

1 Getting and Cleaning Data: Week 1 Quiz

```
install.packages("data.table", "xlsx", "XML")
```

Question 1

The American Community Survey distributes downloadable data about United States communities. Download the 2006 microdata survey about housing for the state of Idaho using `download.file()` from here:

https://dl.dropbox.com/u/7710864/data/csv_hid/ss06hid.csv

or here:

<https://spark-public.s3.amazonaws.com/dataanalysis/ss06hid.csv>

and load the data into R. You will use this data for the next several questions.

Code Book

The code book, describing the variable names is here:

<https://dl.dropbox.com/u/7710864/data/PUMSDataDict06.pdf>

or here:

<https://spark-public.s3.amazonaws.com/dataanalysis/PUMSDataDict06.pdf>

How many housing units in this survey were worth more than \$1,000,000?

```
# Download 2006 microdata survey
# re: housing for Idaho using download.file()
# setwd("~/DA")
download.file(
  'https://spark-public.s3.amazonaws.com/dataanalysis/ss06hid.csv',
  "ss06hid.csv", method="curl")

# Download the code book:
# download.file(
#   'https://spark-public.s3.amazonaws.com/dataanalysis/PUMSDataDict06.pdf',
#   "PUMSDataDict06.pdf", method="curl")

# load the data into R
idahoData <- read.csv("ss06hid.csv", header=TRUE)

# are we sure it's just Idaho data?
table(idahoData$ST)
#Check the PDF - what does 16 mean?

#any missing data?
```

```
summary(idahoData$ST)
```

```
# How many housing units [are] worth more than $1,000,000?  
table(idahoData$TYPE,idahoData$VAL)
```

```
#from local files  
idahoData <- read.csv("daquiz2.csv", header=TRUE)
```

Question 2

- Consider the variable FES.
- Which of the "tidy data" principles does this variable violate?

Revision

What are the three characteristics of tidy data?

- "***Tidy data***" by Hadley Wickham (RStudio)
- Submission to Journal of Statistical Software
- (<http://vita.had.co.nz/papers/tidy-data.pdf>)

Three Principles from Hadley Wickham's paper

1. Each variable forms a column,
2. Each observation forms a row,
3. Each table/file stores data about one kind of observation.

```
# let's check it out
unique(idahoData$FES)
```

Options

- (i) Each tidy data table contains information about only one type of observation.
(Not so)
- (ii) Each variable in a tidy data set has been transformed to be interpretable.
(No)
- (iii) Tidy data has no missing values.
- (iv) Tidy data has one variable per column.

Question 3

Download the Excel spreadsheet on Natural Gas Acquisition Program here:

https://d396qusza40orc.cloudfront.net/getdata%2Fdata%2FDATA.gov_NGAP.xlsx

Read rows 18-23 and columns 7-15 into R and assign the result to a variable called: `dat`

What is the value of:

```
sum(dat$Zip*dat$Ext,na.rm=T)
```

(original data source: <http://catalog.data.gov/dataset/natural-gas-acquisition-program>)

- (i) NA
- (ii) 36534720
- (iii) 154339
- (iv) 33544718

```
fileUrl <- "https://d396qusza40orc.cloudfront.net/getdata%2Fdata%2FDATA.gov_NGAP.xlsx"
getwd()
download.file(url=fileUrl, destfile="gov_NGAP.xlsx", mode="w", method="curl")

colIndex <- 7:15
rowIndex <- 18:23

library(xlsx)

dat <- read.xlsx(file="gov_NGAP.xlsx",sheetIndex=1,colIndex=colIndex,startRow=18, endRow=23)
head(dat)
summary(dat)

sum(dat$Zip*dat$Ext,na.rm=T)
```

Question 4

Read the XML data on Baltimore restaurants from here:

<https://d396qusza40orc.cloudfront.net/getdata%2Fdata%2Frestaurants.xml>

How many restaurants have zipcode 21231?

