Basic Statistics Using R for Crime Analysis

Jaeyong Choi, Ph.D.

A Member of The Pennsylvania Alliance for Design of Open Textbooks

Cover image Magnifying Glass on White Paper by Nataliya Vaitkevich from Pexels

The contents of this eTextbook were developed under a grant from the Fund for the Improvement of Postsecondary Education, (FIPSE), U.S. Department of Education. However, those contents do not necessarily represent the policy of the Department of Education, and you should not assume endorsement by the Federal Government.

About PA-ADOPT

The Pennsylvania Alliance for Design of Open Textbooks (PA-ADOPT) is made up of four participating institutions from Pennsylvania State System of Higher Education (PASSHE) that are all regional and primarily undergraduate institutions, situated in Southeastern Pennsylvania. The PA-ADOPT project addresses gaps in the open eTextbook marketplace, improve student learning, and mitigate rising student costs. PA-ADOPT was made possible by the US Department of Education Open Textbook Pilot Program.

About OER

Open Educational Resources (OER) are instructional, learning and research materials, digital or non, that open-source and in the public domain or that are licensed so that users have free and perpetual permission to engage in the following activities:

- Retain: the right to make, own, and control copies of the content
- Reuse: the right to use the content in a wide range of ways
- Revise: the right to adapt, adjust, modify, or alter the content itself
- Remix: the right to combine the original or revised content with other open content to create something new
- Redistribute: the right to share copies of the original content, revisions, and remixes with others.

About the Author

Jaeyong Choi

Jaeyong Choi, Ph. D., is an Assistant Professor of Criminal Justice at West Chester University. He has received multiple teaching and research awards, including the 2023 Roslyn Muraskin Emerging Scholar Award from the Northeastern Association of Criminal Justice Sciences, the 2023 CJPR Policy Paper Award from the Sage Publication & Criminal Justice Policy Review, the 2020 Junior Faculty Research Award from the Korean Society of Criminology in America, and the 2018 Teaching Award from the Center for Teaching Excellence at the Indiana University of Pennsylvania. He has published more than 60 peer-reviewed articles in the areas of immigration and criminal justice, cybercrime, criminological theory, and comparative research. His recent work can be found in Criminal Justice & Behavior, Crime & Delinquency, Deviant Behavior, Journal of Criminal Justice, Journal of Interpersonal Violence, Police Quarterly, Policing & Society, and Prison Journal.

Dr. Choi enjoys traveling, taking a walk, and meditating with his wife in his spare time. With a new addition to the family, his days are filled with watching his daughter grow and reach new milestones.

Table of Contents

Preface

I wrote this eTextbook for individuals interested in pursuing careers as crime analysts, especially those in undergraduate programs. Many undergraduate students I've taught understand the significance of data and analysis in effectively operating the criminal justice system. They also recognize that criminal justice agencies value individuals who can proficiently use software programs. However, they often feel discouraged by the challenges they encounter while studying statistics or learning software programs.

Traditional statistics textbooks tend to emphasize the mathematical foundations of statistical techniques, which can be overwhelming and distracting for students. Additionally, students often struggle to see how software programs can be practically applied to analyze crime-related data. Limited access to subscription-based statistical software poses another obstacle. Although students may learn programs like SPSS or Stata while at the university, they often find themselves unable to continue using these programs after graduation, making their acquired skills obsolete.

As an open-source software program, R offers a solution to these challenges. It is freely accessible to anyone, including students, after they graduate. Therefore, I decided to write a freely available book for those interested in becoming crime analysts, focusing on learning statistics without delving too deeply into mathematics. Moreover, this book emphasizes practical applications by utilizing R for data analysis, ensuring students can develop relevant skills beyond the university. I hope that students can easily follow the instructions in this book and replicate the same outcomes using the provided data. This practical experience will demonstrate the value of statistics and R, ideally inspiring students to further their learning in these areas.

Chapter 1. Introduction to Crime Data Analysis, R and RStudio

Data Analysis in the Criminal Justice System

Data analysis in the criminal justice system is often associated with policing. However, intelligence from analysis is a cornerstone across various stages of the criminal justice system, not just within policing. Crime analysts are essential players, providing an objective understanding of the current status and data-driven strategies that inform decision-making and improve effectiveness. Let's examine how crime analysts can contribute to various parts of the criminal justice system.

First, crime analysts are decisive in facilitating investigations and supporting law enforcement agencies in solving crimes. Through meticulous data analysis, they uncover trends and connections that help investigations and lead to the apprehension of offenders. They can also contribute to creating problem-solving strategies to prevent future crimes, allowing agencies to stay ahead of potential threats. For example, analysts can identify hot spot areas and analyze why crime is more concentrated (e.g., motels being used for drug use and prostitution). Once the underlying problem is revealed, crime analysts can discuss with government agencies to take action to address it (e.g., a nuisance abatement lawsuit can be filed by prosecutors against the motel owner whose property has been used for illegal activities repeatedly).

Beyond law enforcement, crime analysts can contribute to broader public safety initiatives and improve overall community well-being. Crime analysts can identify the patterns and provide information to increase the quality of internal operations and resource allocation within agencies. The information from crime analysts enables agencies to address chronic problems more efficiently. Crime analysts can also enhance traffic safety and community quality of life by constructing and implementing a model that allows cars to reach their destination quickly. Furthermore, crime analysts can play significant roles by providing educational materials informing the public of crime-related information and prevention/intervention strategies. They may share the results from their analyses using scientific data. Sharing data-driven information can help communities understand and actively participate in efforts to stop crime and promote the effectiveness of agency-led programs.

Also, crime analysts play vital roles in correctional settings and courts. They examine inmate behavior and trends in correctional settings to identify security

risks, create intervention strategies, and improve overall facility safety. Also, they may assist in evaluating the effectiveness of rehabilitation programs and informing decision-making related to inmate management. Crime analysts can also provide expert testimony and statistical analysis to support court legal proceedings. They may examine crime Information to assess the impact of proposed legislation or policies, evaluate recidivism rates, and inform sentencing decisions. By providing data-driven information, crime analysts add to the fair and effective administration of justice within the legal system.

In short, the work of crime analysts underscores the difficult role of information in promoting safer societies and enhancing the effectiveness of the criminal justice system as a whole, encompassing law enforcement correctional settings and courts.

Importance of Statistics and Statistical Software Programs for Crime Analysts

Statistics and statistical software are essential tools for crime analysts, making them central topics in this book. Proficiency in these areas empowers crime analysts to examine information, identify patterns, effectively make informed decisions, forecast future trends, evaluate interventions, and communicate findings with law enforcement agencies and communities. This book aims to provide comprehensive guidance on these essential skills to improve crime analysts' analytical capabilities and effectiveness in their work.

Crime analysts are crucial in generating essential information for the decision-making of criminal justice agencies. They use statistical software to extract valuable insights from diverse data sources relevant to criminal activities. Crime analysts deliver refined insights to criminal justice agencies by meticulously transforming raw data into meaningful information. For example, the information provided by crime analysts can significantly impact police operations. Once absorbed and understood, such information evolves into actionable knowledge crucial for shaping police actions and strategies, forming the foundation for effective law enforcement initiatives.

The transition from raw data to actionable knowledge involves two interconnected processes. First, data undergoes thorough analysis to extract patterns, trends, and relevant insights. Applying statistical principles and software programs facilitates this transformation from data to structured information. Subsequently, this information is effectively communicated, elevating it to actionable knowledge. Without a solid understanding of statistics

and proficiency in statistical software, analysts will encounter substantial obstacles in extracting actionable knowledge from raw data.

Crime analysts can use crime data that are collected from various sources. Many criminal justice agencies collect firsthand data and make them available to the public. For example, the Bureau of Justice Statistics collects the Police-Public Contact Survey, which includes data regarding US residents who had contact with the police and the nature of this interaction. This dataset is available to the public. If a crime analyst is interested in the circumstances regarding traffic stops, they can use this dataset to analyze the patterns.

Instead of using existing datasets, crime analysts can gather primary data through firsthand methods. For instance, they may conduct community surveys or organizational surveys for police officers to identify crime-related problems or concerns. Crime analysts can also work within correctional settings and study inmate behaviors. By analyzing infraction reports, crime analysts may identify potential security threats such as gang activity, contraband distribution, or inmate conflicts. They can also examine incident reports to observe trends in inmate misconduct, violence, or disciplinary infractions to help prison staff anticipate and prevent future incidents.

The greater the familiarity crime analysts have with statistics and software programs, the more effectively they can bridge the gap between raw data and informed action. Their skillful use of statistics and statistical software programs ensures that the valuable insights from analysis result in informed decision-making within criminal justice agencies. In this book, I will explore a range of statistical analyses, from chi-square tests to multiple regression analysis. I will also demonstrate how to manage and handle data effectively using the R programming language to perform these analyses.

What Is R and RStudio?

R is a computer software that can be used for crime data analysis. R is popular among people who handle and analyze data for several reasons. First, it is free. Anybody can use this software without having to pay a subscription fee. Second, R allows people to conduct various statistical analyses and compute graphical images, from very simple to advanced statistics. A problem with R is that it is not easy to use, especially for those who have not used it before. Writing codes can be overwhelming, but R's user interface does not help make it easy. RStudio is a software that makes R more user-friendly by providing various functions and support.

Setting Up R and RStudio

Now that I have discussed why data analysis is important in the criminal justice field, let us analyze the data to see the value of analysis.

You will need to install R and RStudio, and you can use R Is for Rams to follow the instructions and complete the installation process successfully.

Calculating Using R

While R is incredibly flexible and supports numerous statistical packages, at its core, it functions as a big, powerful calculator. The basic use of R helps you analyze criminal justice data. I believe most of you have not used R before, but if you follow the instructions below, you will not have major trouble producing the same outcome as mine. Since R is a big calculator, we will do some calculations.

Step 1. Launch RStudio

Open the RStudio application on your computer.

Step 2. Choose Script or Console

You can work with R in RStudio using either the script editor or the console. The console lets you execute commands interactively, while the script editor lets you write and save scripts for more complex tasks. Today, we will use the console.

Step 3. Type R Code

In the R console of RStudio, you can directly type R code.

```
#calculate multiplications
5*4*3*2*1
## [1] 120
#you can also do the same thing using the following syntax
factorial(5)
## [1] 120
```

Hashtag # for Annotation

Have you noticed I added a hashtag—"#"—before certain descriptions? The hashtag signals to R that the text following it on the same line is a comment and should not be processed or saved as part of the code. Comments are not essential for code execution but are valuable for explaining and recalling the

purpose of the code. Annotating code is considered a best practice in programming.

A vector is used to store collected elements of the same data type in R. These elements include numbers and characters. It may be easier to see examples to understand the concept of vector.

```
#calculate multiplications
5*4*3*2*1
## [1] 120
#you can also do the same thing using the following syntax
factorial(5)
## [1] 120
#you can create numeric vectors
a1 <- 1
a2 <- 2
#you can create character vectors
b1 <- "criminal"
b2 <- "justice"
```

Did you notice that "<-" was used here? In R, the <- symbol is used as an assignment operator to assign values to variables. It is used to create or update vectors by assigning a value or expression to a vector name. This operator is often referred to as the "left arrow" operator. Here, we created two vectors, a1 and a2, and assigned 1 and 2, respectively. As you can see above, when you create character vectors, you need to enclose the text or characters within double quotation marks. If you want to check if vector values are properly assigned, you can type a vector name.

```
#you can double check the vector values
a1
## [1] 1
a2
## [1] 2
b1
## [1] "criminal"
b2
## [1] "justice"
```

Let us continue to use R as a calculator. We can calculate using the vectors we created earlier. Using vectors makes our analysis much easier and clearer. A vector can contain multiple elements, and we can write simple code to perform more complex calculations.

```
#you can add numeric vectors
a1+a2
## [1] 3
#you can multiply numeric vectors
a1*a2
## [1] 2
#you can subtract vectors
a1-a2
## [1] -1
#you can divide vectors
a1/a2
## [1] 0.5
```

Please note that you cannot calculate using character vectors. For example, b1+b2 would not work in R. You can combine multiple values of the same data type into a single vector using 'c()' function.

```
#create a numeric vector
x <- c(1,2,3,4,5)
#you can check the outcome of this function
x
[1] 1 2 3 4 5
#create a character vector
y <- c("Life", "is", "good")
```

You can use a function to vectors in R. A function refers to a block of code that allows us to perform different types of tasks. Let me give you an example demonstrating how we can use R to simplify a task.

```
# There are many ways to calculate the arithmetic mean of a numeric
vector. You can type.
(1+2+3+4+5)/5
## [1] 3
# But, an easier way to accomplish the same task is to apply a
function to the vector we created earlier.
mean(x)
## [1] 3
```

The mean() function calculates the arithmetic mean or average of a numeric vector or a group of numeric values. You can compute the arithmetic mean by dividing the sum of all values by the total number of values.

