Data Analysis Assignment 2

COMM 7370 | Spring 2023 | Due: see syllabus

In this assignment, you will learn how to:

- Examine frequency distributions using the `freq()` function.
- Select a subset of data based on criteria using the `filter()` function.
- Recode variables using the `mutate()` and `case_when()` functions.
- Create new variables from existing variables in a data frame.
- Create plots using the `ggplot2` package, which is included in the `tidyverse`.
- Conduct inferential statistical tests (e.g., ANOVA, correlation).

> ❗ Remember to...
>
> 1. Set up your R script as you did in your previous assignment.
>
> 2. Include pseudocode in your R script.

Codebook

Table 1: : Codebook for the HELP dataset.

Variable	Description
`age`	R age at baseline
`anysub`	Use of any substance post-detox (no, yes)
`cesd`	Center for Epidemiologic Studies Depression measure at baseline (higher scores indicate more depressive symptoms)
`d1`	Lifetime number of hospitalizations for medical problems (measured at baseline)
`daysanysub`	Time (in days) to first use of any substance post-detox
`dayslink`	Time (in days) to linkage to primary care
`drugrisk`	Risk Assessment Battery drug risk scale at baseline
`e2b`	Number of times in the past 6 months R entered a detox program (measured at baseline)

Variable	Description
`female`	Biological sex (female coded high)
`sex`	Biological sex (male, female)
`g1b`	Experienced serious thoughts of suicide in last 30 days (measured at baseline)
`homeless`	Housing status (housed, homeless)
`i1`	Average number of drinks consumed per day in past 30 days (measured at baseline)
`i2`	Maximum number of drinks consumed per day in past 30 days (measured at baseline)
`id`	R identifier
`indtot`	Inventory of Drug Use Consequences (InDUC) total score (measured at baseline)
`linkstatus`	Post-detox linkage to primary care (yes coded high)
`link`	Post-detox linkage to primary care (no, yes)
`mcs`	SF-36 Mental Component Score (measured at baseline; lower scores indicate worse status)
`pcs`	SF-36 Physical Component Score (measured at baseline; lower scores indicate worse status)
`pss_fr`	Perceived social support by friends (measured at baseline; higher scores indicate more support)
`racegrp`	Race/Ethnicity (black, hispanic, white, other)
`satreat`	Any BSAS substance abuse treatment at baseline (no, yes)
`sexrisk`	Risk Assessment battery
`substance`	Primary substance of abuse (alcohol, cocaine, heroin)
`treat`	Randomized to HELP clinic (no, yes)

Instructions

1) Load the following packages in R and read in the HELP data using the `read_csv()` function.[1]

   - `tidyverse`
   - `summarytools`
   - `rstatix`

2) Often, the first step in data analysis is to examine variables of interest. To do so, we use frequency distributions. To create a frequency table, we can use the `freq()` function in the `summarytools` package. For example, if I want to examine the frequency distribution of variable, `var1`, in the data frame, `df1`, I would use the following code:

```
df1 %>%
        freq(var1) # Examine freq dist of var1 in df1 data frame
```

[1] If you need a reminder of how to do this, check Data Analysis Assignment 1.

Now, examine the frequency distributions of the variables sex and d1 from the HELP data.

a) How many patients in the study are female?

b) How many patients in the study have never been hospitalized for medical problems?

c) What percentage of patients in the study have been hospitalized fewer than 5 times (i.e., 4 or fewer)?

ℹ XQuartz (for Mac users)

If you are having trouble with the summarytools package, you may see the following error message:

```
Error: package or namespace load failed for ‘summarytools’:
.onLoad failed in loadNamespace() for 'tcltk', details:
call: fun(libname, pkgname)
error: X11 library is missing: install XQuartz from xquartz.org
```

To resolve this error, simply read the warning and follow the instructions (i.e., install XQuartz from the source provided).
If you continue to experience issues with summarytools, try reinstalling XQuartz. If you continue to encounter problems, install summarytools directly from Github in R using the following code:

```
install.packages("devtools")
devtools::install_github("dcomtois/summarytools",
                         ref = "no-x11-check")
```

3) Next, we will learn to subset the data to include only respondents who meet certain criteria. To do so, we use the filter() function. If, for example, I want to examine the ages of respondents in the HELP data who are female and have had fewer than 10 hospitalizations, I would use the following code:

```
hdata %>%
        filter(sex == "female" & d1 < 10) %>%
        freq(age)
```

Your turn: Subset the data to include respondents whose primary substance of abuse is cocaine and who are at least 40 years old. Then, examine a frequency distribution of age among this subset.

a) How many patients are included in this subset?

b) What is the mean age in this subset? **Hint:** Use the descr() function to get descriptive statistics.

