

# Laboratory Exercise Week 5

Brian Tipton - STAT 380 Section 001

9/20/23

## Directions:

- Write your R code inside the code chunks after each question.
- Write your answer comments after the # sign.
- To generate the word document output, click the button Knit and wait for the word document to appear.
- RStudio will prompt you (only once) to install the knitr package.
- Submit your completed laboratory exercise using Blackboard's Turnitin feature. Your Turnitin upload link is found on your Blackboard Course shell under the Laboratory folder.

---

For this exercise, you will need to use the packages `mosaic` and `dplyr`.

```
# install packages if necessary
if (!require(mosaic)) install.packages(`mosaic`)
if (!require(dplyr)) install.packages(`dplyr`)
# load the package in R
library(mosaic) # load the package mosaic to use its functions
library(dplyr) # load the package dplyr to use data management functions
```

1. For decades it's been suspected that schizophrenia involves anatomical abnormalities in the hippocampus, an area of the brain involved with memory. The following data bearing on this issue are from Suddath et al. (1990) and were used by Ramsey and Schafer (3rd ed., 2013, p. 31. Display 2.2). The researchers obtained MRI measurements of the volume of the left hippocampus from 15 pairs of identical twins discordant for schizophrenia, i.e, one the twin is affected with schizophrenia The data are displayed in the following table.

**Do not delete this code chunk**

```
# another way to load a small data set using `read.table()`
schizophrenia <- read.table(header = T, text="
pair    affected    unaffected
1       1.27       1.94
2       1.63       1.44
3       1.47       1.56
4       1.39       1.58
5       1.93       2.06
6       1.26       1.66
7       1.71       1.75
8       1.67       1.77
9       1.28       1.78
10      1.85       1.92
11      1.02       1.25
```

```

12     1.34     1.93
13     2.02     2.04
14     1.59     1.62
15     1.97     2.08
")
is.data.frame(schizophrenia) # check if object `schizophrenia` is a valid data frame

```

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

```
str(schizophrenia)
```

```

## 'data.frame':   15 obs. of  3 variables:
## $ pair      : int  1 2 3 4 5 6 7 8 9 10 ...
## $ affected   : num  1.27 1.63 1.47 1.39 1.93 1.26 1.71 1.67 1.28 1.85 ...
## $ unaffected: num  1.94 1.44 1.56 1.58 2.06 1.66 1.75 1.77 1.78 1.92 ...

```

```
schizophrenia
```

```

##      pair affected unaffected
## 1      1      1.27      1.94
## 2      2      1.63      1.44
## 3      3      1.47      1.56
## 4      4      1.39      1.58
## 5      5      1.93      2.06
## 6      6      1.26      1.66
## 7      7      1.71      1.75
## 8      8      1.67      1.77
## 9      9      1.28      1.78
## 10     10      1.85      1.92
## 11     11      1.02      1.25
## 12     12      1.34      1.93
## 13     13      2.02      2.04
## 14     14      1.59      1.62
## 15     15      1.97      2.08

```

i. use ``mutate()`` to create a new variable ``diff`` which is the difference of the MRI measurements of each pair.

ii. use the pipe ``%>%`` operator add the new variable ``diff`` as column to ``schizophrenia``.

iii. use ``summarise`` to compute the average difference of the MRI measurements. Use the pipe ``%>%`` operator.

iv. use ``summarise`` to compute the standard deviation of the difference of the MRI measurements. Use the pipe ``%>%`` operator.

v. based on your answers in (iii) and (iv), do you think there is evidence in favor of the initial hypothesis?

### Code chunk

```

# start your code
library(dplyr)
## i. use `mutate()` to create a new variable `diff` which is the difference of the MRI measurements of each pair
mutate(schizophrenia, diff = affected - unaffected)

```

```

##      pair affected unaffected diff
## 1      1      1.27      1.94 -0.67
## 2      2      1.63      1.44  0.19
## 3      3      1.47      1.56 -0.09

```

```
## 4      4      1.39      1.58 -0.19
## 5      5      1.93      2.06 -0.13
## 6      6      1.26      1.66 -0.40
## 7      7      1.71      1.75 -0.04
## 8      8      1.67      1.77 -0.10
## 9      9      1.28      1.78 -0.50
## 10     10     1.85      1.92 -0.07
## 11     11     1.02      1.25 -0.23
## 12     12     1.34      1.93 -0.59
## 13     13     2.02      2.04 -0.02
## 14     14     1.59      1.62 -0.03
## 15     15     1.97      2.08 -0.11
```

*## ii. use the pipe `%>%` operator add the new variable `diff` as column to `schizophrenia`.*

```
schizophrenia = schizophrenia %>% mutate(diff = affected - unaffected)
```

*# iii. use `summarise` to compute the average difference of the MRI measurements. Use the pipe `%>%` op*

```
schizophrenia %>% summarise(avg_diff = mean(diff))
```

```
##      avg_diff
## 1 -0.1986667
```

*# iv. use `summarise` to compute the standard deviation of the difference of the MRI measurements. Use*

```
schizophrenia %>% summarise(std_diff = sd(diff))
```

```
##      std_diff
## 1 0.2382935
```

*# last R code line*

- The Behavioral Risk Factor Surveillance System (BRFSS) is an annual telephone survey run by the Centers of Disease Control in the United States. The BRFSS is designed to identify risk factors in the adult population and report emerging health trends. For example, respondents are asked about their diet and weekly physical activity, their HIV/AIDS status, possible tobacco use, and even their level of healthcare coverage.

We only consider a subset of the original survey that contains 20,000 observations.