R Packages (How To Install and Load Packages)

The basic R functions that come with the software are highly versatile but have limitations. Additional functions are available in packages developed by researchers and contributors worldwide to extend R's capabilities, which are then integrated into the R open-source platform. Throughout this textbook, we will leverage many of these packages. One widely used package we will incorporate is the tidyverse[JC1] package. The tidyverse encompasses various R packages tailored for data science and statistical analysis – simply it is a package that contains multiple packages. These packages synergistically interact to offer a unified framework for data manipulation, visualization, and analytics. Let's practice installing a package and operating it. Before using a package, you must first install it. You can accomplish the installation by using the R function install.packages(), as illustrated below:

```
Install.packages("tidyverse")
```

After installing the tidyverse package, it needs to be loaded for use. Unlike the installation process, each time you wish to use a package, it must be loaded beforehand.

```
Library(tidyverse)
# Load the tidyverse package
library(tidyverse)
```

To briefly demonstrate how you can use the loaded package, I will use the built-in 'USArrests' dataset. You do not need to download this since the USArrests dataset is actually embedded in the basic R package. The dataset includes information on the number of arrests per 100,000 residents for assault, murder, and rape from the 50 US states in 1973. Additionally, it provides for the percentage of the population residing in urban areas. Using the tidyverse package, I filter the dataset to include only certain states and visualize the relationship between the two variables below. Specifically, I wanted to filter the dataset to include only states with a murder rate above 7 and then visualize the number of assaults per 100,000 annually and the percent of the state population living in urban areas using a scatter plot.

```
# Print the first few rows of the dataset
print(head(USArrests))
# Filter the dataset to include only states with a murder rate above
7
filtered_data <- USArrests %>%
    filter(Murder > 7)
```

```
# Print the filtered dataset
print(filtered_data)
# Visualize the relationship between Assault and UrbanPop
ggplot(filtered_data, aes(x = Assault, y = UrbanPop)) +
    geom_point() +
    labs(title = "Relationship between Assault and UrbanPop",
            x = "Assault", y = "UrbanPop") + theme_minimal()
```

Conclusion

This chapter has provided a comprehensive introduction to the role of crime analysts and has emphasized the importance of understanding statistics and R programming. It has outlined the fundamental concepts of R and RStudio, along with basic code examples. In the upcoming chapter, we will delve more deeply into data transformation and visualization techniques.

Chapter 2. Introduction to Data Formations and Graphics

The 2012 General Social Survey

In this chapter, we will practice transforming survey data and making graphs using the transformed data. The survey data we will use is from the 2012 General Social Survey (GSS) conducted by the National Opinion Research Center (NORC). The GSS collects information from a nationally representative sample of the non-institutionalized US population (ages 18 years and older). Specifically, the NORC employs a stratified, multistage area probability sampling method to select households across the USA. The GSS monitors changes and constants in the American population's attitudes, behaviors, and attributes. The collected data covers various topics, including demographics, social attitudes, religion, politics, and the like. The dataset's variables we will focus on are related to perceptions of the police's use of force. Two of the survey questions asked in the 2012 GSS were "Are there any situations you can imagine in which you would approve of a policeman striking an adult male citizen?" and "Would you approve of a police officer striking a citizen who had said vulgar and obscene things to the policeman?" We will focus on these variables to see how people feel about police use of force.

Importing the Data Using the Haven Package

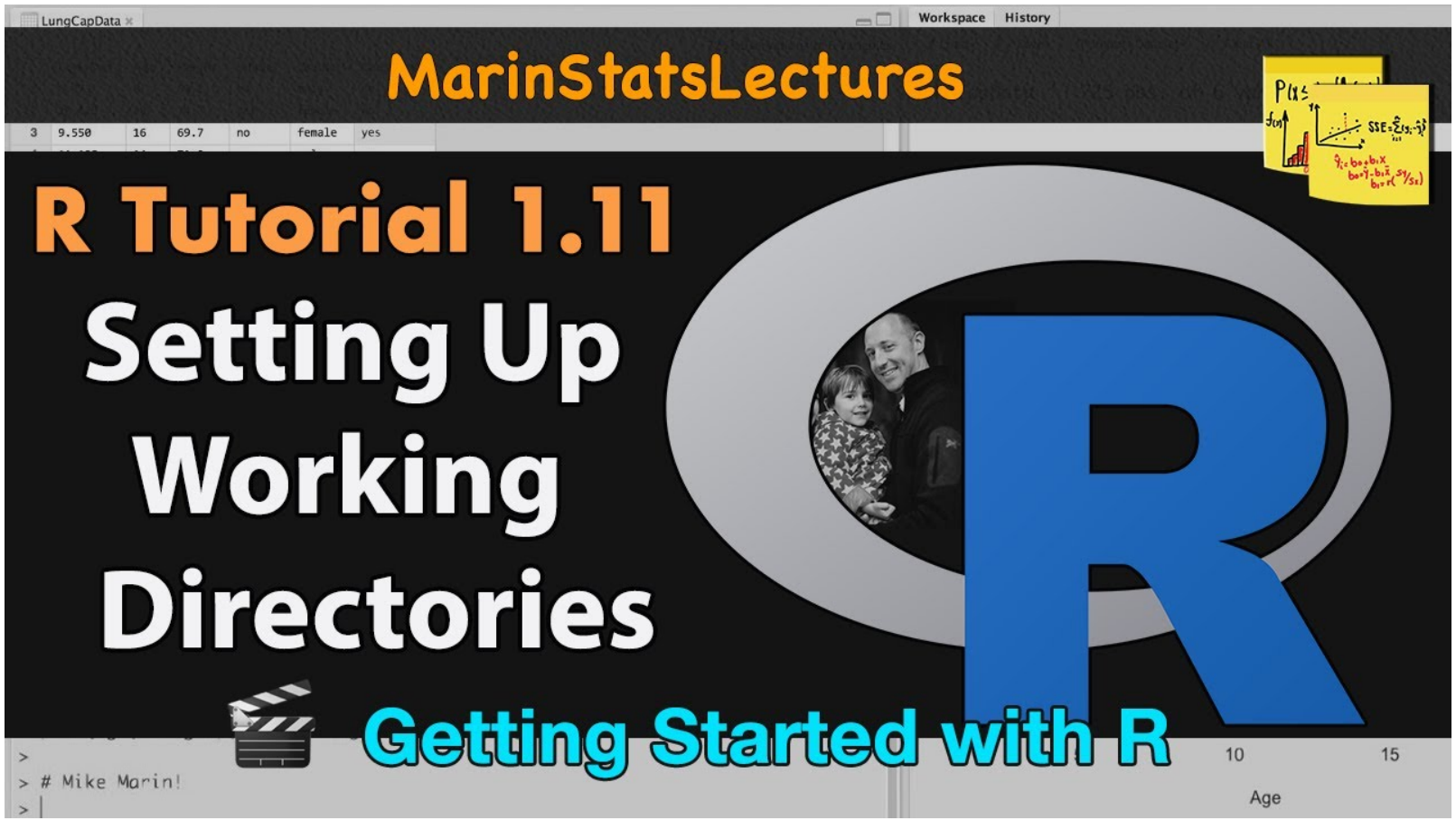
The haven package is a very useful tool within the R environment because it allows users to import data that has different file extensions created in popular statistical software packages such as SAS, SPSS, and Stata. The haven package enables users to import and export datasets effortlessly by bridging the gap between different data formats. As I mentioned in the first chapter, installing and loading the haven package is the first step.

```
install.packages('haven')
library(haven)
```

"library(haven)" loads the haven package in R, providing access to functions like read_sav() and write_sav(). Now, we are ready to read the 2012 GSS data that will be manipulated for the purpose of practice. You will be able to download this file from the shared Google Drive folder containing the 2012 GSS data. The downloaded 2012 GSS data is an SPSS data file, and SPSS data files are typically based on the .sav extension. I will read the data from the file located on the following path: "C:/Users/75JCHOI/OneDrive - West Chester University of PA/

WCU Research/R/data/GSS. 2012.sav". You will have to use your own file location to read the data. Please watch the following video to learn how to set up a working directory in R.

Setting up Working Directories in R



Watch "Setting up Working Directories in R" on YouTube (closed captioned)

I will assign the read data to a new data frame labeled GSS.2012. Please note that we could read this SPSS file because we used the haven package, which allows users to import data from different software programs.

```
GSS.2012 <- read_sav("C:/Users/75JCHOI/OneDrive - West Chester
University of PA/WCU Research/R/data/GSS.2012.sav")
```

View Function

To check what the imported data looks like, use the view() function. The view() function opens a data viewer window in R, allowing you to interactively explore the contents of a data frame.

```
view(GSS.2012)
```

The syntax above displays the contents of the GSS.2012 data frame. If you check the data, you may recognize that each row represents an individual who participated in the survey, and columns represent different variables. For instance, the column labeled as RACE refers to a respondent's race. You will see five variables included in this dataset: RACE, SEX, POLHITOK, POLABUSE, and

AGE. Although many more variables were included in the 2012 GSS, I retained only our variables of interest here to simplify the data management process.

You will see a number below each variable. For instance, respondent's race (RACE) was coded as 1 = White, 2 = Black, and 3 = Other. SEX represents a respondent's sex (1 = Male and 2 = Female). POLHITOK represents a respondent's response to the following question: 'Are there any situations you can imagine in which you would approve of a policeman striking an adult male citizen?' The response options for this item were 1 (yes) and 2 (no). POLABUSE represents a respondent's response to the following question: 'Would you approve of a police officer striking a citizen who had said vulgar and obscene things to the policeman?' The response options for this item were also 1 (yes) and 2 (no). Finally, AGE represents a respondent's age.

Summary Function

There may be too much information to review if you try to read the entire content displayed from the view function. So, reviewing summarized information from our data frame may be better. The summary() function gives a quick rundown of what's inside a data frame, showing both the layout and key statistics for each variable.

```
summary(object = GSS.2012)
```

The results show the minimum, 1st quartile, median, mean, 3rd quartile, and maximum values. What about NA's? NA's indicate the number of missing values in each variable. For example, the summary statistics provided for AGE are as follows:

Minimum (Min.): 18.00

1st Quartile (1st Qu.): 33.00

Median: 47.00

Mean: 48.19

3rd Quartile (3rd Qu.): 61.00

Maximum (Max.): 89.00

Number of missing values (NA's): 5

These statistics give insights into the distribution of ages in the dataset. For instance, the youngest person in the dataset is 18 years old. A quarter of the people are 33 or younger. The median age is 47, meaning half are younger, and

half are older than that. The average age is a bit higher than the median age at 48.19 years, which suggests there might be some older individuals raising the average. Three-quarters of the people are 61 or younger, and the oldest person is 89. Five missing age values that need to be handled based on what kind of analysis or modeling you're doing.

Categorical vs Numerical Variables

What are categorical and numerical variables? In statistical analysis, we use these two types of variables to handle different kinds of data. Categorical variables deal with qualitative data, meaning they sort observations into specific groups or categories. Think of things like gender, race, marital status, education level, and car type. On the other hand, numerical variables handle quantitative data. These are numbers that show amounts or quantities, like age, height, income, or temperature. Examples of numerical variables include age, height, weight, income, and temperature.

Our summary statistics from the summary function suggest that some issues arise with the way our variables are coded. For instance, the RACE variable has been given summary statistics like "Min.", "1st Qu.", "Median", "Mean", "3rd Qu.", "Max.", and "NA's". This is incorrect because RACE is being treated as a numerical variable instead of a categorical one.

There is no average for the categorical variable of race. Calculating the mean of race categories (e.g., "WHITE," "BLACK," "OTHER") would imply a numeric relationship between these categories. However, race categories are qualitative and do not have inherent numeric values. Treating them as numeric would lead to misinterpretation and potentially incorrect conclusions. On the other hand, calculating the average age, which is 48.19 years, makes perfect sense. However, it's illogical to apply the mean to categorical variables like race. Race categories (e.g., White, Black, Asian) don't have a numerical order or value that can be averaged as numerical values do.

We need to recode RACE and SEX, so that they are classified as categorical variables instead of numerical variables. Even POLHITOK and POLABUSE are categorical because the responses "yes" and "no" represent distinct categories rather than numerical values with inherent order or magnitude. While the coded values 1 and 2 for POLHITOK and POLABUSE may appear to be numerical, they are used here to represent categories rather than quantities. So, these variables also need to be transformed properly.

Dplyr Package

We will use the dplyr package to recode some of our variables. The dplyr package provides a set of grammars of data manipulation, offering a consistent set of verbs (functions) that help us solve common data manipulation challenges. We do not need to install this R package. Do you remember that we installed the tidyverse package in the previous chapter? The tidyverse package is a collection of several packages, including dplyr and ggplot2. So dplyr was already installed when we installed the tidyverse package.

One of the dplyr verbs is mutate(), and this function can be used to add new variables (columns) that are functions of existing variables. First, we will recode RACE to a factor, ensuring that it is treated as a categorical variable. However, we want to create a copy of the dataset in case we make mistakes and need to restore the original file. Again, you remember that we can create a new dataset using the left-arrow operator. This new dataset will be named "GSS.2012.cleaned".

```
GSS.2012.cleaned<-GSS.2012 %>%
mutate(RACE = as.factor(x = RACE))
```

The first line, GSS.2012.cleaned <- GSS.2012 %>%, creates a new dataset called "GSS.2012.cleaned". This new dataset will include all our data manipulations after the %>% operator. The %>% operator, pronounced "pipe," lets us string multiple functions together in a sequence. This makes the code easier to read and understand, especially when handling complex data tasks. The second line, mutate(RACE = as.factor(x = RACE)), uses the mutate() function from dplyr to modify the "RACE" variable in the "GSS.2012" dataset. It changes the "RACE" variable into a factor using the as.factor() function, ensuring it is treated as a categorical variable. The x = RACE argument specifies the variable to be transformed. The first RACE in the parenthesis indicates the name of the variable in the new dataset. You can keep it as it is or create a new variable with a new name. This line of code ensures that the "RACE" variable is treated as a categorical factor variable in the dataset.

