R_Code_1.Rmd

Rohit Dhankar

20 July 2017

This is the 2nd in series of R Code Files.

Refer the GitHub Repository , for all Code files —> https://github.com/RohitDhankar/R-Beginners-Online-Virtual-Learning-S

Its a good practise from time to time to keep a track of our current Working Directory and list out all the Objects in our R ENVIRONMENT - specially so when we are committing changes to a Git Remote.

```
getwd()
```

```
## [1] "/home/dhankar/Desktop/R_Own/Proj_1"
```

```
#
ls()
```

```
## character(0)
```

We could remove any object with command - rm("Object Name")

We can also use print() , to view any objects stored value.

```
# Code Section -1
a1 <- "FINANCE"
b1 <- "MARKETING"
c1 <- "SALES"
d1 <- 3.1416
char_vector <- c("x","d","c","f")
print(a1)
```

```
## [1] "FINANCE"
```

```
#
print(char_vector)
```

```
## [1] "x" "d" "c" "f"
```

```
#
```

Going further with VECTORS .

We combine two or more vectors to get another vector .

```
# Code Section -2
num_vector <- c(22,22,33,33,44)
print(num_vector)
```

```
## [1] 22 22 33 33 44
```

```
num_vector1 <- c(11,12,13,14,15)
#
num_vector3 <- c(num_vector,num_vector1)
print(num_vector3)
```

```
##  [1] 22 22 33 33 44 11 12 13 14 15
```

Some basic Maths and Stats with VECTORS.

```
# Code Section -3
num_vector3 + 5
```

```
##  [1] 27 27 38 38 49 16 17 18 19 20
```

```
# Adds NUMERIC VALUE = 5 to all ELEMENTS of the Num Vector.
```

```
# Code Section -4

num_vector1 * num_vector3
```

```
##  [1] 242 264 429 462 660 121 144 169 196 225
```

```
# First 5 elements of - num_vector3 multiplied by the Five Elements
# of num_vector1 and again the Next 5 elements of num_vector3
# multiplied by the Five Elements of num_vector1
```

Check out the LENGTH of a VECTOR with length()

```
# Code Section -5

length(num_vector1 * num_vector3)
```

```
## [1] 10
```

```
# Code Section -6

#num_vector1 %*% num_vector3 # Error in num_vector1 %*% num_vector3 : non-conformable arguments

# Vectors are not of same Length above - below they are of same length

nv <- c(1,2,3,4,5)
nv1 <- c(6,7,8,9,10)

nv %*% nv1 # Inner Product of same Length Vectors
```

```
##      [,1]
## [1,]  130
```

```
# Algeberic Dot Product as defined by WikiPedia - "https://en.wikipedia.org/wiki/Dot_product"
```

Operate upon a ELEMENT of the Vector.

```
# Code Section -7


log(num_vector3[2]) # Log Base 2 of 22
```

```
## [1] 3.091042
```

```
#
log(22)
```

```
## [1] 3.091042
```

```
#
```

Converting a CHAR Vector into a NUMERIC Vector .

```
# Code Section -8
ch_v <- c("11","12","13","14","15")
```

```
#
class(ch_v)
```

```
## [1] "character"
```

```
#ch_v + 2 # Error in ch_v + 2 : non-numeric argument to binary operator
# Cant do a Math operation on CHAR Vector - lets Convert into NUM Vector
#
nm_v <- as.numeric(ch_v)
#
class(nm_v)
```

```
## [1] "numeric"
```

```
nm_v + 2
```

```
## [1] 13 14 15 16 17
```

```
#
#Summary of the Num Vector as below :-
#
summary(nm_v+2)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      13      14      15      15      16      17
```

```
#
summary(nm_v+5)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      16      17      18      18      19      20
```

```
#
sum(nm_v+5)
```

```
## [1] 90
```

```
#
sd(nm_v+5)
```

```
## [1] 1.581139
```

```
#
max(nm_v+5)
```

```
## [1] 20
```

```
#
min(nm_v+5)
```

```
## [1] 16
```

```
#
mean(nm_v+5)
```

```
## [1] 18
```

```
#
median(nm_v+5)
```

```
## [1] 18
```

```r
#
#The Quantile -
#
quantile(nm_v+5)
```

```
##   0%  25%  50%  75% 100% 
##   16   17   18   19   20
```

