Intro to R for Epidemiologists

Lab 2 (1/22/15)

Many of these questions go beyond the information provided in the lecture. Therefore, you may need to use R help files and the internet to search for answers. Feel free to ask questions of the instructor, the TAs, or your classmates, but try to work through as much as you can independently.

For the lab, you are expected to create an R script (.R file in the R editor) with your code corresponding to each question. Begin each question with a commented line of code indicating the question. As an example:

```
# Jenna Krall

# Question 1.
head(iris)
```

Part 1. Google flu trends

This data, introduced in lecture, estimates 2013 US flu activity using google searches. The data includes estimated daily flu activity for the whole United States, Georgia, Atlanta, and Health and Human Services Region 4 (which includes Alabama, Florida, Georgia, Kentucky, Mississippi, North Carolina, South Carolina, and Tennessee). Google published their results in Nature (http://www.nature.com/nature/journal/v457/n7232/full/nature07634.html). An article in Science also discusses problems with using this data to predict flu activity (http://www.sciencemag.org/content/343/6176/1203).

Data Source: Google Flu Trends (http://www.google.org/flutrends)

1. Download the dataset "googleflumissing.txt" from www.jennakrall.com/IntrotoRepi/googleflumissing.txt . Open the file in a text editor (e.g. TextEdit on a Mac, Notepad in Windows). What separates the entries in this dataset? See the help page in R for `read.table` and read this dataset into R.

```
# Read in flu data, which has tab separated values
flu <- read.table("googleflumissing.txt", sep = "\t", stringsAsFactors = F,
    header = T)
# Read in flu data using read.delim
flu <- read.delim("googleflumissing.txt", stringsAsFactors = F)
```

2. Find the mean Georgia flu activity. What additional argument do you need?

```
# Find mean
mean(flu$Georgia)
```

```
## [1] NA
```

```
mean(flu$Georgia, na.rm = F)
```

```
## [1] NA
```

```r
# Find mean excluding missing
mean(flu$Georgia, na.rm = T)
```

```
## [1] 2520.404
```

3. Apply the `is.na` function to Georgia flu activity. Save the results of `is.na` to a new vector `na_georgia`. What class is `na_georgia`?

```r
# Which items are NA
na_georgia <- is.na(flu$Georgia)
head(na_georgia)
```

```
## [1]  TRUE  TRUE  TRUE  TRUE  TRUE FALSE
```

```r
# Find class of na_georgia
class(na_georgia)
```

```
## [1] "logical"
```

4. Take the mean of your vector `na_georgia`. What value does it give and what does this correspond to?

```r
# Mean of logical vector is the proportion of trues
mean(na_georgia)
```

```
## [1] 0.8575342
```

```r
# table of na_georgia
table(na_georgia)/sum(table(na_georgia))
```

```
## na_georgia
##     FALSE      TRUE
## 0.1424658 0.8575342
```

5. Create a vector `notna_georgia` that indicates the values that are not missing Georgia flu activity. Use your vector to create a vector of the Georgia flu activity for only those days that are not missing.

```r
# Is Georgia flu data not missing
notna_georgia <- !(is.na(flu$Georgia))
# Subset Georgia flu data based on which not missing
ga_notmiss <- flu$Georgia[notna_georgia]
# Compare to original flu data
head(ga_notmiss)
```

```
## [1] 10380 10231  7813  6294  5344  4755
```

```r
head(flu$Georgia)
```

```
## [1]    NA    NA    NA    NA    NA 10380
```

6. Create a revised Google flu dataset to include only the days where Atlanta had at least 1,000 flu cases.

```r
# Subset flu where Atlanta >= 1000
flu_g1000 <- flu[flu$Atlanta >= 1000, ]
head(flu_g1000)
```

```
##            Date United.States Georgia Atlanta HHSRegion4 flu_high
## NA         <NA>            NA      NA      NA         NA     <NA>
## NA.1       <NA>            NA      NA      NA         NA     <NA>
## NA.2       <NA>            NA      NA      NA         NA     <NA>
## NA.3       <NA>            NA      NA      NA         NA     <NA>
## NA.4       <NA>            NA      NA      NA         NA     <NA>
## 6    2013-01-06         10112   10380   10455       9060     High
```

```r
# Subset flu where Atlanta >= 1000
flu_g1000 <- subset(flu, subset = Atlanta >= 1000)
head(flu_g1000)
```

```
##          Date United.States Georgia Atlanta HHSRegion4 flu_high
## 6  2013-01-06         10112   10380   10455       9060     High
## 13 2013-01-13         10555   10231    9976       8722     High
## 20 2013-01-20          9408    7813    8823       6387     High
## 27 2013-01-27          7209    6294    7383       5132     High
## 34 2013-02-03          5541    5344    5919       4237     High
## 41 2013-02-10          4555    4755    5230       3655     High
```