4) Often, we want to recategorize continuous variables as categorical ones. We call this process **recoding variables**. For example, I want to categorize respondents by their SF-36 Physical Component Score, pcs. I want to create a new variable, dpcs, with two categories, where respondents

with low pcs are those who scored 40 or less on this measure. To do so, I would use the mutate() and case_when functions:

```
hdata <- hdata %>%
        mutate(dpcs = case_when(pcs <= 40 ~ "low",
                                pcs > 40 ~ "high"))

# To check that I have created the new variable, dpcs, correctly, I use freq()
hdata %>%
        freq(dpcs)
```

Your turn: Recode respondents with depressions scores of 30 or lower into a low category and those who have scores higher than 30 into a high category. In other words, recode cesd into a new variable, dcesd (or call it whatever you would like), with only two categories, low (≤ 30) and high (> 30).

a) How many respondents are in each category?

b) What is the mean age of respondents in each of these categories? **Hint:** You can use the filter() and descr() functions to answer this question.

5) Patients with a mental component score (mcs) less than 20 are thought to be at extreme risk of returning to the detoxification unit within the next 12 months. Make a new variable called ExtremeMCS and code it as 1 if a patient is at risk based on his/her mcs score and 0 otherwise. Then, answer the following questions:

   a) How many patients are at risk of returning to the detoxification unit in the next 12 months?

   b) What percentage of patients are at low risk of returning to the detox unit within the next 12 months?

6) Create two new variables, SuicidalThought and HomelessStatus based on g1b and homeless, respectively. If a patient has not experienced thoughts of suicide in the last 30 days, code them as 0 for SuicidalThought. If a patient is housed, code them as 0 for HomelessStatus. We will use these new variables, along with ExtremeMCS to create a scale of risk factors for each patient.

   Suppose ExtremeMCS, SuicidalThought, and HomelessStatus are considered risk factors. Construct a new variable called RiskTotal that quantifies the number of risk factors for each patient (i.e., make it a sum of these 3 variables).

   a) What percentage of patients in the study have fewer than 3 risk factors?

> Tip
>
> To create a new variable, newVar, that is the sum of existing variables, var1 and var2, use the mutate() function.

```
dataframe <- dataframe %>% # write the new variable back into the dataframe
        rowwise() %>%
        mutate(newVar = var1 + var2) # sum to get new variable

# Note the inclusion of the function rowwise() here with nothing in the ().
# We include this to tell R to add the variables row-by-row.

dataframe %>%
        freq(newVar) # check that newVar is created correctly
```

7) Data visualization is an important part of analysis. To start, we will learn how to create a histogram. We will use the `ggplot2` package, which is included in the `tidyverse`. To get started, **read and follow along** using the HELP data with Chapter 5 of COMM 3710: Getting Started with R.

8) Using `ggplot2`, make a histogram of age using `geom_bar`. Describe the distribution: Is the normally distributed? Is it skewed or symmetric? If skewed, is the skew positive or negative? From the graph, what do you think the mean age of respondents is in this dataset?

9) Determine the mean age of patients in the data using functions you have learned. How closely does it match your estimate from the graph?

10) Now, create a scatter plot with `age` on the x-axis and `i1` on the y-axis. Use `geom_point()` to create scatter plots. Is there a relationship between `age` and `i1`? Is it positive or negative? Is it strong or weak?

11) Test the correlations between `age` and `i1`. Some generic code to help you is included below:

    ```
    dataframe %>%
            cor_test(x, y) # Correlation between two continuous variables
    ```

    Is the relationship between `age` and `i1` significant? What is the Pearson's correlation coefficient (r)?

12) Next, we want to test whether there is a difference in SF-36 Physical Component Scores among males and females in the sample. We need to determine which inferential statistical test to use in this case. Making this decision requires that we know whether our variables of interest are categorical or continuous (Figure 1).

    Since `pcs` is continuous and `sex` is categorical with two groups, we can use an independent samples *t*-test. We should also examine the mean of each sample relative to each other by plotting the data. Generic code to do so is included below. The output from the various statistical tests show in Figure 1 can be found in Table 2.

    Once you have conducted the test, state the value of the test statistic and the *p*-value in your R script as comments. Note that for -*p*-values that are less than 0.001, we typically report them as $p < .001$.

```
dataframe %>%
        t_test(varx, vary) # where varx is categorical and vary is continuous

# To graph means with error bars using ggplot2
dataframe |>
        ggplot(aes(x = varx, y = vary)) +
        stat_summary(fun = mean,
                     geom = "point") +
        stat_summary(fun.data = mean_cl_boot,
                     geom = "errorbar")
```

Table 2: Various test statistic names and symbols.