Remark : Use `http` instead of `https` , which caused the message Error: XML content does not seem to be XML:

- (i) 100
- (ii) 127
- (iii) 130
- (iv) 28

<http://www.omegahat.org/RXML/shortIntro.pdf>

```

fileUrl <- "http://d396qusza40orc.cloudfront.net/getdata%2Fdata%2Frestaurants.xml"
doc <- xmlTreeParse(fileUrl, useInternal=TRUE)
doc
rootNode <- xmlRoot(doc)
rootNode

rootNode[[1]]

rootNode[[1]][[1]]

names(rootNode[[1]][[1]])

class(rootNode)
mode(rootNode)

xmlName(rootNode)
names(rootNode)

zipcode <- xpathSApply(rootNode, "//zipcode", xmlValue)
table(zipcode == 21231)

## Also
length(zipcode[zipcode==21231])

## Also
sum(xpathSApply(rootNode, "//zipcode", xmlValue)==21231)

```


Question 5

The American Community Survey distributes downloadable data about United States communities.

Download the 2006 microdata survey about housing for the state of Idaho using `download.file()` from here:

`https://d396qusza40orc.cloudfront.net/getdata%2Fdata%2Fss06pid.csv`

using the `fread()` command load the data into an R object `DT`.

Which of the following is the fastest way to calculate the average value of the variable `pwgtp15` broken down by sex using the `data.table` package?

- (i) `apply(split(DT$pwgtp15, DT$SEX), mean)`
- (ii) `rowMeans(DT)[DT$SEX == 1]; rowMeans(DT)[DT$SEX == 2]`
- (iii) `mean(DT$pwgtp15, by = DT$SEX)`
- (iv) `mean(DT[DT$SEX == 1,]$pwgtp15); mean(DT[DT$SEX == 2,]$pwgtp15)`
- (v) `DT[, mean(pwgtp15, by = SEX)]`
- (vi) `tapply(DT$pwgtp15, DT$SEX, mean)`

```
help(proc.time)
help(system.time)
```

Load in the data

```
fileUrl <- "https://d396qusza40orc.cloudfront.net/getdata%2Fdata%2Fss06pid.csv"
download.file(fileUrl, destfile="./data/microdata3.csv", method="curl")
DT <- fread("./data/microdata3.csv")
file.info("./data/microdata3.csv")$size
```

```

# Option A
st = proc.time()
for (i in 1:100){
  sapply(split(DT$pwgtp15,DT$SEX),mean)
}
print (proc.time() - st)

# Option B
st = proc.time()
for (i in 1:100){
  rowMeans(DT) [DT$SEX==1];rowMeans(DT) [DT$SEX==2]
}
print (proc.time() - st)

# Option C
st = proc.time()
for (i in 1:100){
  mean(DT$pwgtp15,by=DT$SEX)
}
print (proc.time() - st)

# Option D
st = proc.time()
for (i in 1:100){
  tapply(DT$pwgtp15,DT$SEX,mean)
}
print (proc.time() - st)

# Option E
st = proc.time()
for (i in 1:100){
  mean(DT[DT$SEX==1,]$pwgtp15);mean(DT[DT$SEX==2,]$pwgtp15)
}
print (proc.time() - st)

# Option F
st = proc.time()
for (i in 1:100){
  DT[,mean(pwgtp15),by=SEX]
}
print (proc.time() - st)

```

```

system.time(DT[,mean(pwgtp15),by=SEX])
system.time(mean(DT[DT$SEX==1,]$pwgtp15))+system.time(mean(DT[DT$SEX==2,]$pwgtp15))
system.time(sapply(split(DT$pwgtp15,DT$SEX),mean))
system.time(mean(DT$pwgtp15,by=DT$SEX))
system.time(tapply(DT$pwgtp15,DT$SEX,mean))
system.time(rowMeans(DT)[DT$SEX==1])+system.time(rowMeans(DT)[DT$SEX==2])