We can examine whether our data transformation succeeded using the summary function we learned earlier.

```
summary(GSS.2012.cleaned$RACE)
```

We can specify the variable we want to see in the dataset using the $ operator. You will notice that no information is presented regarding minimum, 1st quartile, median, mean, 3rd quartile, and maximum values. Instead, you can see categories 1, 2, and 3:

Category 1: This category has a frequency count of 1477.

Category 2: This category has a frequency count of 301.

Category 3: This category has a frequency count of 196.

However, because we do not know which racial category is associated with each category, we will need to recode the values in the RACE variable using the following syntax:

```
GSS.2012.cleaned<-GSS.2012.cleaned %>%
    mutate(RACE = recode(.x = RACE, "1" = "WHITE")) %>%
    mutate(RACE = recode(.x = RACE, "2" = "BLACK")) %>%
mutate(RACE = recode(.x = RACE, "3" = "OTHER"))
```

“mutate(RACE = recode(.x = RACE, "1" = "WHITE"))” recoded the values in the "RACE" variable by replacing a value of "1" with "WHITE" Similarly, mutate(RACE = recode(.x = RACE, "2" = "BLACK")) recodes the "RACE" variable again. This time, it replaces any value of "2" with “BLACK”. Finally, mutate(RACE = recode(.x = RACE, "3" = "OTHER")) replaced any value of "3" with "OTHER"

We can review our results using the summary function.

```
summary(GSS.2012.cleaned$RACE)
```

The output indicated that there were 1477 whites, 301 blacks, and 196 others.

Even though we broke down this data transformation process into several steps (converting the RACE variable to a factor and then recoding its values to more descriptive labels (“WHITE”), you do not need to create multiple syntaxes because the pipe operator in R can simplify and streamline our codes. For instance, the following syntax can perform a series of data manipulations at the same time.

```
GSS.2012.cleaned<-GSS.2012 %>%
    mutate(RACE = as.factor(x = RACE))%>%
    mutate(RACE = recode(.x = RACE, "1" = "WHITE")) %>%
    mutate(RACE = recode(.x = RACE, "2" = "BLACK")) %>%
mutate(RACE = recode(.x = RACE, "3" = "OTHER"))
summary(GSS.2012.cleaned$RACE)
```

In sum, we can chain multiple actions together, making our code more concise and readable.

Ggplot2 Package

You may wonder if we can visualize the information regarding the RACE variable in the 2012 GSS data in a graph. One of the most popular tools used to create such data visualizations is the ggplot2 package. As with the dplyr, the ggplot2 package is a part of tidyverse. Therefore, you do not need to install or load ggplot2 as long as the tidyverse package is installed and loaded.

The following is a reusable template for making graphs with ggplot2.

```
ggplot(data = <DATA> +
<GEOM_FUNCTION>(mapping = aes(<MAPPINGS>)
```

To create a graph, substitute the section within the angle brackets in the provided code with either a dataset, a geom function, or a set of mappings.

For instance, let's create a bar chart to see how the different racial categories are distributed. Bar charts are handy for showing how often each category appears in a variable. Here's the code using ggplot2 to make the chart:

```
ggplot(GSS.2012.cleaned, aes(x = RACE)) +
    geom_bar()
```

This code initializes a ggplot object with our dataset "GSS.2012.cleaned" and tells ggplot to use the "RACE" variable for the x-axis. Then, it adds bars to the plot with geom_bar(). By default, this function counts how many times each category appears in the "RACE" variable and makes a bar for each one. Now, let's make the chart a bit more informative by adding labels:

```
ggplot(GSS.2012.cleaned, aes(x = RACE)) +
    geom_bar() +
labs(x = "Race", y = "Frequency", title = "Distribution of Race")
```

Here, labs() sets the labels for the x-axis ("Race"), y-axis ("Frequency"), and the plot title ("Distribution of Race"), which gives more context to the chart. With “+” operator, we can add more aesthetic options to the bar graph. You can achieve the same result using the pipe operator:

```
GSS.2012.cleaned %>%
    ggplot(aes(x = RACE)) +
    geom_bar() +
    labs(x = "Race", y = "Frequency", title = "Distribution of Race")
```

Using the pipe operator can give you additional flexibility when more complex tasks need to be added. For example, we can create a new store place using the pipe operator and ggplot.

```
race.bar <- GSS.2012.cleaned %>%
    ggplot(aes(x = RACE, fill = RACE)) +
geom_bar()
race.bar
```

race.bar stores the resulting bar chart plot object, allowing you to further customize or display the plot as needed.

Now, I want to add the colors to the bars in the chart. The bars will be filled with "beige," "black," and "gray" colors, respectively, corresponding to the different levels of the "RACE" variable.

```
race.bar <- GSS.2012.cleaned %>%
    ggplot(aes(x = RACE, fill = RACE)) +
    geom_bar() +
    scale_fill_manual(values = c("beige", "black","gray"),
                      guide = FALSE) +
labs(x = "Race", y = "Frequency", title = "Distribution of Race")
race.bar
```

scale_fill_manual(values = c("beige," "black," "gray"))” sets the fill colors of the bars manually. The values argument specifies a vector of colors to use for filling the bars.

guide = FALSE specifies whether to include a legend or guide for the fill scale. Setting it to FALSE removes the legend, so the colors will be applied directly to the bars without a corresponding legend in the plot.

Finally, you may want to remove the gray background that makes the graph look less professional. You can remove background grids, axis lines, and other non-essential elements, resulting in a clean and simple appearance in ggplot.

```
race.bar <- GSS.2012.cleaned %>%
    ggplot(aes(x = RACE, fill = RACE)) +
    geom_bar() +
    scale_fill_manual(values = c("beige", "black","grey"),
                      guide = FALSE) +
    labs(x = "Race", y = "Frequency", title = "Distribution of Race")
+
theme_minimal()
race.bar
```

theme_minimal() applies the minimalistic theme to the plot created by ggplot2, resulting in a plot with a clean and uncluttered appearance.

In this chapter, I introduced several useful data transformation and visualization packages. Additionally, we reviewed some basic codes and functions that can help you recode the variables and visualize data. In the next chapter, we will learn how to produce descriptive statistics.

Chapter 3. Creating a New Variable and Producing Summary Statistics

Uniform Crime Report

This chapter will teach us how to create a new variable and produce summary statistics. We will use the crime data from the Uniform Crime Report (UCR)—specifically, the 2018 Part 1 crime data from Pennsylvania. More than 18,000 police departments in the US report crime data to the FBI, and the FBI compiles the nationwide data and publishes the UCR.

Crime analysts need to know the differences between Part 1 crime and Part 2 crime. There are two categories of criminal offenses. Part 1 offenses, known as index crimes or serious crimes, are generally felonies that can result in more than a year of incarceration in prisons. Some violent and property crimes that fall under Part 1 offenses include homicide, rape, robbery, aggravated assault, burglary, larceny-theft, motor vehicle theft, and arson. On the other hand, Part 2 offenses are non-index crimes and are considered less serious. These crimes are generally misdemeanors that warrant less than a year of incarceration. Part 2 offenses may include simple assault, vandalism, fraud, drug offenses, disorderly conduct, and other misdemeanors. The biggest limitation of the UCR is that the data only counts crimes reported to the police, so you cannot know how many crimes are underreported. This unreported crime is known as dark or hidden figures of crime, and several data-collecting strategies have been developed to estimate this underreporting. I will introduce these datasets in different chapters. In this chapter, we will learn how to compute descriptive statistics using Part 1 crime data.

Readxl Package

First, you will need to download the file labeled "2018.UCR.PA.xlsx" from the shared Google Drive folder containing the 2018.UCR.PA.xlsx data. The data we will use is from an Excel file. The haven package may be less suitable for importing an Excel file (typically with an xlsx extension) than it was for opening the SPSS file we used previously. So, we should first install and load the readxl package. Then, we will read the file into R.

```
install.packages("readxl")
library(readxl)
X2018.UCR.PA <- read_excel("C:/Users/75JCHOI/OneDrive - West Chester University of PA/WCU Research/R/data/2018.UCR.PA.xlsx")
```

```
View(2018.UCR.PA)
```

This code loads the readxl package and then reads an Excel file at the specified path. The contents of this file are then assigned to an object named X2018.UCR.PA. It’s worth noting that I added an “X” before the object's name “2018.UCR.PA” because R doesn’t allow a number to be the first character in an object name.

When you use the view function to inspect the data, you’ll notice that the dataset's structure differs from the 2012 General Social Survey (GSS) data we worked with in the previous chapter. In the GSS dataset, each row represented a respondent, and each column represented a variable. However, in the UCR dataset, each row represents a city. There are 12 variables in total, including city and population.

The variable “violent crime” in the dataset is computed by aggregating all four different types of Part 1 violent offenses: murder/involuntary manslaughter, rape, robbery, and aggravated assault. Similarly, the variable “property crime” is computed by summing up all four different types of Part 1 property offenses: burglary, larceny-theft, motor vehicle theft, and arson.

Rename Function

Please use the summary function to see if the variables are coded properly.

```
summary(X2018.UCR.PA)
```

The results show that there are some issues with the way variables were labeled. For example, you may notice that the variable name is “Violent\r\ncrime”. This occurs because of how the data is formatted in the Excel file. There was a space between “Violent” and “crime” in the Excel file. We need to rename these variables so that variable names are simplified and easy to type.

Load the tidyverse package and use the rename function from the dplyr package.

```
library(tidyverse)
# Assuming 'my_data' is your data frame or tibble and you want to
change the variable name from 'old_name' to 'new_name'
my_data <- my_data %>%
rename(new_name = old_name)
#Renaming Violent\r\ncrime to violent.crime
X2018.UCR.PA.cleaned <- X2018.UCR.PA %>%
rename(violent.crime = Violent\r\ncrime)
```

Unfortunately, the code above did not work because R was confused about managing backslash characters in the variable name. When a variable name contains special characters such as backslashes, it is important to escape them in R properly.

```
X2018.UCR.PA.cleaned <- X2018.UCR.PA %>%
    rename(violent.crime = 'Violent\r\ncrime')
```

Using a backtick "`" when specifying column names helps to avoid issues with escape characters like "\r\n". But, since there are multiple variables with wrong labels, we can change all of them simultaneously by using the pipe operator as we did in the previous chapter.

```
X2018.UCR.PA.cleaned <- X2018.UCR.PA %>%
    rename(violent.crime = 'Violent\r\ncrime') %>%
    rename(murder.manslaughter = 'Murder
and\r\nnonnegligent\r\nmanslaughter') %>%
    rename(aggravated.assault = 'Aggravated\r\nassault') %>%
    rename(property.crime = 'Property\r\ncrime') %>%
    rename(larceny.theft = 'Larceny-\r\ntheft') %>%
    rename(motor.theft = 'Motor\r\nvehicle\r\ntheft')
summary(X2018.UCR.PA.cleaned)
```

You will see that all variables are properly recoded.

Crime Rates

The term "crime rate" is commonly mentioned in the news. When we hear that crime rates are increasing, it often raises concerns, whereas hearing that crime rates are decreasing tends to make us feel relieved. Yet, what does crime rate mean? The crime rate is determined by dividing the number of Part 1 crimes by the total population and then multiplying the result by 100,000. Therefore, crime rate refers to the number of Part 1 offenses per 100,000 people in a given area. How can we create the crime rate variable for each city in our dataset?

Mutate Function

We can use the mutate function that we learned in the previous chapter. The total number of Part 1 offenses can be computed by adding violent and property crimes to the dataset. This total number of Part 1 offenses will be divided by the population of each city, and the result will be multiplied by 100,000.

```
X2018.UCR.PA.cleaned <- X2018.UCR.PA.cleaned %>%
```

```
    mutate(crime.rate = ((violent.crime + property.crime)/
Population)*100000)
summary(X2018.UCR.PA.cleaned)
```

The mutate() function from the dplyr package was used to add a new variable called “crime.rate” to the “X2018.UCR.PA.cleaned” dataset. Within the mutate() function, the expression (violent.crime + property.crime) / Population) * 100000 calculates the crime rate. It first sums the number of “violent.crime” and “property.crime” incidents, then divides this sum by the “Population” variable, and finally multiplies the result by 100,000 to obtain the crime rate per 100,000 people.

Select Function

To display the names of cities and each city’s crime rate next to each other in R, you can use the select() function from the dplyr package to select only these variables.

```
selected.ucr <- X2018.UCR.PA.cleaned %>%
    select(City, crime.rate)
view(selected.ucr)
```

The code above created a new data frame called “selected.ucr,” containing only the “City” and “crime.rate” columns from the original data frame. Now you can see the information regarding crime rate right next to each city. For example, Abington Township, Montgomery County had 1784.976 Part 1 offenses per 100,000 residents, whereas Adamstown had 915.45504 Part 1 offenses per 100,000 residents.

Arrange Function

Even though it is very useful to view the data frame this way, you may want to see which municipalities have higher crime rates. You can rearrange the data by using the arrange function.

```
arranged.data <- selected.ucr %>%
    arrange(desc(crime.rate))
view(arranged.data)
```

“selected.ucr” was the data frame I used, which only contained the “City” and ” crime.rate” columns. The “arrange(desc(crime.rate))” arranged the data frame in descending order of the “crime.rate” variable. When you view the newly arranged data frame using “view (arranged.data)”, you will see that Wilkes-Barre Township

had the highest crime rate with 17757.009, followed by Frazer Township with 14424.779.

