MATH 3070 Lab Project 3

Pranav Rajan

September 13, 2020

Contents

Remember: I expect to see commentary either in the text, in the code with comments created using `#`*, or (preferably) both!* ***Failing to do so may result in lost points!***

Problem 1 (Verzani problem 2.43)

The `time` *variable in the* `nym.2002` *data set (**UsingR**) contains the time to finish the 2002 New York City Marathon for a random sample of the finishers.*

1. *What percent ran the race in under 3 hours?*

```
# Your code here
require("UsingR")
```

```
## Loading required package: UsingR
```

```
## Loading required package: MASS
```

```
## Loading required package: HistData
```

```
## Loading required package: Hmisc
```

```
## Loading required package: lattice
```

```
## Loading required package: survival
```

```
## Loading required package: Formula
```

```
## Loading required package: ggplot2
```

```
##
## Attaching package: 'Hmisc'

## The following objects are masked from 'package:base':
##
##     format.pval, units

##
## Attaching package: 'UsingR'

## The following object is masked from 'package:survival':
##
##     cancer
```

```r
# get summary of the data
#str(nym.2002)

# There are 1000 observations of 5 variables
# the time vector
times <- nym.2002$time
# print the first 6 values of the time vector
#print(head(time))

# time is calculated in minutes
# calculate times less than 3 hours
result <- sum(times < (3 * 60))/length(times) * 100
print(result)
```

```
## [1] 2.6
```

2. *What is the time cutoff for the top 10%? The top 25%?*

```r
# Your code here
quantile(times, c(0.1, 0.25))
```

```
##     10%     25%
## 208.695 233.775
```

3. *What time cuts off the bottom 10%?*

```r
# Your code here
quantile(times, c(0.9))
```

```
##    90%
## 331.75
```

Problem 2 (Verzani problem 4.1)

The data set `UScereal` *(**MASS**) contains data on cereals sold in the United States in 1993. For this data set, answer the following questions:*

1. *How many rows does the data frame have? Columns?*

```
# Your code here
require("UsingR")

# Rows
NROW(UScereal)
```

```
## [1] 65
```

```
# Columns
NCOL((UScereal))
```

```
## [1] 11
```

2. *How many different manufacturers are included?*

```
# Your code here
# find all the unique manufacturers
unique(UScereal$mfr)
```

```
## [1] N K G R P Q
## Levels: G K N P Q R
```

3. *How many vitamin categories are included?*

```
# Your code here
# find all the unique vitamin categories
unique(UScereal$vitamins)
```

```
## [1] enriched 100%     none
## Levels: 100% enriched none
```

4. *How many cereals have a sugar level above 10?*

```
# Your code here
# get the sugar vector for the cereals (contains 65 values)
sugars <- UScereal$sugars
sum(sugars > 10)
```

```
## [1] 39
```

5. *What is the mean calorie value for cereals with more than 5 grams of fat? Less than or equal to 5?*

```r
# Your code here
fat <- UScereal$fat

# fat greater than 5 grams
greaterThan5 <- mean(fat > 5)
print(greaterThan5)
```

```
## [1] 0.03076923
```

```r
# fat less <= 5
lessThanEqual <- mean(fat <= 5)
print(lessThanEqual)
```

```
## [1] 0.9692308
```

6. *What is the mean calorie value for cereals on the middle shelf (2)?*

```r
# Your code here
shelf2Cereals <- subset(UScereal, shelf == 2)
shelf2Calories <- shelf2Cereals$calories
mean(shelf2Calories)
```

```
## [1] 129.8162
```

Problem 3 (Verzani problem 4.2)

R uses lists for many purposes behind the scenes. For example, the output of `lm(mpg ~ wt, data=mtcars)` returns a list. Create this object, then answer the following:

1. *How many components does this list have?*

```r
# Your code here
linearModel <- lm(mpg ~ wt, data=mtcars)
print(length(linearModel))
```

```
## [1] 12
```

2. *What are the names of the components?*

```r
# Your code here
names(linearModel)
```

```
##  [1] "coefficients"  "residuals"     "effects"       "rank"
##  [5] "fitted.values" "assign"        "qr"            "df.residual"
##  [9] "xlevels"       "call"          "terms"         "model"
```

3. *What kind of data is held in the `residuals` variable?*

```r
# Your code here
# get the residual data component
residualsData <- linearModel["residuals"]
# determine the type of the residuals data component using the information from str()
typeof(residualsData$residuals)
```

```
## [1] "double"
```

Problem 4

Create a data frame containing the data in the following table:

First	Last	Age
Marcus	Holstein	23
Samuel	Adams	56
Gus	McPherson	43
Margaret	Olsen	41
Zim	Newbold	95

```r
# Your code here
peopleData <- data.frame("First" = c("Marcus, Samuel, Gus, Margaret, Zim"), "Last" = c("Holstein", "Ada
str(peopleData)
```

```
## 'data.frame':    5 obs. of  3 variables:
##  $ First: chr  "Marcus, Samuel, Gus, Margaret, Zim" "Marcus, Samuel, Gus, Margaret, Zim" "Marcus, Sa
##  $ Last : chr  "Holstein" "Adams" "McPherson" "Olsen" ...
##  $ Age  : num  23 56 43 41 95
```