Basic Data Analysis

Using R

Dr. Austin Brown

Kennesaw State University

Introduction to Data Analysis

- **Objective**: Learn to generate basic descriptive statistics and visualizations.
- **Importance**: Essential for understanding and summarizing data.

Introduction to Data Analysis

- So far, we have learned how to import, inspect, and perform some basic transformations on data.
- After this is complete, we can now focus on analyzing the data to gain insights and answer questions.
- At a foundational stage, this involves generating descriptive statistics and creating visualizations to summarize and present the data.

Introduction to Data Analysis

- Remember in a previous module on understanding column contents, we learned that we generally have two different types of data:
  - Numeric data: Data that represents quantities or numbers.
  - Categorical data: Data that represents categories or groups.
- This difference is not arbitrary: it has implications for the types of analyses we can perform and methods we have available to us.

Introduction to Data Analysis

- Let's once again use the HEART.csv file to demonstrate some basic data analysis techniques.
- As before, go ahead and upload the file to your RStudio Cloud project folder and import the data using the `read.csv` function.
- Now, let's explore some basic analysis of numeric data.

Analyzing Numeric Data

- One of the first steps in analyzing numeric data is to calculate summary statistics.
- In the heart dataset, suppose we want to calculate the sample mean, median, and standard deviation of the `Height` and `Weight` columns.
- While there are several ways to do this, one of the simplest is to use the `get_summary_stats()` function from the `rstatix` package.
- The `get_summary_stats()` function provides a concise summary of the data, including the mean, median, standard deviation, and other key statistics.

Analyzing Numeric Data

```r
## Read in HEART.csv file ##
heart <- read.csv("HEART.csv")
## Install rstatix package ##
install.packages("rstatix")
## Load rstatix package ##
library(rstatix)
## Calculate summary statistics for Age and Weight ##
get_summary_stats(heart, c("Height", "Weight"))
```

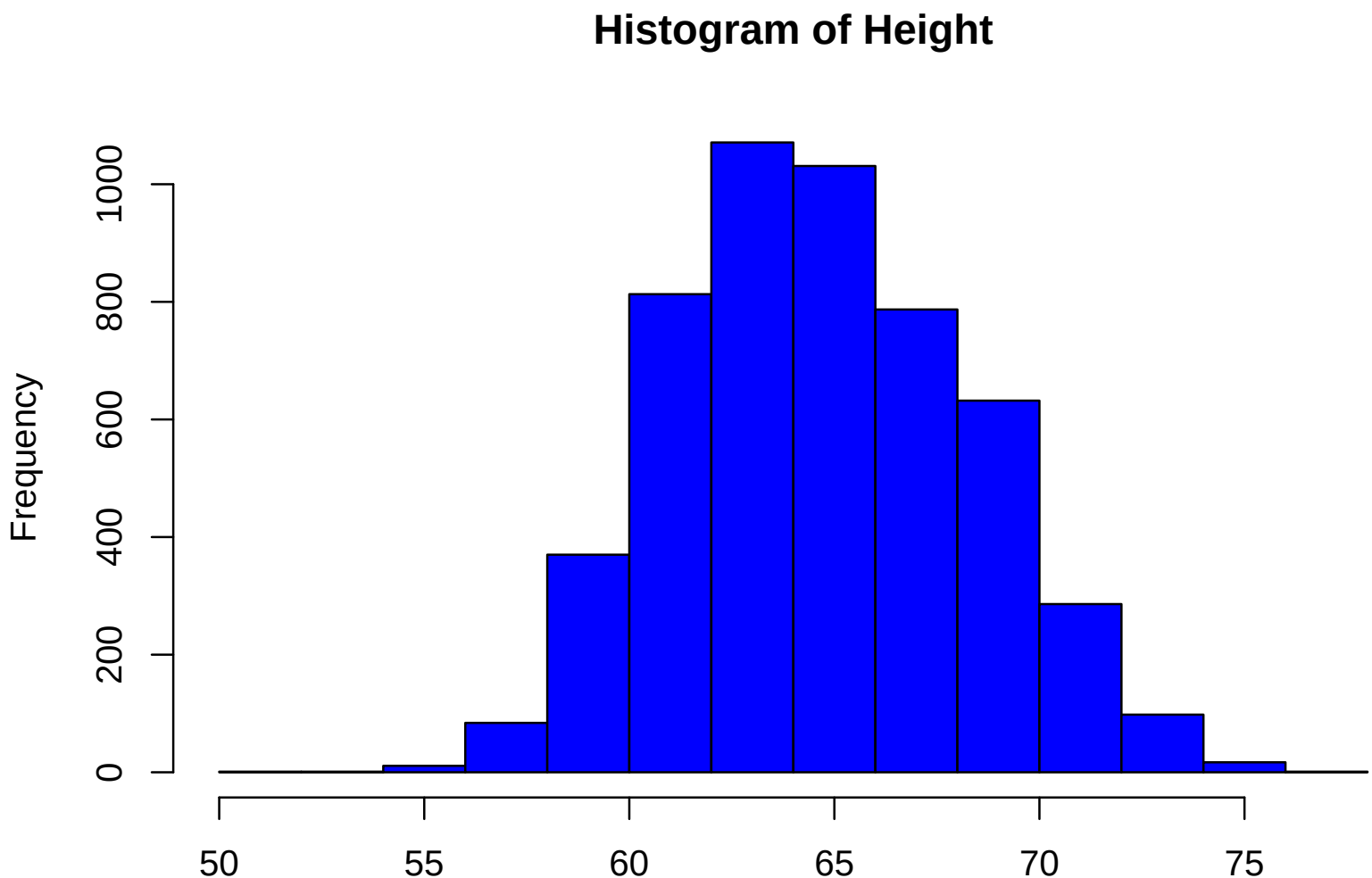
```
# A tibble: 2 x 13
  variable     n   min   max median    q1    q3   iqr   mad  mean    sd    se
  <fct>    <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1 Height    5203  51.5  76.5   64.5  62.2  67.5  5.25  3.71  64.8  3.58 0.05
2 Weight    5203  67   300    150   132   172   40    29.7  153.  28.9  0.401
# i 1 more variable: ci <dbl>
```

