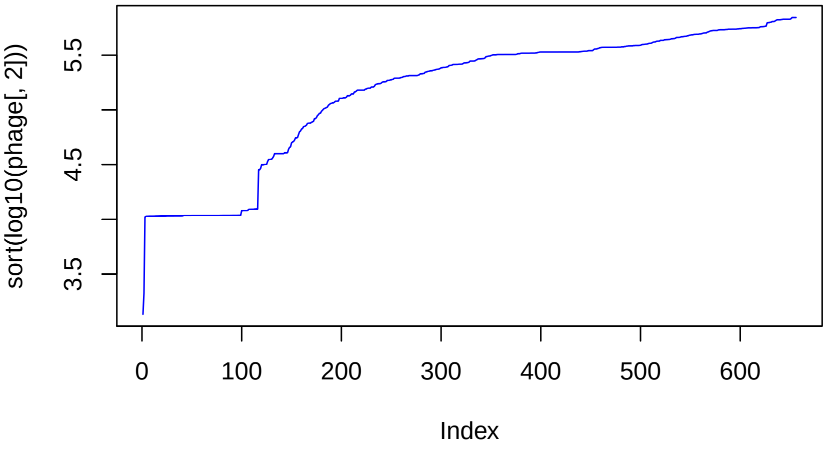
  1. Sort the genome sizes (column 2) and plot them in a line with increasing values.

     ```
     > plot(sort(phage[, 2]), type = "l",
     +     col = "blue")
     ```

[1] Look for the useful function for this case

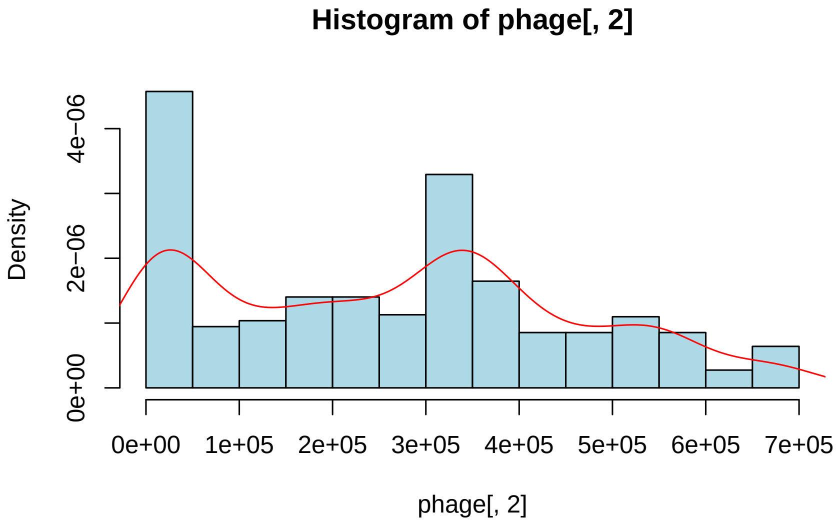


```
> plot(sort(log10(phage[, 2])), type = "l",
+      col = "blue")
> print("You can say that using log10 does help on this case")
[1] "You can say that using log10 does help on this case"
```



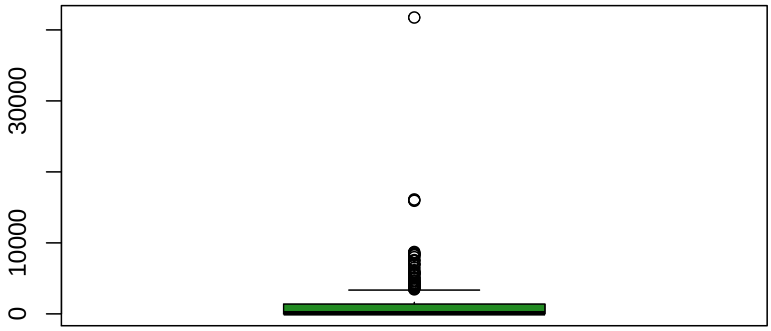
2. Plot a histogram with a density line for the same data.

```
> hist(phage[, 2], col = "light blue",
+      prob = T)
> lines(density(phage[, 2]), col = "red")
```



3. Plot a boxplot for the differences between contigous sorted genomes. Meaning, 2nd smallest - smallest, 3rd smallest - 2nd smallest, etc.[2]

```
> contig <- diff(sort(phage[, 2]))
> boxplot(contig, col = "forest green")
```

