**Day\_2\_session\_01**

**Module IV- Analysis of Data**

**Part I: Descriptive statistics for Categorical variables**

In this session, we explore some basic commands to calculate descriptive statistics and generate associated graphs.

First, we will check if the required packages for this session are installed, if not we install them, and load them into our working environment.

req <- substitute(require(x, character.only = TRUE))

libs<-c("psych", "tidyverse", "lessR", "patchwork")

sapply(libs, function(x) eval(req) || {install.packages(x); eval(req)})

**Import Data**

job <- read\_csv("JobSatisfaction.csv")

**## Perform Analysis**

req <- substitute(require(x, character.only = TRUE))

job <- read\_csv("JobSatisfaction.csv")

### Nominal, categorical & ordinal data analysis

count(job,Gender)

count(job, JobSat1)

count(job, Gender, Location)

xtabs(~Gender+Location, data=job)

## NA issue

xtabs(~Gender+Location, data=job, addNA = T)

## Frequency and percentage in one table

# Calculate the frequency of each gender

gender\_freq <- table(job$Gender)

# Calculate the percentage using prop.table()

gender\_percentage <- prop.table(gender\_freq) \* 100

# Create a data frame to combine frequency and percentage

gender\_summary <- data.frame(Gender = names(gender\_freq),

Frequency = as.vector(gender\_freq),

Percentage = as.vector(gender\_percentage))

# Display the summary table

print(gender\_summary, digits = 2)

## Data visualization

BarChart(JobSat1,data = job)

?BarChart

BarChart(JobSat1,data = job, by1=Location)

BarChart(JobSat1,data = job, by=Location)

PieChart(Gender,data = job)

job$gender1 <- ifelse(job$Gender == "Man", 1, 2)

job1 <- na.omit(job)

gender\_counts <- table(job1$gender1)

# Create a pie chart

pie\_chart <- pie(gender\_counts,

labels = c("Male", "Female"),

col = c("red", "green"),

main = "Gender Distribution")

# Adding a legend

legend("topright", c("Male", "Female"), fill = c("red", "green"))

# Create a 3D pie chart

pie3D(gender\_counts,

labels = c("Male", "Female"),

explode = 0.1, # Explode the slices for emphasis

main = "Gender Distribution")

**### Side-box plot using simulated data with ggplot2 package**

library(ggplot2)

# Simulated data

set.seed(123) # Setting seed for reproducibility

categories <- rep(c("Category A", "Category B", "Category C"), each = 30)

values <- rnorm(90, mean = c(10, 15, 8), sd = c(2, 3, 1))

# Create a data frame

data <- data.frame(category = categories, value = values)

# Create a side-by-side box plot

ggplot(data, aes(x = category, y = value, fill = category)) +

geom\_boxplot() +

labs(title = "Side-by-Side Box Plot",

x = "Category",

y = "Value") +

theme\_minimal() +

theme(legend.position = "none")

**Part II: Descriptive statistics for Continuous variables**

### Continuous variable

# Install and load necessary packages if not already installed

library(psych)

library(officer)

library(flextable)

## Summary statistics using psych package

describe(job)

## Create Summary stat table and export as word file

# Generate summary statistics using the describe() function

summary\_stats <- describe(job)

# Convert the summary statistics into a data frame

summary\_df <- as.data.frame(summary\_stats)

# Create a flextable from the summary data frame

ft <- flextable(summary\_df)

# Specify the file path for the Word document

output\_file <- "E:/R\_workshop\_SUST/summary\_statistics.docx"

# Save the flextable as a Word document

doc <- read\_docx()

doc <- body\_add\_flextable(doc, value = ft)

print(doc, target = output\_file)

## Visualization (base R function is not recommended)

hist(job$Tenure\_Yrs, col = c("red","green","blue"))

##Violin plot Using lessR package

Plot(Tenure\_Yrs, data = job)

Plot(Tenure\_Yrs, data = job, by1 = Location)

**Part IV: T-Test**

**One Sample T-Test**

A one-sample t-test is used to determine whether the mean of a single sample differs from a known or hypothesized population mean. Let's test whether the mean sepal length of the iris dataset significantly differs from a hypothesized population mean of 5.8.

