DATA202/STAT483 Assignment 2

Due: Thursday 8 August 2019, Worth 15%

Instructions

- Prepare your assignment using Rmarkdown.
- Submit your solutions in a single file named `assignment2.Rmd` through the ECS submission system
- The YAML header of your Rmarkdown file must contain your name and ID number in the author field, and should have the output format set to `pdf_document`. For example:

```
---
title: "DATA202 Assignment 2"
name: "Richard Arnold. 30XXXXXXX"
date: "25 July 2019"
output: "pdf_document"
---
```

- While you are developing your code you may find it easiest to have the output set to `html_document`, but change it to `pdf_document` when you submit.
- In your submission embed any executable R code in code chunks, and make sure both the R code and the output is displayed correctly when you knit the document.
- If there are any R code errors, then the Rmarkdown file will not knit, and no output will be created at all. So if you can't get your code to work, but want to show your attempted code, then put `eval=FALSE` in the header of the R code chunk that is failing.

```
```{r eval=FALSE}
your imperfect R code
```
```

- Where appropriate, make sure you include your comments on the output within the R markdown document.
- Make sure you check the `.pdf` file that is created when you submit, to see if it has knitted correctly with no errors.
- It can be difficult to get the automated numbering to work in Rmarkdown in a long document like this. So title each question answer with its question number as `Q1.`, `Q2.`, … etc. instead of `1.`, `2.`, … .

Assignment Questions

Q1. (6 Marks) The following function takes a scalar `x`, and calculates `x!`, i.e. the factorial of `x`.

```
fact <- function(x) {
    ans <- 1
    for(i in 1:x) {
      ans <- ans*i
    }
    return(ans)
}
```

a. The function computes its answer by building the result up using a `for()` loop - briefly explain how this works.

b. What does the `return()` function do?
c. Why must `x` be an integer greater than 0? (does it work for `x=0` ?)
d. Modify the function to give the correct value 1 when `x` is zero, and an error message when `x` is negative. Show the R code of the modified function.

Q2. (2 Marks) Use `sprintf()` to print out the result of `fact()`, so that the following code

```
for(x in 1:10) {
    s <- ... Your R code using sprintf() here ...
    cat(s)
}
```

produces the output

```
Factorial  1 is        1
Factorial  2 is        2
Factorial  3 is        6
Factorial  4 is       24
Factorial  5 is      120
Factorial  6 is      720
Factorial  7 is     5040
Factorial  8 is    40320
Factorial  9 is   362880
Factorial 10 is  3628800
```

(**Note:** don't worry if you can't get the first line to align exactly with the others.)

Q3. (5 Marks) The following function scrapes all arrivals from the arrivals board at Wellington Airport

```
get.arrivals <- function() {
    library(httr)
    library(XML)
    url <- "https://www.wellingtonairport.co.nz/flights/"
    mynrow <- function(x) ifelse(is.null(nrow(x)),NA,nrow(x))
    doc <- htmlParse(rawToChar(GET(url)$content))
    tabs <- readHTMLTable(doc)
    n.rows <- sapply(tabs, mynrow)
    atab <- tabs[[which.max(n.rows)]]
    atab <- atab[!is.na(atab$Airline),]
    names(atab) <- gsub(" ","",(gsub("\n","",names(atab))))
    return(atab)
}
arr <- get.arrivals()
arr[1:3,]
```

```
##       From Sched.  Est.         Airline Flight Gate          Status
## 2 Blenheim  11:55 11:45      Sounds Air  S8276      Arrived at Gate
## 3   Nelson  12:00 11:49 Air New Zealand NZ8308   10 Arrived at Gate
## 4 Tauranga  12:05 11:56 Air New Zealand NZ5255   18 Arrived at Gate
```

a. Explain what the following statement does, and why it is necessary here.

```
library(httr)
```

b. By modifying the function `get.arrivals()`, write a function called `get.departures()` that it scrapes departures instead.
c. Modify `get.arrivals()` so that it takes one argument: `origin`. If `origin` is specified it should give flight arrivals only from that place. If `origin` is `All` it should show all arrivals. Show the code, and the output of `get.arrivals("Christchurch")`.

Q4. (9 Marks) Write a function, showing all R code in your answer, with the following properties:

- Three arguments, one being a **data frame**, the next being the **name of a variable** in that data frame, and the third being a **filename**;
- The function should plot a **boxplot** of the variable if it is numeric, and a **barplot** if it is character or if it is a factor
- The function should remove any missing values in the specified column
- It should return the **number of non-missing values** in the that column
- If the filename is `NA` then it should draw the plot to the screen, otherwise it should write the output to a jpeg graphics file with the specified name. (Remember if you do write to a jpeg file, then you need to close the file before the end of the function.)
- For no extra marks, use the … argument to pass extra arguments (such as axis labels and a title) to the barplot and histogram functions.
- Demonstrate the function by
    - Making a boxplot of the `Total` variable in the `allpokemon` dataset
    - Making a bar plot of the `Marital` variable in the `surf` dataset For each of these show the diagram you get when outputting to the screen, and also run the function to create two appropriately named jpeg files. Run `list.files()` to show that these two files have been created.

Q5. (17 Marks) The New Zealand Department of Internal Affairs maintains records of the names of babies born in New Zealand, and releases statistics on the occurrences of those names. Data on baby names from the website www.data.govt.nz “Baby name popularity over time” is available in the file `babynames.csv` on the course website.

- Read the dataset into an R data frame called `babynames`
- Summarise basic characteristics of the data set: column names, data types, and the number of rows in the data set. State clearly what each column contains. You will need to go to the source website www.data.govt.nz to find some of this information.
- How many babies are registered in the data set in all? Why is this less than the total number of babies born in New Zealand in the same period?
- Draw a suitably labelled barplot of the number of babies recorded in the dataset per year in the dataset.
- Draw a barplot of the number of distinct names recorded per year in the data set
- What was the most common baby name registered in 2018 and how often was it used? (There are various ways to do this in R: an easy first step is to create a dataset with just contains the 2018 data.)
- What was the most common baby name for girls registered in 1900? How often was it used?
- Choose a name from the data set (it could be your own, or a name you like, or a random name), and plot the number of baby registrations with that name in each of the years in the data set.
- Choose one further name, and plot the two diagrams side by side (you’ll need to use a call to `par()`.) Use the `xlim` and `ylim` arguments in the `plot()` function to ensure that the two diagrams have the same horizontal and vertical axes.
- Write an R function called `plotnames` which takes three arguments: the name of the data frame containing the baby names data, and two strings, `name1` and `name2` and creates the same side-by-side plot as in the previous question. So to make side-by-side plots of “Richard” and “Alan” you should be able to type

`plotnames(babynames, "Richard", "Alan")`.

- Make sure each plot frame is labelled with a title saying which name it is for.
- **Make sure the function stops with an error message** if either of the names is not present in the data set.
- The function should return the maximum number of occurrences of the two names as a two component vector.
- Demonstrate the function with the names “Hannah” and “Karen”.

(Assignment total: 39 Marks)