```

Optional Question related to Question 1 and 2

- Use the data from previous question.
- How many households have 3 bedrooms and 4 total rooms?
- How many households have 2 bedrooms and 5 total rooms?
- How many households have 2 bedrooms and 7 total rooms?

```
#USING TABLE
#Rooms on Rows , Bedrooms on Columns
#dnn adds dimension names

table(idahoData$RMS,idahoData$BDS,dnn=list("RMS","BDS"))
```

Another Way of Doing it

```
# How many households have 3 bedrooms and 4 total rooms?
nrow(idahoData[!is.na(idahoData$BDS) & idahoData$BDS==3 &
               !is.na(idahoData$RMS) & idahoData$RMS==4,])
# How many households have 2 bedrooms and 5 total rooms?
nrow(idahoData[!is.na(idahoData$BDS) & idahoData$BDS==2 &
               !is.na(idahoData$RMS) & idahoData$RMS==5,])
# How many households have 2 bedrooms and 7 total rooms?
nrow(idahoData[!is.na(idahoData$BDS) & idahoData$BDS==2 &
               !is.na(idahoData$RMS) & idahoData$RMS==7,])
```

Optional Question related to Question 1 and 2

- Use the data from previous Questions
- Create a logical vector that identifies the households on greater than 10 acres who sold more than \$10,000 worth of agriculture products.
- Assign that logical vector to the variable 'agricultureLogical'.
- Apply the 'which()' function like this to identify the rows of the data frame where the logical vector is 'TRUE'.

```
# Like this (this wont run yet)
which(agricultureLogical)
```

What are the first 3 values that result?

```
# Showing off a bit
q6cols <- c("ACR", "AGS")
which(names(idahoData) %in% q6cols)

# logical vector
agricultureLogical <- idahoData$ACR==3 & idahoData$AGS==6

# and:
which(agricultureLogical)
```

Optional Question related to Question 1 and 2

- Use the data from previous question.
- Create a logical vector that identifies the households on greater than 10 acres who sold more than \$10,000 worth of agriculture products.
- Assign that logical vector to the variable `agricultureLogical`.
- Apply the `which()` function like this to identify the rows of the data frame where the logical vector is TRUE and assign it to the variable `indexes`.

```
indexes = which(agricultureLogical)
```

If your data frame for the complete data is called `dataFrame` you can create a data frame with only the above subset with the command:

```
subsetDataFrame = dataFrame[indexes,]
```

Note that we are subsetting this way because the NA values in the variables will cause problems if you subset directly with the logical statement.

How many households in the `subsetDataFrame` have a missing value for the mortgage status (MRGX) variable?

```
indexes <- which(agricultureLogical)
subsetIdahoData <- idahoData[indexes,]

# And then:
nrow(subsetIdahoData[is.na(subsetIdahoData$MRGX),])
```

Optional Question Related to Question 5

In addition to the data from Question 3, the American Community Survey also collects data about populations. Using `download.file()`, download the population record data from:

https://dl.dropbox.com/u/7710864/data/csv_hid/ss06pid.csv

or here:

<https://spark-public.s3.amazonaws.com/dataanalysis/ss06pid.csv>

- Load the data into R. Assign the housing data from Question 3 to a data frame `housingData` and the population data from above to a data frame `populationData`.
- Use the merge command to merge these data sets based only on the common identifier "SERIALNO".
- What is the dimension of the resulting data set?

```
download.file(
  'https://spark-public.s3.amazonaws.com/dataanalysis/ss06pid.csv',
  'ss06pid.csv', method='curl')

rm(idahoData)
housingData <- read.csv("ss06hid.csv", header=TRUE)
populationData <- read.csv("ss06pid.csv", header=TRUE)

dim(merge(housingData,
  populationData, by="SERIALNO", all=TRUE))
```