Set up your hypotheses.

Null Hypothesis (H0): The mean sepal length in the iris dataset is equal to 5.8.

Alternative Hypothesis (Ha): The mean sepal length in the iris dataset is not equal to 5.8.

# Load the iris dataset

data(iris)

# Perform the one-sample t-test

result\_one\_sample <- t.test(iris$Sepal.Length, mu = 5.8)

# View the test result

result\_one\_sample

##Compare the p-value to your chosen significance level

#to determine whether to reject the null hypothesis.

**Two Independent Sample T-Test**

A two-independent-sample t-test is used to compare the means of two independent groups to determine if there is a significant difference between them.

Let's test whether there is a significant difference in sepal length (Sepal.Length) between two species of iris flowers: Setosa and Versicolor.

Set up your hypotheses.

Null Hypothesis (H0): The mean sepal length of Setosa and Versicolor iris flowers is equal.

Alternative Hypothesis (Ha): The mean sepal length of Setosa and Versicolor iris flowers is not equal.

# Filter data for Setosa and Versicolor species

setosa\_sepal\_length <- iris$Sepal.Length[iris$Species == "setosa"]

versicolor\_sepal\_length <- iris$Sepal.Length[iris$Species == "versicolor"]

# Perform the two-sample t-test

result\_two\_sample <- t.test(setosa\_sepal\_length, versicolor\_sepal\_length)

# View the test result

result\_two\_sample

**Paired sample T-Test**

We'll use the ToothGrowth dataset, which contains data on the length of tooth growth in guinea pigs under different supplement conditions.

Let's say we want to test if there is a significant difference in tooth growth between two delivery methods of vitamin C supplements: OJ (orange juice) and VC (ascorbic acid).

Null Hypothesis (H0): The mean tooth growth after using OJ is equal to the mean tooth growth after using VC.

Alternative Hypothesis (Ha): The mean tooth growth after using OJ is not equal to the mean tooth growth after using VC.

# Load the ToothGrowth dataset

data(ToothGrowth)

# View the first few rows of the dataset

head(ToothGrowth)

# Summary statistics of the dataset

summary(ToothGrowth)

# Structure of the dataset

str(ToothGrowth)

#Prepare the data

# Create vectors for OJ and VC

oj\_tooth\_length <- ToothGrowth$len[ToothGrowth$supp == "OJ"]

vc\_tooth\_length <- ToothGrowth$len[ToothGrowth$supp == "VC"]

# Perform the paired sample t-test

result\_pt <- t.test(oj\_tooth\_length, vc\_tooth\_length, paired = TRUE)

# View the test result

result\_pt

**Chi-squared Goodness fit test**

It is primarily used to assess if a sample data set follows a particular theoretical distribution or if there is a significant association between two categorical variables. For example:

- Checking if the distribution of blood types in a sample population follows the expected distribution (e.g., 40% Type A, 30% Type B, 20% Type AB, 10% Type O).

- Testing whether the frequency of certain traits in a population (e.g., eye color) matches expected Mendelian ratios.

- Investigating whether there is a significant association between gender and a particular medical condition (e.g., heart disease) by examining a contingency table of gender (male/female) and medical condition (present/absent).

- Studying whether there is a significant relationship between a person's education level and their voting preferences in an election.

Example: Let's say, using a hypothetical dataset, we want to test whether the distribution of political party affiliations in a sample of voters matches an expected distribution.

Null Hypothesis (H0): The observed distribution of voter party affiliations matches the expected distribution.

Alternative Hypothesis (Ha): The observed distribution of voter party affiliations is different from the expected distribution.

# Create a hypothetical dataset of voter party affiliations

voter\_data <- c("Republican", "Democrat", "Independent", "Republican", "Republican","Democrat", "Independent", "Independent", "Democrat", "Independent")

# Create a table of observed frequencies

observed <- table(voter\_data)

# Expected frequencies based on the assumed distribution

expected <- c(0.4 \* length(voter\_data), 0.4 \* length(voter\_data), 0.2 \* length(voter\_data))

# Calculate the Chi-squared statistic

chi\_squared <- sum((observed - expected)^2 / expected)

# Calculate the p-value

p\_value <- 1 - pchisq(chi\_squared, 2)

# Set the significance level

alpha <- 0.05

# Make a decision

if (p\_value < alpha) {

cat("Reject the null hypothesis. The observed distribution is different from the expected distribution.")