Cut Function

Now, you may want to categorize towns based on population. You can create categories representing different population ranges. Cities may be divided into five different categories: small (population between 0 [inclusive] and 10,000 [exclusive]), medium (population between 10,000 [inclusive] and 50,000 [exclusive]), large (population between 50,000 [inclusive] and 100,000 [exclusive]), very large (population between 100,000 [inclusive] and 500,000 [exclusive]), and metropolitan (population 500,000 [inclusive] and above).

```
# Define the breaks for population categories
breaks <- c(0, 10000, 50000, 100000, 500000, Inf)
# Define the labels for the population categories
labels <- c("Small", "Medium", "Large", "Very Large",
"Metropolitan")
# Create a new variable 'population_category' based on the
population ranges
X2018.UCR.PA.cleaned <- X2018.UCR.PA.cleaned %>%
mutate(population.category = cut(Population, breaks = breaks, labels
= labels, include.lowest = TRUE))
summary(X2018.UCR.PA.cleaned$population.category)
```

I defined the breaks for the population categories. Then, I defined labels for the population categories corresponding to each range. I used the cut() function to categorize the towns into the specified categories by population size. The resulting variable “population.category” will contain the population category for each town. Now, each town in your dataset will be categorized into one of the specified population categories based on its population size. According to the results of the summary statistics, only one metropolitan city and two very large cities exist. The majority of the cities were either small or medium.

Group_By Function

Now, you might want to check the average crime and the total number of crimes or each population category. We are going to use the group_by function from dplyr to perform these tasks.

```
crime.table <- X2018.UCR.PA.cleaned %>%
    group_by(population.category) %>%
```

```
    summarize(avg.crime.rate = mean(crime.rate, na.rm = TRUE),
              total.crimes = sum(crime.rate, na.rm = TRUE))
crime.table
```

The “group_by(population.category)” function groups the data by the “population.category”variable. Then, the “summarize() function” calculates summary statistics within each group. In this case, it computes the average crime rate (“avg.crime.rate”) and the total number of crimes (“total.crimes”) for each population category. The resulting crime.table contains summary statistics for each population category, including the average crime rate and the total number of crimes.

Geom_Histogram()

Finally, we can create a graph to see the distribution of crime rates across Pennsylvania cities. Specifically, we can create a histogram that is useful for displaying the distribution of continuous data by dividing it into bins (i.e., bar charts are useful for visualizing the frequency of each category in categorical variables). We are going to use ggplot2 to accomplish this task.

```
crime.rate.table <- X2018.UCR.PA.cleaned %>%
    ggplot(aes(x = crime.rate, fill = ..count..)) +
    geom_histogram() +
    labs(x = "Crime rates", y = "Frequency", title = "Distribution of
Crime Rates") +
    theme_minimal()
crime.rate.table
```

“X2018.UCR.PA.cleaned” is the data frame containing the variable I wanted to visualize, and “aes(x = variable)” specifies the crime.rate variable that I wanted to plot on the x-axis. “geom_histogram()” was the function used to create the histogram. “fill = ..count.. within the aes()” function assigned a fill color to the bars based on the count of observations in each bin. This results in a gradient color scale where darker colors represent higher frequencies.

In this chapter, we reviewed how to create a variable and produce summary statistics using the concept of crime rates. In the next chapter, we will learn more about central tendency and spread, which are critical statistical measures that visualize data patterns.

Chapter 4. Central Tendency and Variability

Central Tendency

Central tendency refers to statistical measures that summarize the typical or average value within a group of numbers. These measures include the mean, mode, and median.

- Mean: The mean is the most commonly used measure of central tendency. It is calculated by summing up all values in a dataset and dividing the sum by the total number of cases. The mean has the advantageous mathematical property of minimizing variance.
- Median: The median represents the middle score or measurement in ranked scores or measurements. It divides the distribution into two halves. If the number of scores is even, the median is the average of the two middle scores.
- Mode: The mode is the most frequent score in a dataset. It represents the value that occurs most often among the data points (Vogt & Johnson, 2011).

Variability

Variability refers to the extent to which individual scores in a dataset differ. It measures the dispersion or spread of scores around a central tendency, such as the mean. Two commonly used measures of variability are variance and the standard deviation.

Variance: Variance quantifies the spread of scores in a distribution. A larger variance indicates that individual scores are more spread out from the mean, while a smaller variance indicates that scores are closer to the mean. It is calculated as the average of the squared deviations from the mean, representing the average squared distance of each score from the mean.

Standard Deviation: The standard deviation is the square root of the variance. It provides a measure of variability in the original units of measurement. By taking the square root of the variance, we obtain a measure that is more interpretable and easier to understand (Vogt & Johnson, 2011).

We can compute the central tendency and variability measures using R. We will use the gapminder database, a well-known dataset used in data analysis and

visualization. It contains socio-economic indicators for countries around the world over several decades. We will use R data packages to get this data to download the dataset.

Gapminder Data Package

You can easily install and load data packages in R using the library() function when you want to access the data. Let's use the gapminder data package as an example. Created by Jennifer Bryan for educational purposes, the gapminder data package provides a simplified version of the original Gapminder database found at gapminder.org. This package contains a subset of the data, including six variables (country, continent, year, life expectancy at birth, total population, and GDP per capita) for 142 countries, recorded every five years from 1952 to 2007. In this dataset, each row represents a country, whereas each column represents a variable.

To install and load the gapminder package, follow the same steps you would take to install and load any other R package, as discussed in previous chapters.

```
install.packages("gapminder")
library(gapminder)
```

? And Data

There is a simple way to get more information about the gapminder package.

```
?gapminder
```

This command will open the documentation page for the gapminder dataset, providing details about its structure, variables, and usage. You can also find information on how to load the dataset into your R environment and explore its contents.

Now that you have installed and loaded the gapminder package, load the gapminder dataset into the current R session:

```
data("gapminder")
```

By executing data(gapminder), the gapminder dataset will be available for use in your R environment.

Subset Function

Let’s say that we want to know the mean of the total population of 142 countries in 2007. But, the current gapminder dataset has all data for every 5 years from

1952 to 2007. We can subset the gapminder dataset for the year 2007 using the filter function of the tidyverse package.

```
Library(tidyverse)
gapminder.2007 <- gapminder %>%
    filter(year == 2007)
```

The code above subsetted the gapminder dataset to include only observations for the year 2007.

Mean and Median

We will first use this subsetted dataset to compute the mean and median since the steps to get the mode are slightly more complicated.

```
mean(gapminder.2007$pop)
median(gapminder.2007$pop)
```

The above code produces the mean and median: The mean total population in 2007 for the countries in the gapminder dataset is approximately 44,021,220, and the median total population in 2007 for the countries in the gapminder dataset is approximately 10,517,531.

Mode

To compute the mode of the total population in 2007 for the countries in the gapminder dataset, you need to download the DescTools package that allows you to use the mode function.

```
install.packages("DescTools")
library(DescTools)
Mode(gapminder.2007$pop)
```

It is not an error that you do not see a single value for the mode; this happened because there was no most frequent score in the dataset. There was no same number of total population across countries in 2007.

Variance and Standard Deviation

Using the following syntaxes, you can compute the variance and standard deviation of the total population in 2007 for the countries in the gapminder dataset.

```
var(gapminder.2007$pop)
sd(gapminder.2007$pop)
```

Conclusion

In this chapter, we learned how to use R data packages and how to compute central tendency and variability measures. Central tendency and variability are critical statistical concepts that provide valuable insights into the characteristics of a dataset. In the next chapter, we will learn how to check the reliability of a scale.

References

Vogt, W. P., & Johnson, R. B. (2011). Dictionary of statistics & methodology: A nontechnical guide for the social sciences. Sage.

Chapter 5. Reliability of a Scale

Reliability vs Validity

In criminal justice, we often aim to study variables or concepts that are symbolic or abstract in nature. For instance, it's not possible to precisely measure the levels of disadvantage in society as we would measure weight using a scale. Instead, we rely on multiple indicators, such as poverty, low education, or employment status, to capture the degree of disadvantage. Ensuring the accuracy and consistency of these measures is critical, as invalid and unreliable measures can undermine the quality of our analysis.

The reliability of a scale refers to its consistency and freedom from measurement error. Essentially, it reflects how stable or consistent a measure, test, or observation is internally and across repeated uses. When multiple measurements of the same subject under the same conditions produce highly similar results, we consider the instrument reliable. On the other hand, the validity of a scale is about whether an instrument or test accurately measures what it’s supposed to, or how free it is from systematic error. Validity requires reliability, meaning that a measure must be consistent to be valid. However, it’s important to note that reliability alone does not guarantee validity (Vogt & Johnson, 2011).

Let me provide an example to illustrate reliability. Imagine you step on your weight scale, and it shows 160 pounds. You step off and on again, and it reads 150 pounds this time. Repeating the process again, it shows 140 pounds. In this case, your scale wouldn’t be considered reliable because it gives inconsistent readings.

However, your scale would be reliable if you stepped on the scale multiple times and consistently got the same reading, say 170 pounds each time. Even if the reading isn’t accurate (maybe you weigh 160 pounds), the scale is reliable if it consistently gives the same result.

Determining the validity of a scale is often more challenging than estimating its reliability because validity assessment requires more nuanced judgments about the accuracy and appropriateness of those measurements. Reliability can be assessed using statistical methods and quantifiable measures of consistency, but validity is a multifaceted concept involving various dimensions (e.g., content, criterion, and construct validity). In this chapter, we will only focus on the reliability of a scale.

Many social constructs in social science are abstract, and it is very challenging to measure consistently, unlike our physical weight, which is an objective physical property. Thankfully, there is a statistical way to examine the reliability of a scale for crime analysts. In this chapter, we will use the data from the National Crime Victimization Survey (NCVS) to compute the reliability of a scale.

National Crime Victimization Survey

I noted that crimes that go unreported to the police are often called the "dark figure of crime" because they remain undiscovered and unreported, making them hidden from official records. Estimating unreported crimes is crucial for the criminal justice system to function properly.

To address the limitations of official crime measurement methods, the National Crime Victimization Survey (NCVS) was initiated. Conducted by the United States Census Bureau for the Bureau of Justice Statistics, the NCVS surveys over 200,000 individuals from 150,000 randomly selected households. The goal is to generate findings representative of the entire U.S. population. The survey, conducted every six months for three years, asks respondents about their experiences with crime, including whether they have been assaulted and whether they have reported the crime to the police. The survey aims to uncover the true extent of unreported and undiscovered crimes.

Despite its benefits, the NCVS has limitations. One major challenge is that respondents may not always be forthcoming about their victimization experiences. For example, individuals may feel uncomfortable discussing sexual offenses openly. Additionally, the accuracy of respondents' memories may be affected since they are asked about victimization events every six months, leading to potential confusion about timelines. Furthermore, the survey design primarily focuses on street crimes, with limited coverage of other offenses. This narrow focus may result in the underrepresentation of certain types of crimes in the data, such as Part 2 offenses. Nonetheless, the NCVS provides valuable insights into unreported crime.

Test-Retest and Internal Consistency Methods

Two commonly used indicators of a scale's reliability are test-retest reliability and internal consistency. Test-retest reliability involves administering a scale to the same individuals on two separate occasions and calculating the correlation between the scores obtained each time. A high test-retest correlation indicates greater reliability of the scale.

On the other hand, internal consistency assesses the extent to which the items covering the scale all measure the same underlying attribute. There are various methods to measure internal consistency, with Cronbach's alpha coefficient being the most widely used statistic which is also available in R.

Cronbach's Alpha Coefficient

Cronbach's alpha coefficient is one of the most commonly used indicators of internal consistency. Ideally, the Cronbach alpha coefficient of a scale should be above 0.7 (DeVellis, 2012). However, Cronbach's alpha values are influenced by the number of items in the scale. For short scales, such as those with fewer than ten items, it is common to observe relatively low Cronbach values (e.g., below 0.7). In such cases, reporting the scale's mean inter-item correlation may be appropriate. An optimal range for the inter-item correlation is between 0.2 and 0.5 (Pallant, 2016).

Importing the Data in Stata Format

First, you will download the revised data (rNCVS2016.dta) from the shared Google Drive folder containing the rNCVS2016.dta data. I removed numerous cases and variables that are less pertinent to the focus of this chapter. Next, we need to import this data into R. The data is in Stata format, indicated by the .dta extension. You can read the data from the haven package discussed in a previous chapter.

```
library(haven)
rNCVS2016 <- read_dta("C:/Users/75JCHOI/OneDrive - West Chester
University of PA/WCU Research/R/data/rNCVS2016.dta")
View(rNCVS2016)
```

Guardianship

We would like to examine the internal consistency of capable guardianship against identity theft in this dataset. Capable guardianship, the presence of a measure that deters criminal activity, involves a potential target's ability to protect themselves. There are seven items considered to create the capable guardianship scale. Respondents were asked if they had taken seven specific self-protection measures in the past 12 months, including:

- Checking their credit report
- Changing passwords on financial accounts

- Purchasing credit monitoring services or identity theft insurance
- Destroying documents containing personally identifying information
- Reviewing banking or credit card statements for unfamiliar charges
- Using a security software program on their computer to protect against credit card loss or theft
- Purchasing identity theft protection from a company offering protection services

These items were labeled as Guardian1, Guardian2, Guardian3, Guardian4, Guardian5, Guardian6, and Guardian7.The response options for each item ranged from 0 (no) to 1 (yes).