Statistical Test	Test Statistic Name	Test Statistic Symbol
Chi-squared test	Chi-squared value	χ^2
t-test	t-value	t
ANOVA	F-value	F
Pearson's correlation	Pearson's correlation coefficient	r

Submission

Submit your R script (which should have a `.R` extension) to Canvas. Your R script should:

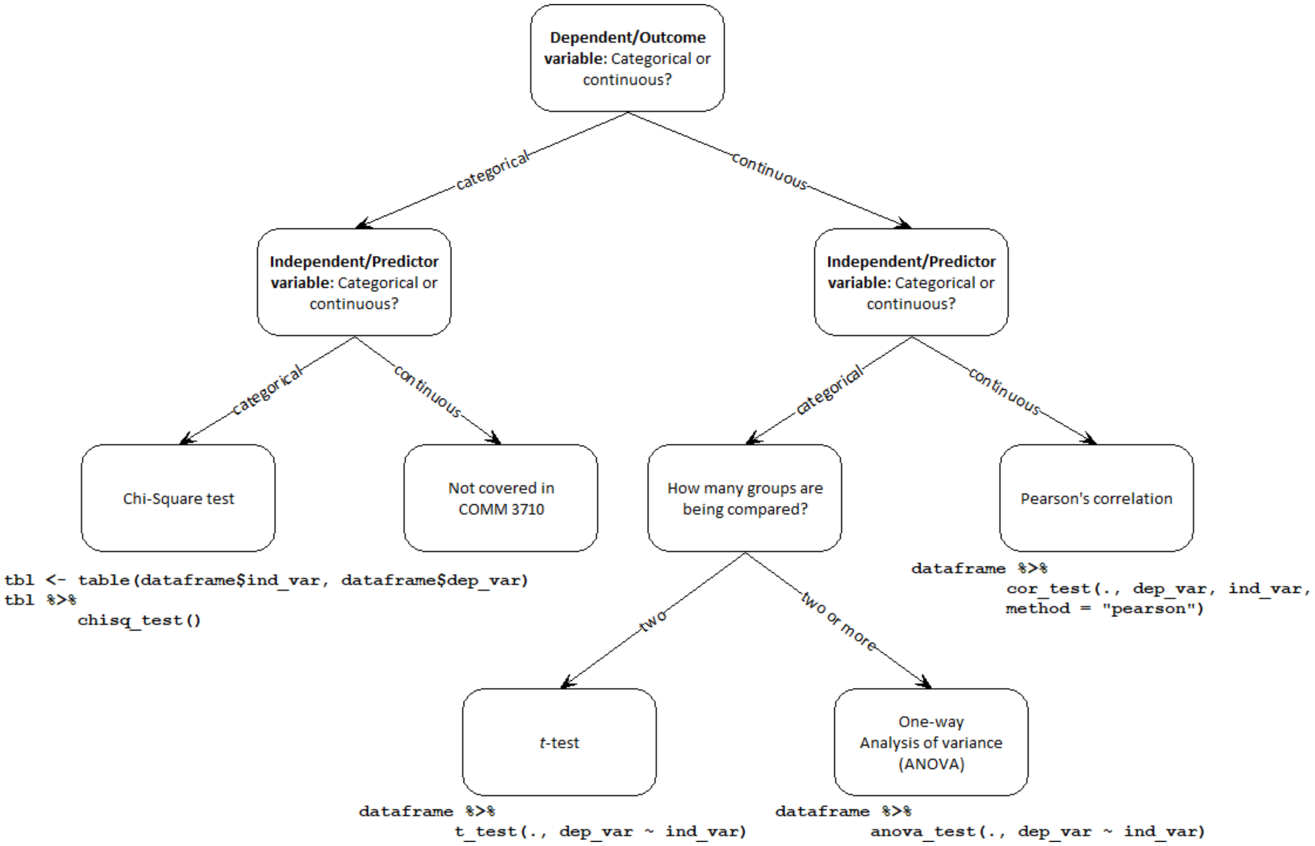
1) Include code to install and load the packages.
2) Contain comments and/or pseudocode.
3) Run in its entirety without errors.

To ensure that your R script runs without errors, you should:

- Save your script.
- Completely shut down RStudio or restart your R session.
- Reopen RStudio and your `.R` script.
- Run the entire script by clicking the "Run" button in the top right of the R script.

❗ Important

These standards apply to all submissions in this course that require R scripts. You should follow these instructions for preparation, naming, and saving of your R script for *all* of your data analysis assignments.


Dependent/Outcome variable: Categorical or continuous?
categorical
continuous
Independent/Predictor variable: Categorical or continuous?
Independent/Predictor variable: Categorical or continuous?
categorical
continuous
categorical
continuous
Chi-Square test
Not covered in COMM 3710
How many groups are being compared?
Pearson's correlation
two
two or more
t-test
One-way Analysis of variance (ANOVA)

```
tbl <- table(dataframe$ind_var, dataframe$dep_var)
tbl %>%
      chisq_test()
```

```
dataframe %>%
        cor_test(., dep_var, ind_var,
        method = "pearson")
```

```
dataframe %>%
        t_test(., dep_var ~ ind_var)
```

```
dataframe %>%
        anova_test(., dep_var ~ ind_var)
```

Figure 1: Decision-making flowchart for selecting a statistical test.