Assignment 6

Chase Darlington

October 18, 2018

BSDS 100: Intro to Data Science with R

Assignment 6

by Chase S Darlington (University of San Francisco)

Directions: For all questions in this assignment, write complete sentences and fully answer any question

that is asked, and use R to answer each question. Provide all R code and solutions by knitting your final RStudio file into a single file named your name CA6.pdf. Late assignments will automatically have 10 points deducted, if submitted within a week of the due date. Assignments submitted after the answer key is posted will not be accepted and will receive zero points.

1. (2 pts) What is the advantage of storing (and loading) data as a .csv file rather than a .xlsx file?

- .csv files are comma delimited and thus lack a strict dat structure. Consequently, .csv files are flexible. .xlsx files, on the other hand have a defined structure and are not as readily flexible as .csv files. Therefore, .csv files are preferred for storing and loading data because they are more flexible and easier to work with across most programming languages.

2. (6 pts) Name three different types of data sets that you may be interested in loading into R and describe at least one function that can be used to input each of the types of data set. (2 pts per example)

- CSV
- read.csv(path) can translate these data sets into data frames if the data set is in the form of a csv file. read.table() and read.csv2() can do essentially the same. Then, the data can be manipulated from there.
- TXT
- read.table(path)
- HTML
- readHTMLtable(url, which=“”)
- OR use getURL() and then readHTMLtable()
- XML
- xmlTreeParse(url)

```r
data <- read.csv("C:\\Users\\Chase Darlington\\Downloads\\sqllab_chase_darlington_ica_3_20180915T015124
head(data)
```

```
##             V1          V2                V3
## 1   event_name      device number_of_events
## 2    home_page macbook pro            12675
## 3 like_message macbook pro             8161
## 4   view_inbox macbook pro             7588
## 5        login macbook pro             5579
## 6 send_message macbook pro             4413
```

3. Answer the following questions.

(a) (2 pts) Create a data frame that has the following four columns:

- Numbers: the numbers 1 through 50, where each number is repeated twice in a row. (e.g. 1 1 2 2 3 3 . . .)
- Logicals: a vector of length 100 whose jth entry is TRUE if the jth entry of Numbers is even and FALSE if the jth entry of Numbers is odd.
- Rev.Numbers: the vector Numbers but in reverse order.
- Weirdness: the sum of Logicals and Rev.Numbers.

```
df <- data.frame(NA, NA, NA, NA, NA)
df <- data.frame(1:100, round(seq.int(1,50,length.out=100)), ifelse(1:100%%2==0, TRUE, FALSE), round(se
colnames(df) <- c("RowNum", "Numbers", "Logicals", "Rev. Numbers", "Weirdness")
df <- data.frame(1:100, round(seq.int(1,50,length.out=100)), ifelse(1:100%%2==0, TRUE, FALSE), round(se
colnames(df) <- c("RowNum", "Numbers", "Logicals", "Rev. Numbers", "Weirdness")
head(df)
```

```
##   RowNum Numbers Logicals Rev. Numbers Weirdness
## 1      1       1    FALSE           50        50
## 2      2       1     TRUE           50        51
## 3      3       2    FALSE           49        49
## 4      4       2     TRUE           49        50
## 5      5       3    FALSE           48        48
## 6      6       3     TRUE           48        49
```

(b) (2 pts) What are the data types for each of these columns?

```
sapply(df, class)
```

```
##       RowNum      Numbers     Logicals Rev. Numbers    Weirdness
##    "integer"    "numeric"    "logical"    "numeric"    "numeric"
```

(c) (2 pts) Describe why the variable Weirdness is an Integer variable.

- The logicals are interpreted as binary (False=0, True=1), and the Weirdness column is computed accordingly.

(d) (2 pts) Save this data frame to any chosen directory as a .RData object named MyDataFrame.

```
save(df, file="MyDataFrame.RDa")
```

(e) (2 pts) Remove the data from your workspace, then reload MyDataFrame and print out the first 6 entries in each column of the data frame.

```
rm(df)
rm()

load("MyDataFrame.RDa")
head(df)
```

```
##   RowNum Numbers Logicals Rev. Numbers Weirdness
## 1      1       1    FALSE           50        50
## 2      2       1     TRUE           50        51
## 3      3       2    FALSE           49        49
## 4      4       2     TRUE           49        50
## 5      5       3    FALSE           48        48
## 6      6       3     TRUE           48        49
```