7. Find the mean flu activity in Georgia when the Atlanta flu activity is above its median.

   · Hint: break this question down into steps,

   a. Find the median of Atlanta flu activity
   b. Find the rows when Atlanta flu activity exceeds its median
   c. Create a vector of Georgia flu activity with the rows in (b.).
   d. Take the mean of the vector in (c.).

```r
# We can do this by steps: 1. Find median flu in Atlanta
med_atl <- median(flu$Atlanta, na.rm = T)
med_atl
```

```
## [1] 1410
```

```r
# 2. Find which Atlanta flu activity days are greater than the median
atl_above_med <- which(flu$Atlanta > med_atl)
head(atl_above_med)
```

```
## [1]  6 13 20 27 34 41
```

```r
# 3. Vector of GA flu activity meeting criteria in 2
ga_restrict <- flu$Georgia[atl_above_med]
# 4. Find mean
mean(ga_restrict)
```

```
## [1] 4145.115
```

```r
# Or can do it in one step:
mean(flu$Georgia[which(flu$Atlanta > median(flu$Atlanta, na.rm = T))])
```

```
## [1] 4145.115
```

8. The flu season is generally from October through April. Create a new variable indicating when it is flu season. Specifically, your variable should be TRUE when it is flu season and FALSE when it is not. Hint: To create a date, use `as.Date`. You may also need to look again at logical operators in R (`?"|"`).

```r
# 1. Which dates are before May 1, 2013 or after October 1, 2013
start <- as.Date("2013-05-01")
end <- as.Date("2013-10-01")
is_fluseason <- (flu$Date < start | flu$Date >= end)
```

Part 2. FIFA World Cup goals

The FIFA (Fédération Internationale de Football Association) World Cup is an international men's soccer (football) competition held every four years. We are interested in the total number of goals scored in the past four touraments for the USA, Germany, Brazil, and Italy.

1. Create a matrix in R of total goals scored that has the following information,

$$\begin{bmatrix} 7 & 14 & 18 & 5 \\ 2 & 14 & 10 & 12 \\ 5 & 16 & 9 & 4 \\ 5 & 18 & 11 & 2 \end{bmatrix}$$

   Hint: Use the `matrix` function.

```r
# Entering data by hand
men <- matrix(data = c(7, 14, 18, 5, 2, 14, 10, 12, 5, 16, 9, 4, 5, 18, 11,
    2), nrow = 4, byrow = T)
men
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    7   14   18    5
## [2,]    2   14   10   12
## [3,]    5   16    9    4
## [4,]    5   18   11    2
```

2. Name the columns of your matrix "USA", "Germany", "Brazil", "Italy." Name the rows "2014", "2010", "2006", "2002" using the `seq()` command.

```r
# Add column and rownames
colnames(men) <- c("USA", "Germany", "Brazil", "Italy")
rownames(men) <- seq(2002, 2014, by = 4)
```

3. Remove the column corresponding to Brazil and print the new matrix.

```r
# Select the remaining columns
men2 <- men[, c("USA", "Germany", "Italy")]
men2 <- men[, colnames(men) != "Brazil"]
men2
```

```
##      USA Germany Italy
## 2002   7      14     5
## 2006   2      14    12
## 2010   5      16     4
## 2014   5      18     2
```

4. Provide the mean and median of total goals scored over the entire time period for Germany.

```r
# Mean and median goals for Germany
mean(men[, "Germany"])
```

```
## [1] 15.5
```

```r
median(men[, "Germany"])
```

```
## [1] 15
```

5. Order the matrix you created so that Italy's goals are organized in decreasing order.

```r
# Order the goals for Italy
ord_italy <- order(men[, "Italy"], decreasing = T)
men[ord_italy, ]
```

```
##      USA Germany Brazil Italy
## 2006   2      14     10    12
## 2002   7      14     18     5
## 2010   5      16      9     4
## 2014   5      18     11     2
```