Psych Package

The “psych” package in R is a comprehensive toolbox for psychometric and psychological research. It offers various functions for conducting various types of analyses, including factor analysis, reliability analysis, and item response theory.

```
install.packages("psych")
library(psych)
```

Alpha Function

Now let’s compute Cronbach’s alpha.

```
# Select the columns corresponding to the Guardian variables and
store them in a separate dataframe
guardian_data <- rNCVS2016 %>%
select(starts_with("Guardian")) # Select columns starting with
"Guardian"
# Calculate Cronbach's alpha
alpha_result <- alpha(guardian_data, check.keys = TRUE)
# Print the results
print(alpha_result)
```

I used the tidyverse package to select from the dataset “rNCVS2016” the columns whose internal consistency I wanted to examine. Then, the alpha function from the psych package was used to calculate the Cronbach’s alpha for the selected columns of the dataset. The parameter check.keys = TRUE was included to ensure that the function checks for missing data and calculates alpha accordingly.

The syntax print(alpha_result) printed the results of the Cronbach's alpha analysis to the console. The output includes various statistics related to internal consistency reliability, such as the Cronbach's alpha value, item statistics, and frequencies.

Reporting the Results Regarding the Internal Consistency

Here is how you report the results regarding internal consistency.

While the reliability of the Low Self Control (LSC) scale (Cronbach α = .67) was slightly lower than the widely accepted standard (α = .70, DeVellis, 2012), this alpha is sensitive to the small number of items in the scale. Briggs and Cheek (1986) suggested that as the mean inter-item correlation (.22) for the items is above .2, this scale can be regarded as reliable, indicating that the seven items are reliable sources to capture the levels of guardianship.

Conclusion

In this chapter, we covered the concept of reliability and explored various methods to measure it, including calculating internal consistency using R. The next chapter will delve into cross-tabulation and the chi-square test. These statistical tools are widely utilized for analyzing the relationship between two categorical variables.

References

DeVellis, R. F. (2012). Scale development: Theory and applications (Vol. 26). Sage.

Pallant, J. (2016). *SPSS survival manual* (6th ed.). McGraw Hill.

Vogt, W. P., & Johnson, R. B. (2011). Dictionary of statistics & methodology: A nontechnical guide for the social sciences (4th ed.). Sage.

Chapter 6. Chi-Squared Test

Hypothesis Testing

When evaluating social phenomena, especially regarding crime, people often make judgments. However, such judgments are not always correct. Researchers conducting empirical studies make tentative statements about phenomena and make decisions about the validity of those statements. These tentative statements are called hypotheses, and if one denies a hypothesis that happens to be true, they commit an error in judgment.

When making decisions based on sample statistics about attributes of a population, i.e., unknown facts, one must judge whether something is like that or not. However, such judgments are not always accurate and can sometimes be wrong. In other words, researchers may make errors in judgment.

Specifically, hypothesis testing entails comparing empirically observed sample findings with the theoretically expected outcomes if the null hypothesis were true. The null hypothesis represents the hypothesis that a researcher aims to reject, thereby supporting its alternative. This hypothesis often posits that two or more variables are not related. To compare the null and alternative hypotheses, the researcher calculates the probability of the observed outcome occurring solely due to chance or random error (Vogt & Johnson, 2011).

NHST Steps

This hypothesis testing is also known as null hypothesis significance testing, or NHST. In the context of NHST, it is recommended to follow these five steps:

- Step 1: Formulate the null and alternative hypotheses.
- Step 2: Calculate the test statistic.
- Step 3: Determine the probability (p-value) of obtaining a test statistic at least as extreme as the observed value, assuming no relationship exists.
- Step 4: If the p-value is very small, typically less than 5%, reject the null hypothesis.
- Step 5: If the p-value is not small, typically 5% or greater, retain the null hypothesis.

Chi-Squared Test

The first hypothesis testing covered in this book is the chi-squared test, commonly called the one-sample chi-square. It is frequently employed to compare the proportion of cases from a sample with either hypothesized values or those previously obtained from a comparison population. In the data file, only one categorical variable and a designated proportion against which to evaluate the observed frequency are required. This test may assess whether no difference exists in the proportion within each category (e.g., 50%/50%) or against a specific proportion derived from a previous study.

For instance, 44 male respondents used credit cards, while 62 female respondents used credit cards for online purchases in our example. From this tabulation, we may conclude that more female respondents use credit cards for online purchases than their male counterparts. There is a difference in the usage of credit cards for online purchases between male and female groups. However, we have no evidence to determine whether this difference is statistically significant or by accident. That's why we conduct a hypothesis test to confirm our findings with statistical evidence. Below, we will evaluate the usage of credit cards for online purchases in the past 12 months between men and women using the NCVS data we used in the last chapter.

NHST Steps for Chi-Squared Test

Step 1: Formulate the Null and Alternative Hypotheses.

The first step in conducting the chi-squared test is to write the null and alternative hypotheses.

- H0: The usage of credit cards for online purchases in the past 12 months is the same across men and women.
- HA: The usage of credit cards for online purchases in the past 12 months is not the same across men and women.

Step 2: Calculate the Test Statistic.

The test statistic to use when examining a relationship between two categorical variables is the chi-squared statistics, χ^2.

First, you will download the revised data from the 2016 NCVS from the shared Google Drive. Then, you will need to first load the data using the following syntax.

```
library(haven)
rNCVS2016 <- read_dta("C:/Users/75JCHOI/OneDrive - West Chester
University of PA/WCU Research/R/data/rNCVS2016.dta")
View(rNCVS2016)
```

We will perform a chi-squared test on a contingency table using R. The variable male was coded as 0 for women and 1 for men. The PayCred variable represents whether the respondent used credit cards for online purchases in the past 12 months. Those who used it were coded as 1, and those who did not were coded as 0.

```
chisq.test(x = rNCVS2016$Male,
           y = rNCVS2016$PayCred)
```

The test statistic was $\chi^2 = 0.54045$

Step 3: Determine the Probability (P-Value) of Obtaining a Test Statistic at Least as Extreme as the Observed Value, Assuming no Relationship Exists.

The probability of observing a chi-squared value of 0.54 in our sample—assuming no association between men and women in the population in using credit cards for online purchases in the past 12 months—is calculated to be 0.4622, indicating a p-value greater than 0.05.

Step 4: if the P-Value Is Very Small, Typically Less Than 5%, Reject the Null Hypothesis.

Step 4 is not relevant in this situation.

Step 5: if the P-Value Is Not Small, Typically 5% or Greater, Retain the Null Hypothesis.

The probability that the null hypothesis, stating "The usage of credit cards for online purchases in the past 12 months is the same across men and women," holds true in the population, based on our sample data, is calculated to be 0.4622, indicating a p-value greater than 0.05. This relatively high probability suggests that the null hypothesis is likely true and should not be rejected.

Reporting the Results

So, how do we write up the results based on our test?

We conducted a chi-squared test to examine the null hypothesis, which posited no association between using credit cards for online purchases in the past 12

months and gender. Our analysis failed to reject the null hypothesis, suggesting no statistically significant association between the two variables [χ^2 (1) = 0.54; $p > .05$].

For a demonstration of how a chi-square test is applied and reported, I recommend reviewing the work of Choi and Han (2022). Their study provides a clear example of applying and reporting a chi-square test using the data from NCVS.

Conclusion

In this chapter, we conducted first hypothesis testing using a chi-squared test. In the next chapter, we will use a different statistical test called t-test to compare different groups.

References

Choi, J., & Han, S. (2022). Exploring gender disparity in capable guardianship against identity theft: A focus on internet-based behavior. *International Journal of Criminal Justice, 4*(1), 25-48.

Vogt, W. P., & Johnson, R. B. (2011). Dictionary of statistics & methodology: A nontechnical guide for the social sciences (4th ed.). Sage.

Chapter 7. T-Test

Introduction to T-Test

In the previous chapter, I demonstrated that the chi-squared test is employed when examining whether a significant association exists between two categorical variables. For instance, it can be used to test if there is a correlation between the use of credit cards (yes = 1 or no = 0) for online purchases and gender (male = 1 or female = 0). However, what if one of the variables we want to examine is continuous data, not categorical? For instance, suppose we want to investigate whether there is a difference in the number of items purchased online in the past 12 months between men and women. A chi-squared test will not be the most effective statistical tool here. Instead, the t-test is a statistical significance test frequently utilized to evaluate the difference between two group means, such as the average score on antisocial attitudes among inmates who participated in cognitive behavioral therapy (CBT) and those who did not. For example, even if inmates who completed CBT have lower antisocial attitudes than those who did not, it is a different question if There are two main types of t-tests:

- Independent-samples t-test: This test is employed when comparing the mean scores of two distinct groups of individuals or conditions.
- Paired-samples t-test: This test is used when assessing the mean scores within the same group of individuals on two separate occasions, or when dealing with matched pairs of data.

In this chapter, we will use two fictitious datasets to conduct these two types of t-tests.

Cognitive Behavioral Therapy

Those who study crime and criminals should pay great attention to the issue of rehabilitation. A significant portion of inmates eventually reintegrate into society, where they play crucial roles as members of our communities. A variety of rehabilitation programs have been developed to alter the belief systems and behaviors of offenders. Among the most prevalent programs implemented in correctional settings is CBT. CBT programs operate under the premise that holding antisocial attitudes correlates with increased engagement in antisocial behavior. Specifically, these programs posit that exposure to high-risk situations triggers antisocial thoughts and emotions, thereby heightening the probability of engaging in antisocial behavior (Vaske et al., 2011). CBT programs that target

criminal activity have been used as popular interventions in correctional settings because many evaluation studies have shown that CBT reduces recidivism significantly (Landenberger & Lipsey, 2005; Lipsey et al., 2007; Zara, 2019).

Let's consider a scenario where high-risk inmates were selected at random to participate in a CBT program. This program was conducted in a group setting within the prison, with trained facilitators leading individual sessions aimed at tackling the diverse challenges offenders encounter upon their return to the community. These challenges, spanning from addiction and employment to family obligations and victimization, frequently impede successful reintegration. The program consists of 10 sessions spread over a period of 10 weeks, totaling approximately 20 hours of engagement.

How do we assess the effectiveness of this CBT program? Initially, we must define the criteria by which we will evaluate its success. Those responsible for implementing the program must first clarify its objectives. For instance, the program might aim to modify the attitudes, behaviors, or both, of inmates. For instance, if our focus is on behavioral change, we could monitor participants' recidivism rates, such as rearrest or reconviction. Alternatively, we could concentrate on shifts in participants' attitudes.

Another critical consideration is the research design. We could compare participants in the program to those who did not participate, or within-group changes could be analyzed by comparing participants' attitudes before and after the program.

Independent-Samples T-Test

First, we will examine a scenario where high-risk inmates were randomly chosen to participate in a CBT program. In this context, we will compare the antisocial attitudes of those who were randomly selected for the program with those who were not. For our analysis, let's assume that antisocial attitudes were assessed using a scale developed by Farrington and McGee (2017). This scale comprises a 24-item self-reported instrument with a 4-point response format, including statements like "If someone does the dirty on me, I always try to get my own back" or "I enjoy watching people getting beaten up on TV." The higher antisocial attitudes score reflects higher levels of antisocial attitudes (e.g., aggressiveness).

We will use the fictitious dataset I constructed to contrast inmates who engaged in a CBT program with those who did not. In this hypothetical study, there were 100 high-risk inmates in the prison. Fifty inmates were randomly allocated to participate in a 10-week CBT program, while the other 50 were randomly

assigned not to participate. If the program yielded an impact, we anticipate observing reduced levels of antisocial attitudes among those who participated compared to their counterparts.

Let's first download the data from the shared Google Drive folder containing the CBT_dataset_independent.csv data. Then, load the data for this study using the syntax below. Each row in this data represents each inmate. The group variable is a categorical variable where the category "Participants" refers to those who participated in a CBT program, whereas "Non-Participants" refers to those who did not. The "AntiSocial" variable represents inmates' antisocial attitudes.

```
library(readr)
CBT_dataset_independent <- read_csv("C:/Users/75JCHOI/OneDrive -
West Chester University of PA/WCU Research/R/data/
CBT_dataset_independent.csv")
View(CBT_dataset_independent)
```

EExecuting library(readr) allows me to load the readr package into my R session. The readr package, part of the tidyverse collection, is specifically crafted to simplify importing flat-file data, such as CSVs and text files, into R. You may already be familiar with the next step, but it is worth noting that "read_csv" is a handy function from the readr package tailored for reading CSV files.

NHST Steps for Independent-Samples T-Test

Following the NHST Steps we covered in the previous chapter, we will conduct the independent-samples t-test because we are comparing the mean scores of two distinct groups of individuals (i.e., CBT participants and non-participants).

Step 1: Formulate the Null and Alternative Hypotheses.

- H0: There is no difference in mean antisocial attitudes between CBT participants and non-participants.
- HA: There is a difference in mean antisocial attitudes between CBT participants and non-participants.

Step 2: Calculate the Test Statistic.

The following code facilitates the execution of an independent-samples t-test. In R, a formula typically comprises a single variable on the left, denoted by a ~ (tilde), followed by one or more predictors on the right, which help predict the variable on the left. In statistical tests, the variable on the left of the formula

represents the dependent variable (e.g., antisocial attitudes), while those on the right represent the independent variables (e.g., CBT program participation).

```
twosampt <- t.test(formula = CBT_dataset_independent$AntiSocial ~
                     CBT_dataset_independent$Group)
twosampt
```

The output should generate results from Welch's t-test. Welch's t-test differs slightly from the traditional t-test formula, primarily used when the data deviates from the assumption of equal variances. The t-test () output shows a t-statistic of 4.859. This t-value is well above a t-value of 1.96, which is the historical cut-off point for significance at the 95% confidence level.

