E-Content on Data Manipulation and Visualization through Statistical Software R (Open Source Software)

Dr Gurpreet Singh Tuteja and Dr Dhiraj Kumar Singh
Department of Mathematics
Zakir Husain Delhi College
(University of Delhi)
Jawaharlal Nehru Marg, Delhi - 110002

Day 2: July 28, 2020

1 Session 2

Today we begin with more concepts of R. We ave already learnt data types and now we begin with matrices which are two dimensional arrays.

1.1 Matrices

Matrices can also be created directly from vectors by adding a dimension attribute.
$m \leftarrow 1:10$
$dim(m) \leftarrow c(2,5)$
m
Matrices can be created by column-binding or row-binding with the cbind() and rbind() functions.
$x \leftarrow c(2,3)$
$y \leftarrow c(4,5)$
rbind(x,y)
cbind(x,y)
cbind in R appends or combines vector, matrix or data frame by columns. cbind() function in R appends or joins, two or more dataframes column wise. Same column bind operation can also be performed using bind_cols() function

of the dplyr package.
It can be created simply :

```
> x<- matrix(c(1,2,3,4,5,6),2,3)
> x
     [,1] [,2] [,3]
[1,]    1    3    5
[2,]    2    4    6
> y<- matrix(c(1,2,3,4,5,6),2,3,byrow=TRUE)
> y
     [,1] [,2] [,3]
[1,]    1    2    3
[2,]    4    5    6
```

1.2 Lists

Lists are a special type of vector that can contain elements of different classes. Lists are a very important data type in R and you should get to know them well. Lists, in combination with the various "apply" functions discussed later, make for a powerful combination. Lists can be explicitly created using the list() function, which takes an arbitrary number of arguments.
$x \leftarrow list(1, "a", TRUE, 1 + 4i)$
$x[2]$
We can also create an empty list of a pre-specified length with the vector() function
$x \leftarrow vector("list", length = 5)$ x

1.3 Factors

Factors are categorical variables that have a fixed number of levels. A simple example of a factor might be a variable called gender with
two levels: 'female' and 'male'. If you had three females and two males, you could create the factor like this:
$gender \leftarrow factor(c("female", "male", "female", "male", "female"))$
$x \leftarrow factor(c("yes", "yes", "no", "yes", "no"))$
gender
x

1.4 Missing Values

Missing values are denoted by NA or NaN for an undefined mathematical operations.

is.na() is used to test objects if they are NA
is.nan() is used to test for NaN
NA values have a class also, so there are integer NA, character NA, etc.
A NaN value is also NA but the converse is not true
$x \leftarrow c(1, 2, NaN, NA, 10, 3)$
$is.na(x)$
$is.nan(x)$

1.5 Data Frames

Data frames are used to store tabular data in R. They are an important type of object in R and are used in a variety of statistical modeling applications. Data frames are represented as a special type of list where every element of the list has to have the same length. Each element of the list can be thought of as a column and the length of each element of the list is the number of rows.
Unlike matrices, data frames can store different classes of objects in each column. Matrices must have every element be the same class (e.g. all integers or all numeric).In addition to column names, indicating the names of the variables or predictors, data frames have a special attribute called row.names which indicate information about each row of the data frame. Data frames are usually created by reading in a data set using the read.table() or read.csv().
However, data frames can also be created explicitly with the data.frame() function or they can be coerced from other types of objects like lists.Data frames can be converted to a matrix by calling data.matrix(). While it might seem that the as.matrix() function should be used to coerce a data frame to a matrix, almost always, what you want is the result of data.matrix().
$x \leftarrow data.frame(foo = 1:4, bar = c(T, T, F, F), Flag = c(\text{“}a\text{”}, \text{“}b\text{”}, \text{“}c\text{”}, \text{“}d\text{”}))$
x

1.6 Names

R objects can have names, which is very useful for writing readable code and self-describing objects. Here is an example of assigning names to an integer vector.
x ¡- 1:3
$names(x) \leftarrow c(\text{“}NewYork\text{”}, \text{“}Seattle\text{”}, \text{“}LosAngeles\text{”})$
?names
x
Column names and row names can be set separately using the colnames() and rownames() functions
$m \leftarrow matrix(1:4, nrow = 2, ncol = 2)$

m
$colnames(m) \leftarrow c(\text{“}h\text{”},\text{”}f\text{”})$
$rownames(m) \leftarrow c(\text{“}x\text{”},\text{”}z\text{”})$
m

Object Set	column names	Set row names
data frame	names()	row.names()
matrix	colnames()	rownames()

1.7 Simple Built-in Statistical Functions

$x \leftarrow 1:20$
sum(x)
mean(x)
var(x)
mad(x)
sd(x)
range(x)
quantile(x)
max(x)
min(x)
length(x)
which.max(x)
which.min(x)

```
> x <- 1:20
> x
 [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20
> sum(x)
[1] 210
> mean(x)
[1] 10.5
> var(x)
[1] 35
> sd(x)
[1] 5.91608
> range(x)
[1]  1 20
> quantile(x)
   0%   25%   50%   75%  100%
 1.00  5.75 10.50 15.25 20.00
> min(x)
[1] 1
> max(x)
[1] 20
> which.max(x)
[1] 20
> which.min(x)
[1] 1
```

We can apply the function log on x using sapply:
$y \leftarrow sapply(x, log)$
The co-variance and correlation of x and y are given as:
cov(x,y)
cor(x,y)
$x \leftarrow c(1:15, 2:20, 5:18)$
x
unique(x)
table(x)
mean(x)
$x \leftarrow rnorm(5)$
x
mean(x)
Consider another important Concept in R
$x^2 - 2$ [1]4.440892e−16 x ← sqrt(2) $x^2 == 2$ [1]$FALSE$ $x^2 - 2$ [1]4.440892e−16

1.8 Logical Arithmetic

Arithmetic involving logical expressions is very useful in programming and in selection of variables. If logical arithmetic is unfamiliar to you, then persevere with it, because it will become clear how useful it is, once the penny has dropped. The key thing to understand is that logical expressions evaluate to either true or false (represented in R by TRUE or FALSE), and that R can coerce TRUE or FALSE into numerical values: 1 for TRUE and 0 for FALSE.
$x \leftarrow 0:6$
$x < 4$
$all(x > 0)$
$any(x < 0)$
We can use the answers of logical functions in arithmetic sum(x¡6)
We can multiply $(x < 4)$ by other vectors:
$(x < 4)$
runif(7)
$(x < 4) * runif(7)$
Let us see simple algebra
0/0
[1]NaN

$Inf - Inf$
[1]NaN

Inf/Inf
[1]NaN

$is.finite(10)$
[1]$TRUE$
$is.infinite(Inf)$
[1]$TRUE$
Consider the following:
$vmv \leftarrow c(1:16, NA, NA, 9:12)$
vmv
[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 NA NA 9 10 11 12
$vmv \leftarrow seq(along = vmv)[is.na(vmv)]$
vmv
[1]17 18

Let us consider a vector:
$x \leftarrow c(1, 3, 4, 5, 2, 4)$
unique(x)
[1]13452
$x[duplicated(x)]$
[1]4
$x[!duplicated(x)]$
[1]13452.

This ends the second session. You are requested to appear in the quiz. Time allotted for the quiz is 15 mins.