4. Load the Airport data that we investigated in the Input Output Lecture. Then write code to answer each of the following:

```
airports <- read.csv(file = "https://raw.githubusercontent.com/abbiepopa/bsds100/master/Data/airports.c
head(airports)
```

```
##   iata              airport             city state country      lat
## 1  00M              Thigpen      Bay Springs    MS     USA 31.95376
## 2  00R Livingston Municipal       Livingston    TX     USA 30.68586
## 3  00V          Meadow Lake Colorado Springs    CO     USA 38.94575
## 4  01G         Perry-Warsaw            Perry    NY     USA 42.74135
## 5  01J     Hilliard Airpark         Hilliard    FL     USA 30.68801
## 6  01M    Tishomingo County          Belmont    MS     USA 34.49167
##        long
## 1  -89.23450
## 2  -95.01793
## 3 -104.56989
## 4  -78.05208
## 5  -81.90594
## 6  -88.20111
```

```
summary(airports)
```

```
##       iata                       airport                 city          state     
##  00M    :   1   Jackson County    :   5   Greenville:  11   AK     : 263  
##  00R    :   1   Monroe County     :   5   Houston   :  10   TX     : 209  
##  00V    :   1   Municipal         :   5   Jackson   :  10   CA     : 205  
##  01G    :   1   Franklin County   :   4   Columbus  :   9   OK     : 102  
##  01J    :   1   Lancaster         :   4   Madison   :   8   FL     : 100  
##  01M    :   1   Plymouth Municipal:   4   (Other)   :3316   (Other):2485  
##  (Other):3370   (Other)           :3349   NA's      :  12   NA's   :  12  
##                            country          lat              long        
##  Federated States of Micronesia:   1   Min.   : 7.367   Min.   :-176.65  
##  N Mariana Islands             :   1   1st Qu.:34.688   1st Qu.:-108.76  
##  Palau                         :   1   Median :39.434   Median : -93.60  
##  Thailand                      :   1   Mean   :40.037   Mean   : -98.62  
##  USA                           :3372   3rd Qu.:43.373   3rd Qu.: -84.14  
##                                        Max.   :71.285   Max.   : 145.62  
## 
```

(a) (2 pts) What are the names of the variables in this data set and what are their data types?

```
sapply(airports, class)
```

```
##      iata   airport      city     state   country       lat      long 
##  "factor"  "factor"  "factor"  "factor"  "factor" "numeric" "numeric"
```

(b) (2 pts) What is the mean and standard deviation of the longitude of these airports?

```
mean(airports$long)
```

```
## [1] -98.6212
```

```
sd(airports$long)
```

```
## [1] 22.86946
```

(c) (2 pts) What is the minimum and maximum latitude of these airports?

```
min(airports$lat)
```

```
## [1] 7.367222
```

```
max(airports$lat)
```

```
## [1] 71.28545
```

(d) (2 pts) Which airport has the minimum latitude? The maximum latitude?

```
airports[match(min(airports$lat), airports$lat),]
```

```
##      iata           airport city state country      lat     long
## 2796  ROR Babelthoup/Koror <NA>  <NA>   Palau 7.367222 134.5442
```

```
airports[match(max(airports$lat), airports$lat),]
```

```
##      iata                         airport   city state country      lat
## 1004  BRW Wiley Post Will Rogers Memorial Barrow    AK     USA 71.28545
##          long
## 1004 -156.766
```

(e) (2 pts) Add a new observation (row) to this data frame. Add whatever you would like as the new input, but make sure that each variable maintains its original data type. (i.e. if the longitude variable is numeric, make sure that it remains numeric after the new observation is added).

```
newdata <- data.frame(factor("USF"), factor("BSDS Airport"), factor("San Francisco"), factor("CA"), fac
colnames(newdata)<-c("iata", "airport", "city", "state", "country", "lat", "long")
sapply(newdata, class)
```

```
##      iata   airport      city     state   country       lat      long
##  "factor"  "factor"  "factor"  "factor"  "factor" "numeric" "numeric"
```

```
newdata
```

```
##   iata      airport          city state country     lat     long
## 1  USF BSDS Airport San Francisco    CA     USA 37.7765 122.4506
```

```
newairports <- rbind(airports, newdata)
head(newairports)
```

```
##   iata              airport             city state country      lat
## 1  00M             Thigpen       Bay Springs    MS     USA 31.95376
## 2  00R Livingston Municipal       Livingston    TX     USA 30.68586
## 3  00V          Meadow Lake Colorado Springs    CO     USA 38.94575
## 4  01G         Perry-Warsaw            Perry    NY     USA 42.74135
## 5  01J     Hilliard Airpark         Hilliard    FL     USA 30.68801
## 6  01M    Tishomingo County          Belmont    MS     USA 34.49167
##         long
## 1  -89.23450
## 2  -95.01793
## 3 -104.56989
## 4  -78.05208
## 5  -81.90594
## 6  -88.20111
```

```
newairports[which(newairports$iata=="USF"),]
```

```
##      iata      airport          city state country     lat     long
## 3377  USF BSDS Airport San Francisco    CA     USA 37.7765 122.4506
```

(f) (2 pts) Save your new data frame as a .csv, a .txt, and a .RData file.

```
save(newairports, file = "Airports.csv")
save(newairports, file = "Airports.txt")
save(newairports, file = "Airports.RDa")
```