Step 3: Determine the Probability (P-Value) of Obtaining a Test Statistic at Least as Extreme as the Observed Value, Assuming no Relationship Exists.

The p-value in this output was shown in scientific notation as 4.485e-06.

You may be confused because you do not know how to convert this number. Here is the way to convert p-value to a regular numeric format.

```
# Your p-value in scientific notation
p_value <- 4.485e-06
# Convert p-value to regular numeric format
formatted_p_value <- format(p_value, scientific = FALSE)
# Print the p-value
formatted_p_value
```

Now you know that the probability of obtaining a t-statistic of 4.859 is 0.000004485 if the null hypothesis is true, which is substantially smaller than .05, the conventionally used standard. R produces a regular numeric p-value, but when the p-value is very low, it may display scientific notation.

Steps 4 & 5: if the P-Value Is Very Small, Typically Less Than 5%, Reject the Null Hypothesis, but if the P-Value Is Not Small, Typically 5% or Greater, Retain the Null Hypothesis.

The t-statistic fell within the rejection region. The probability of this sample coming from a population where the mean antisocial attitudes for both CBT participants and non-participants are equal is exceedingly low. Therefore, it is probable that the sample is from a population where CBT participants and non-participants exhibit statistically significant different mean antisocial attitudes.

Reporting the Results of an Independent-Samples T-Test

The results from our independent-samples t-test can be presented as follows:

An independent-sample t-test was conducted to compare the antisocial attitudes scores for CBT participants and non-participants. There was a significant difference in scores for participants (M = 46.46) and non-participants (M = 55.39; t (97.924) = 4.86, p < .05, two-tailed).

If you want to see how t-test results are reported in an academic peer-reviewed journal, please see Choi et al. (2020). My colleagues and I compared levels of fear of crime between women and men using independent-samples t-tests.

Density Plot

You may wonder if there is a way to visualize the difference in antisocial attitudes between CBT participants and non-participants. Density plot is a useful tool that graphically represents a distribution of scores or values that take the form of a smooth curve. You can create a density plot using ggplot2 and tidyverse to visualize the distribution of antisocial attitudes between two groups (presumably CBT participants and non-participants) from the dataset "CBT_dataset_independent".

```
library(ggplot2)
library(tidyverse)
dens_cbt <- CBT_dataset_independent %>%
    ggplot(aes(x = AntiSocial,
               fill = Group)) +
    geom_density(alpha = .7) +
    theme_minimal() +
    labs(x = "Antisocial Attitudes", y = "Probability Density") +
    scale_fill_manual(values = c('gray', 'black'),
               name = "Group")
dens_cbt
```

- library(ggplot2) loads the ggplot2 package, which is used for creating data visualizations in R.
- library(tidyverse) loads the tidyverse package, a collection of R packages including ggplot2 for data manipulation and visualization.

- dens_cbt <- CBT_dataset_independent %>% creates a ggplot object called dens_cbt. It uses the pipe operator %>% to pass the CBT_dataset_independent data frame into the subsequent ggplot code.
- ggplot(aes(x = AntiSocial, fill = Group)) begins the ggplot object and specifies the aesthetics (aes) mapping. It sets the x-axis to the "AntiSocial" variable and the fill color to the "Group" variable.
- geom_density(alpha = .7) + adds a density layer to the plot, displaying the distribution of antisocial attitudes for each group. The alpha parameter controls the transparency of the density curves.
- theme_minimal() + sets the plot theme to minimal, which removes gridlines and background elements for a cleaner appearance.
- labs(x = "Antisocial Attitudes", y = "Probability Density") + sets the x-axis and y-axis labels.
- scale_fill_manual(values = c('gray', 'black'), name = "Group") manually sets the fill colors for the groups (presumably CBT participants and non-participants) and provides a legend title.
- dens_cbt displays the density plot.

You can also conduct an independent-samples t-test with directionality in R. Specifically, if we were curious about whether CBT participants have lower levels of antisocial attitudes compared to non-participants, you can perform this in R as well. When you specify the alternative hypothesis's direction, it is called a one-tailed test. This is beyond the scope of this book though, so we will move on to the next item in this chapter: paired-samples t-test.

Paired-Samples T-Test

I mentioned that there is another way to evaluate whether a CBT program impacts participants: to compare antisocial attitudes before and after completing a CBT program. However, it should be noted that randomized experiments such as our example above are considered better than pre- and post-test experimental designs. This is because random assignment can ensure that potential confounding variables are evenly distributed between the treatment and control groups. On the other hand, pre- and post-test experimental designs using the same group of people may not control for all these possible confounding variables, such as time-related effects. Additionally, the participants may be systematically different at baseline when designing pre- and post-test experiments.

Let's first download the data from the shared Google Drive folder containing the CBT_dataset_paired.csv data. In this fictitious data, 100 high-risk inmates completed a CBT program, and their survey responses to antisocial attitudes before and after the program were recorded in this dataset. Pre_CBT_Antisocial represents antisocial attitudes before a CBT program, whereas Post_CBT_Antisocial reflects antisocial attitudes after a CBT program. Let's load the data as we did in the earlier section.

```
library(readr)
CBT_dataset_paired <- read_csv("C:/Users/75JCHOI/OneDrive - West
Chester University of PA/WCU Research/R/data/
CBT_dataset_paired.csv")
View(CBT_dataset_paired)
```

NHST Steps for Paired-Samples T-Test

Step 1: Formulate the Null and Alternative Hypotheses.

- H0: There is no difference in antisocial attitudes between the pre-CBT and post-CBT assessments.
- HA: There is a difference in antisocial attitudes between the pre-CBT and post-CBT assessments.

Step 2: Calculate the Test Statistic.

We will use the t.test() function but we will use the paired = TRUE argument to conduct a paired t-test.

```
pairedsampt = t.test(x = CBT_dataset_paired$Pre_CBT_Antisocial,
       y = CBT_dataset_paired$Post_CBT_Antisocial,
       paired = TRUE)
pairedsampt
```

The t.test () output shows a t-statistic of 5.4437.

Step 3: Determine the Probability (P-Value) of Obtaining a Test Statistic at Least as Extreme as the Observed Value, Assuming no Relationship Exists.

The p-value was initially displayed in scientific notation as 3.796e-07. To convert it to a standard numeric format, we will apply the same method used above. The resulting p-value is 0.0000003796, indicating a very low probability of detecting

a mean difference of 5.245796 in antisocial attitudes between the pre-CBT and post-CBT assessments.

Steps 4 & 5: if the P-Value is Very Small, Typically Less Than 5%, Reject the Null Hypothesis, but if the P-Value Is Not Small, Typically 5% or Greater, Retain the Null Hypothesis.

The t-statistic fell within the rejection region. The probability of this sample coming from a population where the mean antisocial attitudes for pre-CBT and post-CBT assessments are equal is exceedingly low. Therefore, it is probable that the sample is from a population where CBT and post-CBT assessments exhibit different mean antisocial attitudes.

Reporting the Results of a Paired-Samples T-Test

Since the results do not show mean scores of antisocial attitudes for pre-CBT and post-CBT assessments, you may want to produce these statistics before reporting the results of a paired-samples t-test.

```
CBT_dataset_paired %>%
    summarize(m_Pre_CBT_Antisocial = mean(x = Pre_CBT_Antisocial),
              m_Post_CBT_Antisocial = mean(x = Post_CBT_Antisocial),
              sd_Pre_Antisocial = sd(x = Pre_CBT_Antisocial),
              sd_Post_Antisocial = sd(x = Post_CBT_Antisocial))
```

Now you have mean scores and standard deviation values of antisocial attitudes for pre-CBT and post-CBT assessments.

The results from our paired-samples t-test can be presented as follows:

A paired-sample t-test was conducted to compare the antisocial attitudes scores for pre-CBT and post-CBT assessments. There was a significant difference in antisocial attitudes for pre-CBT (M = 49.50) and post-CBT assessments (M = 44.25; t (99) = 5.44, p < .05, two-tailed).

If you want to see how t-test results are reported in an academic peer-reviewed journal, please see Choi (2020). In this paper, I used a paired-samples t-test to see changes in perceptions of the police among participants before and after watching videos related to the police.

Conclusion

We have covered two types of t-tests: the independent-samples t-test for comparing the means of two unrelated groups and the paired-samples t-test for

comparing the means of two related groups. In the next chapter, we will delve into analysis of variance (ANOVA), which becomes handy when comparing mean scores across more than two groups. Additionally, you will learn about post hoc tests in ANOVA, which help identify statistically significant differences among multiple means.

References

Choi, J. (2020). Asymmetry in media effects on perceptions of police: An analysis using a within-subjects design experiment. *Police Practice and Research*, 1-17. https://doi.org/10.1080/15614263.2020.1749624

Choi, J., Yim, H., & Lee, D. R. (2020). An examination of the shadow of sexual assault hypothesis among men and women in South Korea. *International Criminal Justice Review, 30*(4), 386-405.

Farrington, D. P., & McGee, T. R. (2017). The integrated cognitive antisocial potential (ICAP) theory: Empirical testing. In A. A. J. Blokland & V. R. Van Der Geest (Eds.), *Routledge international handbook of life- course criminology* (pp. 11–28). Routledge.

Landenberger, N. A., & Lipsey, M. W. (2005). The positive effects of cognitive–behavioral programs for offenders: A meta-analysis of factors associated with effective treatment. *Journal of Experimental Criminology, 1*(4), 451-476.

Lipsey, M. W., Landenberger, N. A., & Wilson, S. J. (2007). Effects of cognitive-behavioral programs for criminal offenders. *Campbell Systematic Reviews, 6*(1), 1-27.

Vaske, J., Galyean, K., & Cullen, F. T. (2011). Toward a biosocial theory of offender rehabiltiation: Why does cognitive-behavioral therapy work? *Journal of Criminal Justice, 39*(1), 90-102.

Zara, G. (2019). Cognitive-behavioral treatment to prevent offending and to rehabilitate offenders. In D. P. Farrington, L. Kazemian, & A. R. Piquero (Eds.), *The Oxford handbook of developmental and life-course criminology* (pp. 694–725). Oxford University Press.

Chapter 8. Analysis of Variance

Introduction to ANOVA

Analysis of variance (ANOVA) involves testing the statistical significance of differences among the mean scores of two or more groups on one or more variables. It serves as an extension of the t-test discussed in our previous class. The ANOVA procedure involves calculating an F ratio, which compares the variance between the groups to the variance within the groups. A large F ratio indicates greater variability between groups, attributed to the independent variable, compared to within each group, which represents random variability.

Media Exposure and Perceptions of the Police

Crime analysts do not always focus on numbers related to crime. Their research subjects can involve how people feel about crime or what people think about the police because residents' perceptions of crime and police can impact the operations of criminal justice professionals. In this chapter, we will use a trimmed version of data collected firsthand, focusing on a study I conducted. In my experiment (Choi, 2018), participants were randomly assigned to watch one of three video clips. The "police misconduct" condition depicted victims or their family members reflecting on police use of force. The "positive police" condition focused on the risks of police work and the sacrifices made by police officers. The "mixed" condition combined elements from both the "police misconduct" and "positive police" conditions to generate mixed messages about the police.

This study aimed to examine whether there are differences in perceptions of the police among groups assigned to watch different police-related videos. There are various methods to measure such perceptions, which have been a subject of debate among policing scholars (Brown & Benedict, 2002; Cao, 2015). In my study, I employed confidence in the police scale, which was consistent with previous research (Gau, 2011; Reisig et al., 2007). This scale includes the following items:

- People’s basic rights are well-protected by police officers in my community.
- Police officers can be trusted to make decisions that are right for my community.
- Most police officers in my community do their jobs well.
- Police officers in my community are generally honest.

Participants' responses to these survey items were measured on visual analog scales, which are psychometric scales allowing respondents to specify their responses visually on a continuous line between two ends.

One-Way Analysis of Variance

In this chapter, we will focus on one-way ANOVA, a statistical method used to analyze differences among the means of three or more independent groups on a single dependent variable. It is important to note that there are other types of ANOVA as well. For instance, two-way ANOVA examines the effects of two categorical independent variables on a single continuous dependent variable.

You will first need to load the data. You will download the file from the shared Google Drive folder containing the media and police.sav data. Since the data is an SPSS file, you will follow the steps that we covered previously. Each row represents each participant. Condition refers to the experimental condition to which each participant was assigned. ConPolT1 represents respondents' confidence in the police before watching the video, and ConPolT2 represents respondents' confidence in the police after watching the video.

```
library(haven)
media_and_police <- read_sav("C:/Users/75JCHOI/OneDrive - West Chester University of PA/WCU Research/R/data/media and police.sav")
View(media_and_police)
```

When importing an SPSS file into an R environment, you might encounter the need for data recoding to effectively manage and clean your data. For instance, you may need to adjust the data type of certain variables for better suitability. In this case, let's focus on recoding the "Condition" variable.