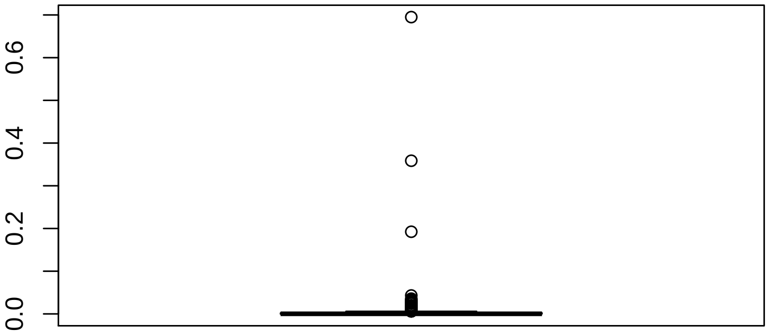


```
> contig <- diff(sort(log10(phage[,
+     2])))
```

[2] You might want to use apropos searching for diff. . .

```
> boxplot(contig, col = "forest green")
> print("Boxplot without log10 was more useful")
```

```
[1] "Boxplot without log10 was more useful"
```

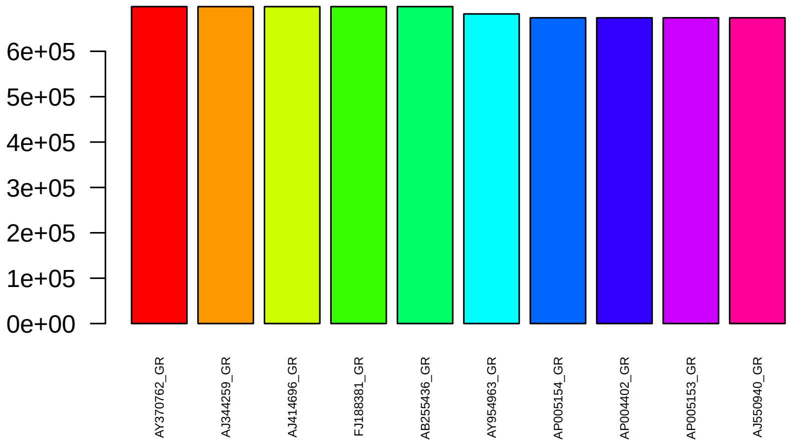


4. Make a barplot showing the 10 biggest genomes. Include the names[3] on the x axis and every bar has to have a different color and/or density.[4]

```
> top <- sort(phage[, 2], decreasing = T)[1:10]
> names <- NULL
> for (i in 1:10) {
+     names <- c(names, phage[which(phage[,
+         2] == top[i]), 1])
+ }
> barplot(top, col = rainbow(10),
+     names.arg = phage[names, 1],
+     cex.names = 0.5, las = 2)
```

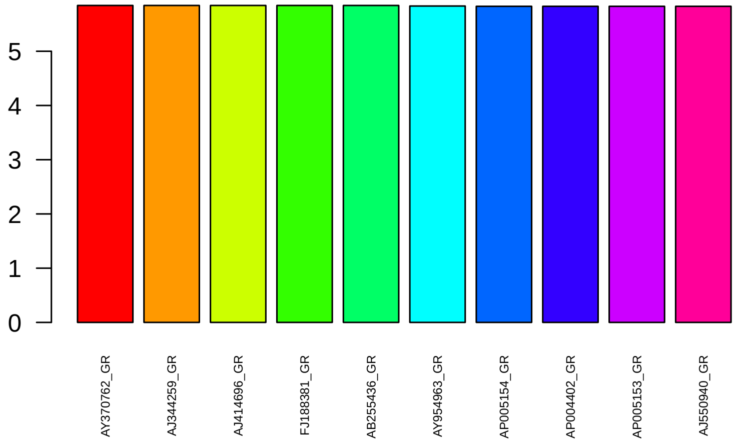
[3] They have to be redable

[4] The `which` function might be useful.



```
> barplot(log10(top), col = rainbow(10),
+     names.arg = phage[names, 1],
+     cex.names = 0.5, las = 2)
> print("Using log10 has almost no effect")
```

```
[1] "Using log10 has almost no effect"
```



3 Apply functions

1. What is the mean genome size for every type of replicon (column 4)? You have an atomic vector and a factor so use ...

```
> tapply(phage[, 2], phage[, 4],
+     mean)

Chromosome  Segment L  Segment M 
  268053.2   106813.8   106813.8 
 Segment S 
  106813.8 
```