} else {

cat("Fail to reject the null hypothesis. The observed distribution is consistent with the expected distribution.")

}

# Display the Chi-squared statistic and p-value

cat("\nChi-squared statistic:", chi\_squared)

cat("\np-value:", p\_value)

**Chi-squared test of independence**

This test is used to determine whether there is a significant association or relationship between two categorical variables. In other words, it assesses whether the occurrence of one variable is independent of the occurrence of the other variable. It is often applied to contingency tables.

Example 1: We want to test if there is any significant statistical association between employee onboarding status and their turnover status.

# Import Employee Turnover data

emp <- read.csv("EmployeeTurnover.csv")

## construct a contingency table (cross tabulation)

tb1 <- xtabs(~Onboarding+Turnover, data = emp)

print(tb1)

## Chi-square test of independence

#Pearson's Chi-squared test #(2x2)

chisq.test(tb1)

# with p-value = 0.000001476, there is a significant statistical association

prop.table(tb1,2)

## Phi coefficient or pearson corr coeff

library(psych)

phi(tb1)

## Examine the observed and expected value

chisq <- chisq.test(tb1)

chisq$observed

chisq$expected

chisq$residuals ## pearson residuals

Example 2: We use housetasks data set. The data is a contingency table containing 13 housetasks and their distribution in the couple. Test the hypothesis whether wife’s housing tasks are independent of husband’s tasks at .05 significance level.

# Import the data

file\_path <- "http://www.sthda.com/sthda/RDoc/data/housetasks.txt"

housetasks <- read.delim(file\_path, row.names = 1)

head(housetasks)

#Contingency table can be visualized using

#the function balloonplot() [in gplots package].

library("gplots")

# 1. convert the data as a table

dt <- as.table(as.matrix(housetasks))

# 2. Graph

balloonplot(t(dt), main ="housetasks", xlab ="", ylab="",

label = FALSE, show.margins = FALSE)

#It’s also possible to visualize a contingency table as a mosaic plot.

library("graphics")

mosaicplot(dt, shade = TRUE, las=2,

main = "housetasks")

chisq <- chisq.test(housetasks)

chisq

As the p-value 0.0000003 is smaller than the .05 significance level, we do reject the null hypothesis that the wife’s housekeeping tasks habit is independent of her husband’s tasks.

**Correlation Coefficients/heat map**

We can use the cor.test() function in R to perform correlation tests between pairs of variables in your df5 data frame, which contains columns for height, weight, leaf area, shoot area, and flowers.

##Violin plot Using lessR package

library(summarytools)

library(tidyverse)

library(readxl)

library(corrr)

flower <- read\_excel("flower.xls")

## Select a few variables of interest

# Option 1: Table

flower %>% select(height,weight,leafarea, shootarea, flowers) %>%

correlate() %>% shave() %>%

fashion()

# Option 2: plot

flower %>% select(height,weight,leafarea, shootarea, flowers) %>%

correlate() %>% shave() %>%

rplot()

## Option 3

library(psych) # Load the psych package for the corr.test() function

flower %>% select(height,weight,leafarea, shootarea, flowers)->df5

# Run correlation tests for all pairs of variables in df5

correlation\_tests <- corr.test(df5)

# View the results

correlation\_tests

print(correlation\_tests,short=FALSE)

## Option 4

#install.packages("corrplot")

library(corrplot)

# Calculate the correlation matrix

correlation\_matrix <- cor(df5)

# Create a correlation heatmap

corrplot(correlation\_matrix,

method = "color", # Use color to represent correlations

type = "full", # Display the lower triangle of the matrix

tl.cex = 0.7, # Adjust text size for variable labels

tl.col = "black" # Set variable label color to black

)

Plot(Tenure\_Yrs, data = job)

**## When to use Pearson vs. spearman correlations**

library(dplyr)

library(GGally)

# Assuming "flower" is your data frame

flower <- flower %>%

mutate(nitrogen\_numeric = recode(nitrogen,

"high" = 1,

"medium" = 2,

"low" = 3))

flower %>% select(height,weight,nitrogen\_numeric, block) %>%

ggpairs

**##Cronbach Alpha/** **McDonald's ω (omega) is a measure of internal consistency reliability.**

Internal consistency, often measured using Cronbach's Alpha, is a statistical measure of how well the items within a scale or questionnaire correlate with each other. If Alpha is close to 1, it indicates high internal consistency, suggesting that the items in your scale are strongly correlated with each other. McDonald's ω (omega) is a measure of internal consistency reliability, similar to Cronbach's Alpha.