We aim to change the data type of the "Condition" variable to a factor and add category labels to enhance clarity. Specifically, we will label the categories as follows:

- 1 = positive police image
- 2 = negative police image
- 3 = mixed police image (both positive and negative police images)

We learned how to recode a variable and add category labels in a previous chapter.

```
library(tidyverse)
```

```
media_and_police_cleaned <-media_and_police %>%
    mutate(Condition = as.factor(Condition))

class(x = media_and_police_cleaned$Condition)

media_and_police_cleaned <- media_and_police_cleaned %>%
    mutate(Condition = recode(Condition,
                              "1" = "Positive Police Video",
                              "2" = "Negative Police Video",
                              "3" = "Mixed Police Video"))
summary(media_and_police_cleaned)
```

NHST Steps for One-Way ANOVA

We may want to determine whether there is a significant difference in the mean scores for confidence in the police across groups who watched different videos. As mentioned in previous chapters, NHST intends to draw conclusions about the population based on the data from our sample.

Step 1: Formulate the Null and Alternative Hypotheses.

- H0: The mean scores for confidence in the police are equal across groups who watched different videos.
- HA: The mean scores for confidence in the police are not equal across groups who watched different videos.

Step 2: Calculate the Test Statistic.

```
police.by.con <- oneway.test(formula = ConPolT2 ~ Condition,
                             data = media_and_police_cleaned,
                             var.equal = TRUE)
police.by.con
```

The F statistic is 0.49287.

Step 3: Determine the Probability (P-Value) of Obtaining a Test Statistic at Least as Extreme as the Observed Value, Assuming no Relationship Exists.

The p-value obtained is 0.6114, which is considerably larger than the conventional significance level of 0.05. When the p-value is this large, it suggests that the observed result is not statistically significant, indicating no evidence to reject the null hypothesis. It is common to observe such small F-statistics when the null hypothesis is true.

Steps 4 & 5: if the P-Value is Very Small, Typically Less Than 5%, Reject the Null Hypothesis, but if the P-Value Is Not Small, Typically 5% or Greater, Retain the Null Hypothesis.

With a p-value > 0.05, the ANOVA indicates that these groups likely came from a population with similar mean scores for confidence in the police by different experimental conditions.

Reporting the Results From One-Way ANOVA

A one-way ANOVA was conducted to explore the impact of media exposure on confidence in the police. Participants were randomly assigned to watch one of the three conditions (Group 1: Positive Police Video, Group 2: Negative Police Video, and Group 3: Mixed Police Video). There was no statistically significant difference at the $p < .05$ level in confidence in the police $F(2, 296) = 0.49$, $p = 0.61$.

Post-Hoc Test

A post-hoc test is a statistical test conducted after ANOVA to assess the significance of differences between group means when an overall difference is detected. The F ratio from ANOVA indicates the presence of significant differences among the groups, and post-hoc tests aim to identify the specific nature and location of these differences (Vogt & Johnson, 2011).

While the results from our one-way ANOVA suggest that the mean scores for confidence in the police are equal across groups who watched different videos, we will conduct a post-hoc test for demonstration purposes.

There are many types of post-hoc tests, but I will use Tukey's honestly significant difference (HSD) test to identify which groups differ.

```
tukey.police.by.con <-TukeyHSD(x = aov(formula = ConPolT2 ~
Condition, data = media_and_police_cleaned))
```

```
tukey.police.by.con
```

The "diff" column represents the difference between the means in the sample. The "lwr" and "upr" columns denote the lower and upper bounds of a confidence interval around the "diff" value. The "p adj" column displays the adjusted p-value, indicating the statistical significance of the difference after adjusting for multiple comparisons. Not surprisingly, confidence in the police was not significantly different between any of these groups.

Conclusion

In this chapter, we covered ANOVA, which compares the mean scores of more than two groups. In the next chapter, we will learn about correlation.

References

Brown, B., & Benedict, W. R. (2002). Perceptions of the police: Past findings, methodological issues, conceptual issues and policy implications. *Policing: An International Journal of Police Strategies & Management, 25*(3), 543-580.

Cao, L. (2015). Differentiating confidence in the police, trust in the police, and satisfaction with the police. *Policing: An International Journal of Police Strategies & Management, 38*(2), 239-249. https://doi.org/10.1108/pijpsm-12-2014-0127

Choi, J. (2018). *Media exposure, confidence in the police, and police legitimacy* Indiana University of Pennsylvania]. Indiana, PA.

Gau, J. M. (2011). The Convergent and Discriminant Validity of Procedural Justice and Police Legitimacy: An Empirical Test of Core Theoretical Propositions. *Journal of Criminal Justice, 39*(6), 489-498. https://doi.org/10.1016/j.jcrimjus.2011.09.004

Reisig, M. D., Bratton, J., & Gertz, M. G. (2007). The Construct validity and refinement of process-based policing measures. *Criminal Justice and Behavior, 34*(8), 1005-1028. https://doi.org/10.1177/0093854807301275

Vogt, W. P., & Johnson, R. B. (2011). Dictionary of statistics & methodology: A nontechnical guide for the social sciences (4th ed.). Sage.

Chapter 9. Correlation

Introduction to Correlation

Correlation measures the extent to which two or more variables are related to each other and is typically expressed as a correlation coefficient. The correlation test is a basic but commonly used method for examining the relationship between variables by assessing how two variables change together. However, there's a common misconception among students who interpret the warning about correlation: "correlation does not equal causation." Some students mistakenly believe that two correlated variables can never be causally linked. This is an erroneous conclusion. A more accurate understanding of the warning is that correlation does not necessarily imply causation. In other words, while correlation between variables is necessary for causation, it is not always sufficient (Vogt & Johnson, 2011).

Pearson Product-Moment Correlation Coefficient

Correlation measures the strength and direction of the linear relationship between two variables. There are various types of correlations. However, this chapter will focus on a Pearson product-moment correlation coefficient (aka Pearson correlation coefficient, Pearson's r, or Pearson's correlation) designed to evaluate the relationship between continuous variables (or one dichotomous variable and one continuous variable). It is calculated by dividing the covariance (a measure of how two variables covary together) by the product of their standard deviations).

The Pearson correlation coefficient ranges from -1 to +1, indicating the strength and direction of the linear relationship between two variables. A positive correlation (value closer to +1) suggests that as one variable increases, the other also tends to increase. On the other hand, a negative correlation (value closer to -1) indicates that as one variable increases, the other tends to decrease. The absolute value of the correlation coefficient indicates the strength of the relationship between the variables, with larger absolute values representing stronger relationships.

What standards should we use to determine whether a relationship is strong or weak? Many people use Cohen's (1988) guideline to interpret the magnitude of Pearson correlation coefficients. A small correlation, falling within the r = 0.1 to 0.29 range, suggests a relatively weak relationship between the variables under consideration. When the correlation coefficient falls between r = 0.3 and 0.49, it

is classified as a medium correlation, indicating a moderate association between the variables. Conversely, a large correlation, defined as r ≥ 0.5, signifies a robust relationship between the variables.

Finally, a perfect correlation of 1 or -1 indicates that the value of one variable can be precisely determined by knowing the value of the other variable. Conversely, a correlation of 0 indicates no discernible relationship between the two variables.

Computing Correlation Using the USArrests Dataset

We will use the data that we used in the first chapter to estimate correlation. As reviewed in Chapter 1, the built-in 'USArrests' dataset includes information on the number of arrests per 100,000 residents for assault, murder, and rape in each of the 50 states in the United States in 1973 and the percentage of the population residing in urban areas. Each row in the dataset represents a US state. Specifically, we will be evaluating if the number of arrests per 100,000 residents for murder correlates with the number of arrests per 100,000 residents for assault.

Correlation can be obtained through the following functions:

```
# Load the dataset
data("USArrests")
library(tidyverse)
USArrests %>%
summarize(cor.murder.assault = cor(x = Murder, y = Assault, use =
"complete"))
```

The Pearson's product-moment correlation coefficient obtained is 0.8018733. The number of arrests per 100,000 residents for murder was positively correlated with the number of arrests per 100,000 residents for assault. This means that, in the US in 1973, as the number of arrests per 100,000 residents for murder went up, so did the number of arrests per 100,000 residents for assault also increase. According to Cohen's (1988) guideline, this value indicates a very strong correlation between the variables of interest.

However, a crucial aspect is missing: the assessment of statistical significance. Inferential statistics play a vital role here, enabling us to draw conclusions from sample data by inferring insights about a population. Thus, determining whether the observed correlation is statistically significant is essential for meaningful interpretation in inferential statistics. Technically speaking, in our dataset, we do not need to conduct inferential statistics because the data sampled all 50 states instead of only 25 states out of 50 states. But, for demonstration purposes, I will

show how we can conduct a statistical test for correlation coefficients, following the steps for hypothesis testing.

NHST Steps for Pearson's R Correlation Coefficient

Step 1: Formulate the Null and Alternative Hypotheses.

- H0: There is no relationship between the number of arrests per 100,000 residents for murder and the number of arrests per 100,000 residents for assault.
- HA: There is a relationship between the number of arrests per 100,000 residents for murder and the number of arrests per 100,000 residents for assault.

Step 2: Calculate the Test Statistic.

When testing the null hypothesis for the correlation coefficient, we employ a t-statistic to compare the observed correlation coefficient (r) to a hypothesized value of 0. This t-statistic helps us determine whether the observed correlation is statistically significant or if it could have occurred by chance. We will use R codes to perform this task.

```
cor.test(x = USArrests$Murder,
         y = USArrests$Assault)
```

You will see that the correlation coefficient of 0.8018733.

Step 3: Determine the Probability (P-Value) of Obtaining a Test Statistic at Least as Extreme as the Observed Value, Assuming no Relationship Exists.

The p-value, the probability that the t statistic (of 9.2981 in this case) would occur by sampling error, was 2.596e-12, which was essentially much smaller than 0.05.

Steps 4 & 5: if the P-Value is Very Small, Typically Less Than 5%, Reject the Null Hypothesis, but if the P-Value Is Not Small, Typically 5% or Greater, Retain the Null Hypothesis.

The very small p-value (much smaller than 0.05) indicates that a very strong positive relationship between the number of arrests per 100,000 residents for assault and murder is very unlikely if the null hypothesis were true.

Reporting the Results for Pearson's Product-Moment Correlation Coefficient

The number of arrests per 100,000 residents for murder is statistically significant and very strongly positively correlated with the number of arrests per 100,000 residents for assault in 50 US states in 1973 [$r = .80$; $t(48) = 9.30$; $p < .05$]. As the number of arrests per 100,000 residents for murder goes up, the number of arrests per 100,000 residents for assault goes up. While the correlation is .80 in this sample, the correlation is probable between .68 and .88 in the population (95% CI: .68 –.88).

Assumptions That Need To Be Met To Perform Correlation Analysis

It is important to highlight that correlation analysis has specific conditions and assumptions that must be met for accurate interpretation. While these assumptions are crucial for sound statistical analysis, I have not delved deeply into them in this book. This decision stems from the complexity of these assumptions, which could potentially overwhelm those learning the field. Instead, the aim of this book is to provide a broad overview of statistics and analysis, focusing on fundamental concepts rather than intricate technical details.

Ensuring that certain assumptions are satisfied before conducting correlation analysis is crucial. Five key assumptions must be met for reliable results:

- The observations need to be independent of each other. This means that one observation's value should not influence another's value.
- Both variables being analyzed should be continuous. This ensures that the correlation analysis is applicable and meaningful.
- Both variables need to follow a normal distribution. This implies that the data points are evenly distributed around the mean in a bell-shaped curve.
- The relationship between the two variables should be linear. In other words, as one variable increases, the other should either increase or decrease consistently.
- The variance between the two variables should be constant, meaning that the spread of data points around the line of best fit remains consistent throughout the range of values.

It's worth noting that many advanced statistical techniques have been developed precisely because data often fail to meet one or more of these assumptions, highlighting the importance of understanding and addressing these issues in statistical analysis.

Scatter Plot

A scatter plot visually represents the relationship between two variables by displaying individual points on a graph. Each point on the plot corresponds to a unique data point or observation, formed by the intersection of the values of the two variables being studied. By examining the pattern of these points, we can discern the strength and direction of the correlation between the two variables (Vogt & Johnson, 2011). If you want to see how to create a scatter plot, refer to Chapter 1.

Conclusion

In this chapter, we covered correlation and how to compute the Pearson's product-moment correlation coefficient. Chapter 10 will delve into regression analysis, a powerful statistical technique used to understand the relationship between variables. Specifically, we will explore what regression entails and how to calculate regression coefficients, which are essential in quantifying the strength and direction of these relationships.

References

Cohen, J. (1988). Statistical power analysis for the behavioral science. Erlbaum Associates.

Vogt, W. P., & Johnson, R. B. (2011). Dictionary of statistics & methodology: A nontechnical guide for the social sciences (4th ed.). Sage.

Chapter 10. Linear Regression

Introduction to Regression

In a previous chapter, we discussed correlation analysis, which helps us understand the degree of association between two or more variables. Regression analysis is closely linked to correlation analysis, but it offers a more sophisticated way to explore the relationships among variables. Regression is a broad term encompassing a set of statistical methods used for modeling the relationship between a dependent variable and one or more independent variables, such as simple linear regression, multiple linear regression, polynomial regression, logistic regression, and so on. According to Vogt and Johnson (2011), regression analysis serves three primary purposes:

- Predicting the change in a dependent variable for each one-unit increase in an independent variable.
- Predicting the change in a dependent variable associated with a one-unit change in a specific independent variable, while controlling for other independent variables.
- Assessing how much better we can explain or predict a dependent variable by considering all the independent variables together.

Regression analysis is a powerful tool for understanding and quantifying the relationships between variables and making predictions based on those relationships. In this chapter, we will review two forms of linear regression: simple linear regression and multiple linear regression.

