Laboratory Exercise Week 2

Brian Tipton

8/30/23

Directions:

- Write your R code inside the code chunks after each question.
- Write your answer comments after the `#` sign.
- To generate the word document output, click the button `Knit` and wait for the word document to appear.
- RStudio will prompt you (only once) to install the `knitr` package.
- Submit your completed laboratory exercise using Blackboard's Turnitin feature. Your Turnitin upload link is found on your Blackboard Course shell under the Laboratory folder.

My Custom functions used from my local lab projects .Rprofile

```
source("../../.Rprofile", chdir = TRUE)
```

```
catXWithString
```

```
## function (string, x, nl = TRUE, sep = " ")
## {
##     if (nl) {
##         cat(paste(string, toString(x), "\n", sep = sep))
##     }
##     else {
##         cat(paste(string, toString(x), sep = sep))
##     }
## }
```

---

1. You will work with a matrix whose entries are all prime numbers below 30.
   i) Create this matrix using the function `matrix()` with five rows. Save the matrix as `P`.

   ii) Extract the second and third row out of `P`.

   iii) Extract the entry in the fourth row and first column of `P`.

   iv) Generate the transpose of the matrix using the function `t()`. What is the new dimension of this matrix.

Code chunk

```
P <- matrix(c(2, 3, 5, 7, 11, 13, 17, 19, 23, 29), nrow = 5)
P
```

```
##      [,1] [,2]
## [1,]    2   13
```

```
## [2,]    3   17
## [3,]    5   19
## [4,]    7   23
## [5,]   11   29
```

```
cat("SECOND AND THIRD ROWS: \n")
```

```
## SECOND AND THIRD ROWS:
```

```
P[c(2,3),]
```

```
##      [,1] [,2]
## [1,]    3   17
## [2,]    5   19
```

```
cat("\n")
```

```
P[4,1] |> catXWithString(string = "ITEM AT ROW 4 COLUMN 1:")
```

```
## ITEM AT ROW 4 COLUMN 1: 7
```

```
tP <- t(P)
cat("Transposed Matrix: \n")
```

```
## Transposed Matrix:
```

```
tP
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    2    3    5    7   11
## [2,]   13   17   19   23   29
```

```
cat("\n")
```

```
catXWithString(nrow(tP), string = "NUM ROWS:")
```

```
## NUM ROWS: 2
```

```
catXWithString(ncol(tP), string = "NUM COL:")
```

```
## NUM COL: 5
```

2. Create and manipulate a data frame.
   i) Create a `data.frame` named `my.trees` that has the following columns:
      - `Girth = c(8.3, 8.6, 8.8, 10.5, 10.7, 10.8, 11.0)`
      - `Height = c(70, 65, 63, 72, 81, 83, 66)`
      - `Volume = c(10.3, 10.3, 10.2, 16.4, 18.8, 19.7, 15.6)`

   ii) Extract the fifth observational unit.

   iii) Extract the `Girth` column referring to it by name.

   iv) Print out a data frame of all the observations except for the last observation.

Code chunk

```
my.trees <-
  data.frame(
    Girth = c(8.3, 8.6, 8.8, 10.5, 10.7, 10.8, 11.0),
```

```
    Height = c(70, 65, 63, 72, 81, 83, 66),
    Volume = c(10.3, 10.3, 10.2, 16.4, 18.8, 19.7, 15.6)
  )
cat("5th Observational Unit")
```

```
## 5th Observational Unit
```

```
my.trees[5,]
```

```
##   Girth Height Volume
## 5  10.7     81   18.8
```

```
cat("\n")
```

```
cat("GIRTH COLUMN: ")
```

```
## GIRTH COLUMN:
```

```
my.trees["Girth"]
```

```
##   Girth
## 1   8.3
## 2   8.6
## 3   8.8
## 4  10.5
## 5  10.7
## 6  10.8
## 7  11.0
```

```
cat("\n")
```

```
cat("ALL EXCEPT LAST OBSERVATION: ")
```

```
## ALL EXCEPT LAST OBSERVATION:
```

```
my.trees[1:(nrow(my.trees) - 1),]
```

```
##   Girth Height Volume
## 1   8.3     70   10.3
## 2   8.6     65   10.3
## 3   8.8     63   10.2
## 4  10.5     72   16.4
## 5  10.7     81   18.8
## 6  10.8     83   19.7
```

```
cat("\n")
```

3. The popular `iris` data set gives the measurements in centimeters of the variables sepal length and width and petal length and width, respectively, for 50 flowers from each of 3 species of iris. The species are Iris setosa, versicolor, and virginica. The `iris` data set is included with every R installation.
   i) Check the structure of the`iris` data.

   ii) How many variables and observations are in the data set?

   iii) Which variables are `numeric` type?

   iv) Display the first 4 rows of the data.

Code chunk

```
ncol(iris) |> catXWithString(string = "How many variables: ")
```

```
## How many variables:  5
```

```
nrow(iris) |> catXWithString(string = "How many observations: ")
```

```
## How many observations:  150
```

```
is.numeric(iris$Sepal.Length) |> catXWithString(string = "Sepal.Length is Numeric: ")
```

```
## Sepal.Length is Numeric:  TRUE
```

```
is.numeric(iris$Sepal.Width) |> catXWithString(string = "Sepal.Width is Numeric: ")
```

```
## Sepal.Width is Numeric:  TRUE
```

```
is.numeric(iris$Petal.Length) |> catXWithString(string = "Petal.Length is Numeric: ")
```

```
## Petal.Length is Numeric:  TRUE
```

```
is.numeric(iris$Petal.Width) |> catXWithString(string = "Petal.Width is Numeric: ")
```

```
## Petal.Width is Numeric:  TRUE
```

```
is.numeric(iris$Species) |> catXWithString(string = "Species is Numeric: ")
```

```
## Species is Numeric:  FALSE
```

```
cat("FIRST 4 ROWS: ")
```

```
## FIRST 4 ROWS:
```

```
iris[1:4,]
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
## 4          4.6         3.1          1.5         0.2  setosa
```

4. You will work with the passenger Titanic data found on this link.
   i) Read this data directly from the web using its URL.

   ii) Read this data by saving it first into your working directory. Check your working directory using `getwd()` or change its location using the steps covered in the lesson.

   iii) How many passengers and what variables are in the data set?

   iv) Display the first 5 rows of the data.

Code chunk

```
titanic.remote <- read.csv("https://goo.gl/NHb1Pg")
titanic.local <- read.csv("./titanic.csv")
nrow(titanic.remote) |> catXWithString(string = "NUM PASSENGERS: ")
```

```
## NUM PASSENGERS:  1316
```

```r
names(titanic.remote) |> catXWithString(string = "VARIABLES: ")
```

```
## VARIABLES:  X, class, age, sex, survived
```

```r
titanic.remote[1:5,]
```

```
##   X     class    age sex survived
## 1 1 1st class adults man      yes
## 2 2 1st class adults man      yes
## 3 3 1st class adults man      yes
## 4 4 1st class adults man      yes
## 5 5 1st class adults man      yes
```