If Alpha is closer to 0, it suggests lower internal consistency, indicating that the items in your scale may not be measuring the same underlying construct consistently.

Typically, a Cronbach's Alpha of 0.70 or higher is considered acceptable for most research purposes, but the threshold can vary depending on the context and the specific field of study.

##Import Job satisfaction data

job <- read.csv("JobSatisfaction.csv")

library(jmv)

reliability(

data = job,

vars = c("TurnInt1", "TurnInt2","TurnInt3"),

omegaScale = TRUE,

meanScale = TRUE,

sdScale = TRUE,

corPlot = TRUE,

alphaItems = TRUE,

omegaItems = TRUE,

meanItems = TRUE,

sdItems = TRUE,

itemRestCor = TRUE)

reliability(

data = job,

vars = c("Engage1","Engage2","Engage3","Engage4"),

omegaScale = TRUE,

meanScale = TRUE,

sdScale = TRUE,

corPlot = TRUE,

alphaItems = TRUE,

omegaItems = TRUE,

meanItems = TRUE,

sdItems = TRUE,

itemRestCor = TRUE)

**Part V: ANOVA**

**One way ANOVA**

We'll use the InsectSprays dataset, which contains data on the effectiveness of different insect sprays. This dataset contains information about the count of insects killed by various insect sprays.

# Load the InsectSprays dataset

data(InsectSprays)

# View the first few rows of the dataset

head(InsectSprays)

Next, perform a one-way ANOVA to determine if there are significant differences in insect kill counts among different sprays. In this analysis, the dependent variable is the insect count, and the independent variable is the type of insect spray. Also, we may want to perform post-hoc tests (e.g., Tukey's HSD or pairwise t-tests) to identify which specific groups differ from each other.

# Perform the one-way ANOVA

anova\_result <- aov(count ~ spray, data = InsectSprays)

# Summarize the ANOVA results

summary(anova\_result)

#Post-Hoc Tests

InsectSprays %>%

aov(count ~ spray, data = .) %>%

TukeyHSD() #%>%

plot()

**Two-way ANOVA**

We'll use the ToothGrowth dataset, which contains data related to the effect of two supplements on tooth growth. This dataset contains information about the length of tooth growth in guinea pigs exposed to different supplements and doses. We perform a two-way ANOVA to determine if there are significant effects of both the "supplement" and "dose" factors on tooth growth. In this analysis, "len" (tooth length) is the dependent variable, and "supp" (supplement type) and "dose" (dose level) are the independent variables. We also want to perform post-hoc tests to further investigate the differences between groups or levels.

# Load the ToothGrowth dataset

data(ToothGrowth)

# View the first few rows of the dataset

head(ToothGrowth)

# Perform the two-way ANOVA

anova\_result <- aov(len ~ supp \* dose, data = ToothGrowth)

# Summarize the ANOVA results

summary(anova\_result)

#Post-Hoc Tests

ToothGrowth %>%

aov(len ~ supp \* dose, data = .) %>%

TukeyHSD() %>%

plot()

Ref:

* Hadley Wickham’s Advanced R book
* Roger Peng’s R Programming for Data Science book
* DataCamp’s Intermediate R course
* Coursera’s R Programming course
* https://ggplot2-book.org/getting-started.html
* Data Carpentry (http://datacarpentry.org/), data camp, data quest, Kaggle
* Harvard Chan Bioinformatics Core (HBC) under the open access terms of the Creative Commons Attribution license (CC BY 4.0),
* The Book of R: A First Course in Programming and Statistics by Tilman M. Davies
* UC Business Analytics R Programming Guide and R\_bootcamp