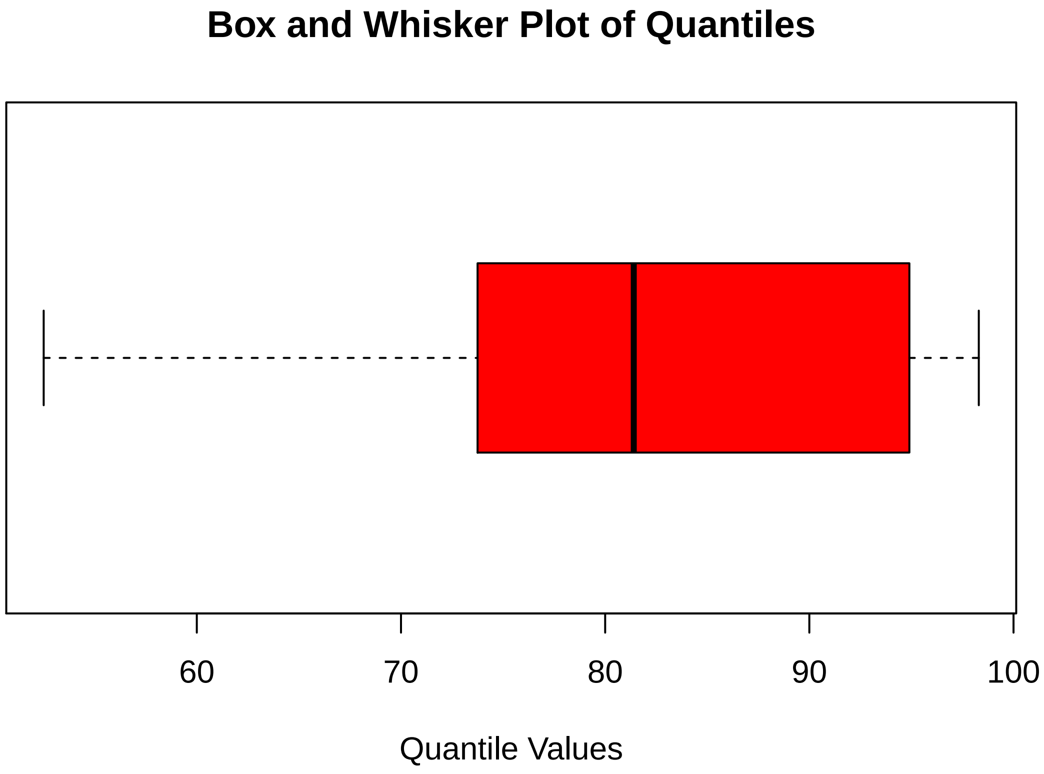
```r
#
quantile(nm_v+100)
```

```
##   0%  25%  50%  75% 100% 
##  111  112  113  114  115
```

```r
#
#We can also specify the Quantile buckets or Percentiles as an argument to the Quantile function :-
#
nmv_q <- c(10,15,20,25,30,35,40,45,50,55,60,65,70,75,80,85,90,100)
percent_1 <- quantile(nmv_q, c(.50,.75,.84, .97, .99))
percent_1
```

```
##   50%   75%   84%   97%   99% 
## 52.50 73.75 81.40 94.90 98.30
```

```r
boxplot(percent_1,col = "red",horizontal = TRUE,
        main = "Box and Whisker Plot of Quantiles",
        xlab = "Quantile Values")
```



```r
# Kindly note how the ARGUMENTs to boxplot()
# have been bumped to the next row - keeping in mind
# the Horizontal space of our PDF knit of the .Rmd file
# Seen above we have the MEDIAN quartile - 50% and the UPPER Quartile - 75% along with THREE more perce
```

Wiki reference – Percentile Rank - "https://en.wikipedia.org/wiki/Percentile_rank" #

We also carry out ONE Way ANOVA or ANALYSIS of VARIANCE test with the BOX and WHISKERS plots as seen below :-