Analyzing Numeric Data

- Perhaps the most common method for visualizing a numeric variable is to create a histogram.
- A histogram is a graphical representation of the distribution of a numeric variable.
- The widths of the bars represent the intervals into which the data is grouped, while the heights of the bars represent the frequency of observations in each interval.
- It is a quick, visual tool for understanding common and uncommon values in a dataset.
- In R, we can create a histogram using the `hist()` function.

Analyzing Numeric Data

```
## Create a histogram of the Height variable ##
hist(heart$Height,main="Histogram of Height",
     xlab="Height",col="blue",border="black")
```



Analyzing Numeric Data

- Notice, we select the `Height` column from the `heart` dataset using the `$` operator.
- We then pass this column to the `hist()` function, along with some additional arguments to customize the appearance of the histogram.

Analyzing Categroical Data

- For categorical data, one of the most common ways to summarize the data is to create a frequency table.
- A frequency table is a tabular representation of the number of times each category appears in the data.
- In R, we can create a frequency table using the `table()` function.
- Suppose we want to create a frequency table for the `Weight_Status` column in the `heart` dataset.

Analyzing Categroical Data

```
## Create a frequency table for the Weight_Status variable
table(heart$Weight_Status)
```

```
                Normal  Overweight Underweight
             6    1472        3550         181
```

Analyzing Categroical Data

- The `table()` function takes a single argument, which is the column of data for which we want to create a frequency table.
- In this case, we pass the `Weight` column from the `heart` dataset to the `table()` function.
- The output tells us how many times each category (e.g., `Normal`, `Overweight`, `Underweight`) appears in the data.
  - Note, the blank category is due to missing values in the dataset.
  - This means that `Weight_Status` contains 6 missing values.

Analyzing Categorical Data

- A common way to visualize categorical data is to create a bar plot.
- A bar plot is a graphical representation of the frequency of each category in a dataset, similar to a histogram for numeric data.
- In R, we can create a bar plot using the `barplot()` function.
- Suppose we want to create a bar plot for the `Weight_Status` column in the `heart` dataset.

Analyzing Categorical Data

```
## Create a bar plot of the Weight_Status
## variable ##
barplot(table(heart$Weight_Status),
        main="Bar Plot of Weight Status",
        xlab="Weight Status",ylab="Frequency",col="blue")
```