Simple Linear Regression Vs. Multiple Linear Regression

Linear regression analysis is commonly used to examine the relationship between one continuous dependent variable and a set of independent variables. Simple linear regression involves examining the linear relationship between the dependent variable and a single independent variable. Conversely, multiple linear regression entails analyzing the impact of multiple independent variables on a dependent variable in the linear relationships.

Ordinary Least Squares (OLS) Model

It is important to note that various types of linear regression models exist, but we will focus on the Ordinary Least Squares (OLS) regression model in this chapter because it is the most widely used. OLS is a statistical estimation technique for determining a regression equation that best represents the relationship between the dependent and independent variables. This method calculates the slope and intercept by minimizing the sum of squared differences between observed and predicted values. Other statistical estimation methods, such as maximum likelihood, are available for establishing a regression model.

Inmate Self-Reported Survey

In the previous chapters, I have discussed various data collection methods (e.g., Uniform Crime Report or National Crime Victimization Survey). Police departments and residents in the community can be great sources of data related to crime, but one source of the data we have not covered yet is inmates. Many inmates are in jails or prisons because they are arrested, prosecuted, and convicted for their accused crimes. If we survey inmates, they may provide useful information that can help us understand crime from offenders' perspectives. This is part of the reason why inmates have been used for various academic articles. For this chapter, I will use the data from an inmate self-reported survey conducted in Korea (Choi & Dulisse, 2021). Specifically, we will first perform a simple linear regression to investigate the relationship between low self-control and risky lifestyles among inmates. Following that, we will conduct a multiple linear regression analysis, considering both low self-control and age as predictors, while evaluating their impact on risky lifestyle, which serves as the dependent variable.

Let's first load the data. You will download the data from the shared Google Drive folder containing the Inmate Survey.sav data. The next steps should be familiar to you at this stage.

```
library(haven)
Inmate_Survey <- read_sav("C:/Users/75JCHOI/OneDrive - West Chester University of PA/WCU Research/R/data/Inmate Survey.sav")
View(Inmate_Survey)
```

A total of 986 inmates from 20 geographically distinct prisons participated in this survey. Risky lifestyles (RL) were assessed using four items that gauge involvement in unstructured criminogenic activities within the prison environment: (a) possession of prohibited items, (b) breaking away from

designated areas, (c) participation in gambling, and (d) involvement in illegal transactions of prohibited products. Participants rated each item on a scale ranging from 0 (never) to 4 (more than 10 times). The scores for these items were summed to obtain a composite measure of risky lifestyles, with higher scores indicating greater involvement. This set of items demonstrates strong internal consistency, prompting students to recall their understanding of reliability testing. Age (AGE) is a continuous variable representing the participants' age. Low self-control (LSC) was assessed based on six items: "I prefer to do things physically rather than verbally," "When encountering difficult or complicated tasks, I usually give up," "I lose my temper easily," "I enjoy doing things that are a little exciting," "I often tease others," and "I prioritize immediate pleasure." A composite measure of low self-control was created by summing the scores on these six items, with higher scores indicating lower levels of self-control.

Assumptions of Linear Regression

In the previous chapter, I emphasized the importance of checking multiple assumptions when conducting statistical analyses, as violating these assumptions can significantly impact linear regression results and lead to biased estimates of coefficients. To ensure the validity of our analysis, we need to consider several additional assumptions. Some of these may already be familiar to you, as they were also necessary for correlation analysis.

- Each observation in our dataset should be independent of the others.
- The outcome variable we are analyzing should be continuous.
- The relationship between the outcome variable and each continuous predictor should be linear.
- The variance of the outcome variable should be constant across all levels of the predictors, with points evenly distributed around the regression line.
- The residuals (the differences between observed and predicted values) should be independent of each other.
- The residuals should follow a normal distribution.
- There should be no strong correlations among the predictor variables, as this can cause numerical instability in the estimation of coefficients.

There are various methods available in R to assess these assumptions. However, discussing them in detail would exceed the scope of our current analysis. For the

purposes of demonstration, we will proceed with the analysis, assuming that these assumptions have been met.

A Scatterplot of Low Self-Control and Risky Lifestyles

We can create a scatterplot of low self-control and risky lifestyles to explore the relationship between these two variables.

```
library(tidyverse)
Inmate_Survey %>%
    ggplot(aes(x = LSC, y = RL, color = "Points")) +
    geom_point(aes(size = "id"), color = "purple", alpha = 0.5) +
    geom_smooth(aes(color = "Linear fit line"), method = "lm", se =
FALSE) +
    theme_minimal() +
    labs(y = "Risky Lifestyles", x = "LSC", color = "", shape = "") +
    scale_size_manual(values = 2, name = "")
```

The graph above should give us a general idea regarding a bivariate relationship between the two variables. The geom_smooth() function with method = "lm" is used to add a linear regression line to the plot, which represents the best-fit straight line through the data points. The aes(color = "Linear fit line") part specifies that the color of this linear fit line will be labeled as "Linear fit line" in the legend, making it distinguishable from other elements in the plot. Setting se = FALSE means that the standard error bands around the linear regression line will not be displayed on the plot. These bands are typically shown by default to indicate the uncertainty or variability of the regression line, but in this case, they are disabled. This line goes up from left to right, showing a positive relationship between low self-control and risky lifestyles. Those with low self-control were more likely to engage in risky lifestyles, which makes sense.

Checking a Correlation Coefficient

You may want to confirm if there is a positive correlation between low self-control and risky lifestyles. You can conduct a correlation analysis such as the one we covered in the previous chapter.

```
library(tidyverse)
Inmate_Survey %>%
    summarize(correlation_lsc_rl = cor(LSC, RL, use =
"pairwise.complete.obs"),
              sample_size = n())
```

There were missing values in our dataset. The pairwise.complete.obs argument allows us to compute the correlation using complete pairs of observations, effectively handling missing values pairwise.

Conducting Simple Linear Regression Analysis

We'll now proceed to calculate the slope and intercept using the Ordinary Least Squares (OLS) method. OLS is employed to minimize the sum of squared differences between the observed and predicted values of the dependent variable by minimizing the overall distance between the data points and the regression line. The y-intercept represents the value of risky lifestyle when low self-control is zero. Meanwhile, the slope denotes the change in risky lifestyle for every one-unit change in low self-control.

```
rl_by_lsc <- lm(formula = RL ~ LSC,
                data = Inmate_Survey, na.action = na.exclude)
summary(object = rl_by_lsc)
```

The linear regression model rl_by_lsc predicts the dependent variable RL (risky lifestyle) based on the independent variable LSC (low self-control) using the lm() function in R. The na.action = na.exclude argument ensures that observations with missing values are included in the analysis rather than being removed.

Based on the results, we can write down the regression equation for our model:

- Risky lifestyles = -0.34 + 0.09 × low self-control

This means that if low self-control increases by one unit in an inmate, risky lifestyles would typically change by 0.09227.

NHST Steps for Simple Linear Regression Model

We may want to make inferences about the population (all inmates within the 20 prisons in South Korea where the current sample was drawn from) using the data we have. That is when we conduct Null Hypothesis Significance Testing (NHST), which was covered previously. Specifically, we may want to assess the statistical significance of the slope in simple linear regression. If the slope (i.e., the unstandardized coefficient of low self-control or the rate of change in risky lifestyle for a one-unit change in low self-control) is not equal to zero, it implies that there is a statistically significant relationship between low self-control and risky lifestyles.

Step 1: Formulate the Null and Alternative Hypotheses.

- H0: The unstandardized coefficient of low self-control is equal to zero.
- HA: The unstandardized coefficient of low self-control is not equal to zero.

Step 2: Calculate the Test Statistic.

The test statistic for the significance of the unstandardized coefficient in OLS regression is the t-statistic we used previously (aka the Wald test).

```
rl_by_lsc <- lm(formula = RL ~ LSC,
                data = Inmate_Survey, na.action = na.exclude)
summary(object = rl_by_lsc)
```

The results indicate that the unstandardized coefficient of low self-control is 0.09, and the t-value is 6.023.

Step 3: Determine the Probability (P-Value) of Obtaining a Test Statistic at Least as Extreme as the Observed Value, Assuming no Relationship Exists.

We can see that the p-value of < 2.45e-09 for the unstandardized coefficient of low self-control is 0.09.

Steps 4 & 5: if the P-Value Is Very Small, Typically Less Than 5%, Reject the Null Hypothesis, but if the P-Value Is Not Small, Typically 5% or Greater, Retain the Null Hypothesis.

The p-value of < 0.05 in our simple linear regression model suggests that there is a very slim probability that the t-statistic for the unstandardized coefficient of low self-control would be as large as observed if the null hypothesis were true. In short, the null hypothesis was rejected in favor of our alternative hypothesis that the unstandardized coefficient of low self-control is not equal to zero.

Reporting the Results From the Simple Linear Regression Model

We found that low self-control reported by inmates is a statistically significant predictor of risky lifestyles (b = 0.09; p < .05) within our sample. Specifically, for every one-unit increase in low self-control among inmates, the predicted increase in risky lifestyle is 0.09 units.

Model Significance for Simple Linear Regression

You might have noticed another p-value toward the bottom of the output, adjacent to the F-statistic for the linear regression. This p-value corresponds to a test statistic that evaluates the improvement of the regression line's fit to the data points compared to the mean value of our dependent variable (risky lifestyles). The F-statistic serves as the test statistic for linear regression, assessing how well the regression line fits compared to the mean value of risky lifestyles. The model fit can be tested by following the NHST steps that we used above.

Step 1: Formulate the Null and Alternative Hypotheses.

- H0: A model including low self-control is not better at explaining risky lifestyles than a baseline model using the mean value of risky lifestyles.
- HA: A model including low self-control is better at explaining risky lifestyles than a baseline model using the mean value of risky lifestyles.

Step 2: Calculate the Test Statistic.

From the provided output above, you can identify the F-value as $F(1, 941) = 36.28$.

Step 3: Determine the Probability (P-Value) of Obtaining a Test Statistic at Least as Extreme as the Observed Value, Assuming no Relationship Exists.

The probability of observing an F-value as large as 36.28, or even larger, if the null hypothesis were true, is very low ($p < 0.05$).

Steps 4 & 5: if the P-Value is Very Small, Typically Less Than 5%, Reject the Null Hypothesis, but if the P-Value Is Not Small, Typically 5% or Greater, Retain the Null Hypothesis.

Given the small p-value, we can reject the null hypothesis in favor of the alternative hypothesis that a model including low self-control is better at explaining risky lifestyles than a baseline model using the mean value of risky lifestyles.

Reporting the Model Significance for the Simple Linear Regression Model

You can add the results regarding the model significance when reporting the results from simple linear regression.

Our model significantly outperformed the baseline model (which used the mean of risky lifestyles) in explaining risky lifestyles ($F(1, 941) = 36.28$; $p < .05$).

Conducting Multiple Linear Regression

Multiple linear regression involves incorporating multiple independent variables to predict the dependent variable. In the context of predicting risky lifestyles among inmates, it's clear that factors beyond just low self-control may play a role. For instance, age could be a significant demographic factor, as younger individuals might be more inclined to engage in risky behaviors compared to older inmates. Therefore, multiple linear regression is better suited for real-life scenarios where multiple factors influence dependent variables. All you need to do is to tweak the R codes that we used to perform a simple linear regression.

```
rl_by_lsc_age<-lm(formula = RL ~ LSC + AGE,
                  data = Inmate_Survey, na.action = na.exclude)
summary(object = rl_by_lsc_age)
```

As you can see, even after including age in our regression model, low self-control remained statistically significant. Low self-control was positively and significantly associated with risky lifestyles ($b = 0.09$; $t = 5.51$; $p < .05$). Age was also a significant predictor of risky lifestyles ($b = -0.01$; $t = -2.30$; $p < .05$). Age was negatively and significantly associated with risky lifestyles.

Model Fit for Linear Regression

In the outputs for both simple linear regression and multiple linear regression, you may have observed multiple R-squared and adjusted R-squared values located just above the F-statistic. These statistics serve to evaluate the overall goodness of fit of the regression model. R-squared (aka the coefficient of determination) is calculated by determining the proportion of variance in the dependent variable that can be explained by the independent variables incorporated in the model. It ranges from 0 to 1, with 0 signifying that the independent variables account for none of the variance in the dependent variable, and 1 indicating that they explain all of the variance.

In our multiple linear regression, for instance, the R-squared value is 0.04252. To determine the percentage of variance explained by the model, multiply this value by 100. Therefore, 4.25% of the variance in risky lifestyles is explained by both low self-control and age. Now, what is adjusted R-squared? As additional variables are added to the model, the R-squared value tends to increase. Adjusted R-squared serves to counteract this tendency by slightly penalizing the R-squared value for each additional variable introduced into the model. This adjustment ensures that the measure appropriately accounts for the complexity of the model and prevents overestimation of its explanatory power.

Conclusion

This chapter introduced the concepts of simple and multiple linear regression, demonstrating how one or more independent variables can be used to predict a single dependent variable. I aimed to give those interested in becoming crime analysts an overview of basic statistics and how R can be employed to conduct statistical analyses. However, it is important to note that this book represents just the starting point of your exploration into statistics and the practical applications of the R programming language. There is much more to discover, and I encourage you to continue your journey toward a deeper comprehension of statistics and the versatile capabilities of R.

References

Choi, J., & Dulisse, B. (2021). Behind closed doors: The role of risky lifestyles and victimization experiences on fear of future victimization among South Korean inmates. *Journal of Interpersonal Violence, 36*(21-22), 10817 –10841. https://doi.org/10.1177/0886260519888186