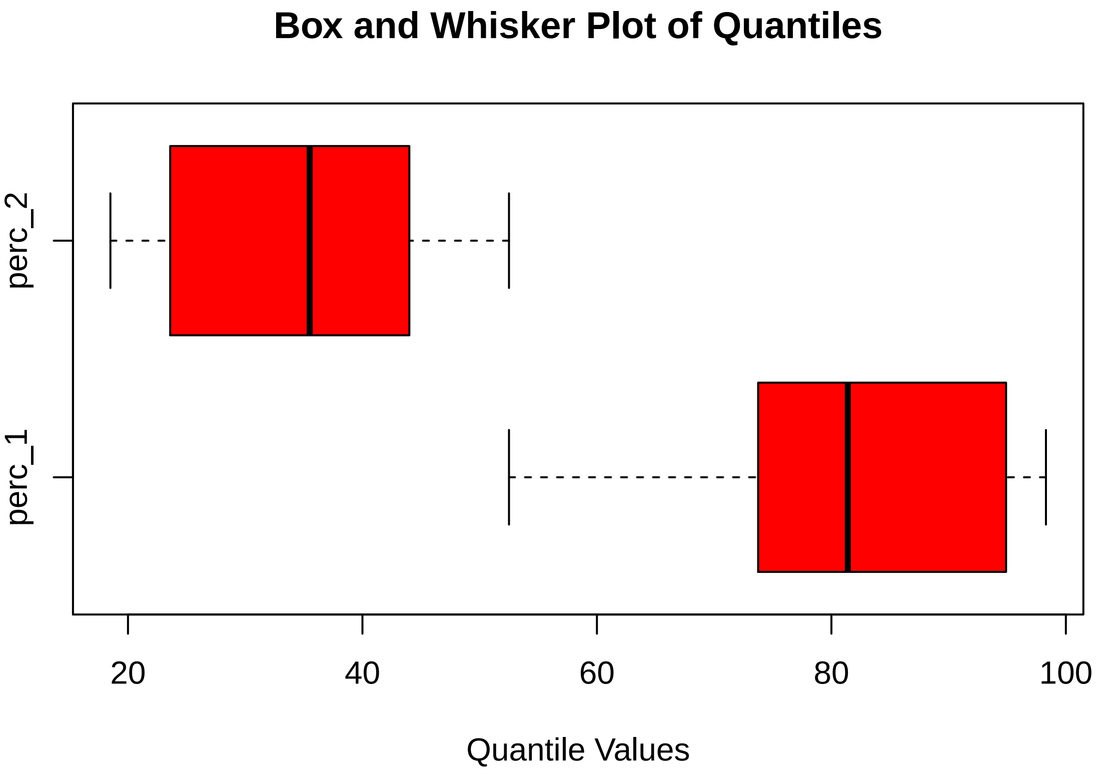
```
# Code Section -9

nmv_q <- c(10,15,20,25,30,35,40,45,50,55,60,65,70,75,80,85,90,100)
percent_1 <- quantile(nmv_q, c(.50,.75,.84, .97, .99))
percent_1
```

```
##   50%   75%   84%   97%   99%
## 52.50 73.75 81.40 94.90 98.30
```

```
percent_2 <- quantile(nmv_q, c(.1, .3, .16, .40, .50))
percent_2
```

```
##  10%  30%  16%  40%  50%
## 18.5 35.5 23.6 44.0 52.5
```

```
boxplot(percent_1,percent_2,col = "red",
        names = c("perc_1","perc_2"),horizontal = TRUE,
        main = "Box and Whisker Plot of Quantiles",
        xlab = "Quantile Values")
```



```
# Kindly note the Quantiles are randomly chosen here
# this is not the best way to choose quantiles
# we shall come back for details later in this text
# Code Section -10
```

```
# Code Section -11
```

```
# Code Section -12
```

```
# Code Section -13
```

```
# Code Section -14
```

```
# Code Section -15
```

```
# Code Section -16
```

```
# Code Section -17
```

sessionInfo()

R version 3.3.2 (2016-10-31) Platform: x86_64-pc-linux-gnu (64-bit) Running under: Ubuntu 16.04.1 LTS

locale: [1] LC_CTYPE=en_IN.UTF-8 LC_NUMERIC=C LC_TIME=en_IN.UTF-8 LC_COLLATE=en_IN.UTF-8
[5] LC_MONETARY=en_IN.UTF-8 LC_MESSAGES=en_IN.UTF-8 LC_PAPER=en_IN.UTF-8 LC_NAME=C
[9] LC_ADDRESS=C LC_TELEPHONE=C LC_MEASUREMENT=en_IN.UTF-8 LC_IDENTIFICATION=C

attached base packages: [1] stats graphics grDevices utils datasets methods base

loaded via a namespace (and not attached): [1] backports_1.0.4 magrittr_1.5 rprojroot_1.1 htmltools_0.3.5 tools_3.3.2 base64enc_0.1-3 yaml_2.1.14
[8] Rcpp_0.12.8 stringi_1.1.2 rmarkdown_1.3 knitr_1.15.1 jsonlite_1.1 stringr_1.1.0 digest_0.6.10
[15] evaluate_0.10

EOF - R_Code_1.Rmd