**Do not delete this code chunk**

```
cdc <- read.csv("http://www.siue.edu/~jpailde/cdc.csv")
str(cdc)
```

```
## 'data.frame':    20000 obs. of  10 variables:
## $ ID      : int   1 2 3 4 5 6 7 8 9 10 ...
## $ genhlth : chr   "good" "good" "good" "good" ...
## $ exerany : int   0 0 1 1 0 1 1 0 0 1 ...
## $ hlthplan: int   1 1 1 1 1 1 1 1 1 1 ...
## $ smoke100: int   0 1 1 0 0 0 0 0 1 0 ...
## $ height  : int   70 64 60 66 61 64 71 67 65 70 ...
## $ weight  : int  175 125 105 132 150 114 194 170 150 180 ...
## $ wt desire: int  175 115 105 124 130 114 185 160 130 170 ...
## $ age     : int   77 33 49 42 55 55 31 45 27 44 ...
## $ gender  : chr   "m" "f" "f" "f" ...
```

i) How many variables are present in this data set? For each variable, identify its data type (e.g. cat

ii) Use `summarise` to compute the average of the variable `weight`.

- iii) Use ``group_by`` to group the rows by ``exerany`` (exercise any). Repeat part (ii) on this grouped data. Do not print the data frame (too long, 20K rows), just print the average weights.
- iv) Repeat part (iii) but now use the grouping variables ``smoke100`` and ``gender``. Comment on what you observe.
- v) Obtain a random sample of 1000 rows and save this into ``cdc.samp1``.
- vi) Repeat parts (ii) to (v) but only using the subset data ``cdc.samp1``. Use the pipe ``%>%`` operator to chain the commands.
- vii) Comment on what differences you observed (if any) between the results in the original sample and the subset.

#### Code chunk

```
# start your code
# i) How many variables are present in this data set? For each variable, identify its data type (e.g. character, integer, etc.)
# 10 Variables
# Variable: ID Type: integer (Continuous)
# Variable: genhlth Type: character (Categorical)
# Variable: exerany Type: integer (Categorical)
# Variable: hlthplan Type: integer (Categorical)
# Variable: smoke100 Type: integer (Categorical)
# Variable: height Type: integer (Continuous)
# Variable: weight Type: integer (Continuous)
# Variable: wt Desire Type: integer (Continuous)
# Variable: age Type: integer (Continuous)
# Variable: gender Type: character (Categorical)
## ii) Use `summarise` to compute the average of the variable `weight`.
cdc %>%
  summarise(avg_weight = mean(weight, na.rm = TRUE))

##   avg_weight
## 1      169.683

# iii) Use `group_by` to group the rows by `exerany` (exercise any). Repeat part (ii) on this grouped data.
cdc %>%
  group_by(exerany) %>%
  summarise(avg_weight = mean(weight, na.rm = TRUE))

## # A tibble: 2 x 2
##   exerany avg_weight
##   <int>     <dbl>
## 1     0      172.
## 2     1      169.

# iv) Repeat part (iii) but now use the grouping variables `smoke100` and `gender`. Comment on what you observe.
cdc %>%
  group_by(smoke100, gender) %>%
  summarise(avg_weight = mean(weight, na.rm = TRUE))

## `summarise()` has grouped output by 'smoke100'. You can override using the
## `.groups` argument.

## # A tibble: 4 x 3
## # Groups:   smoke100 [2]
##   smoke100 gender avg_weight
##   <int> <chr>     <dbl>
```

```
## 1      0 f      151.
## 2      0 m      189.
## 3      1 f      152.
## 4      1 m      190.
```

```
# v) Obtain a random sample of 1000 rows and save this into `cdc.samp1`.
```

```
cdc.samp1 <- sample_n(cdc, 1000)
```

```
# vi) Repeat parts (ii) to (v) but only using the subset data `cdc.samp1`. Use the pipe `%>%` operator
```

```
cdc.samp1 %>%
  summarise(avg_weight = mean(weight, na.rm = TRUE))
```

```
## avg_weight
```

```
## 1      167.785
```

```
cdc.samp1 %>%
```

```
  group_by(exerany) %>%
```

```
  summarise(avg_weight = mean(weight, na.rm = TRUE))
```

```
## # A tibble: 2 x 2
```

```
##   exerany avg_weight
```

```
##   <int>     <dbl>
```

```
## 1      0      168.
```

```
## 2      1      168.
```

```
# last R code line
```

3. Use `sample()` to generate rolls from biased coin with  $Pr(Head) = 0.6$  .

- i) get a sample of size 10 tosses and tally the results
- ii) get a sample of size 30 tosses and tally the results
- iii) get a sample of size 100 tosses and tally the results
- iv) what do you notice with the proportion of heads in each sample?

### Code chunk

```
# star your code
```

```
# i) get a sample of size 10 tosses and tally the results
```

```
sample_10 <- sample(c("H", "T"), size = 10, replace = TRUE, prob = c(0.6, 0.4))
table(sample_10)
```

```
## sample_10
```

```
## H T
```

```
## 7 3
```

```
# ii) get a sample of size 30 tosses and tally the results
```

```
sample_30 <- sample(c("H", "T"), size = 30, replace = TRUE, prob = c(0.6, 0.4))
table(sample_30)
```

```
## sample_30
```

```
## H T
```

```
## 12 18
```

```
# iii) get a sample of size 100 tosses and tally the results
```

```
sample_100 <- sample(c("H", "T"), size = 100, replace = TRUE, prob = c(0.6, 0.4))
table(sample_100)
```

```
## sample_100
```

```
## H T
## 59 41
# last R code line
```