# ED308: Parametric Inference Lab 2

# Part One: Correlation

Define x and y as vector of same length and then:

cor(x, y, method = c("pearson", "kendall", "spearman"))

cor.test(x, y, method=c("pearson", "kendall", "spearman"))

cor(x, y, method = "pearson", use = "complete.obs")

my\_data <- mtcars

head(my\_data, 6)

library("ggpubr")

ggscatter(my\_data, x = "mpg", y = "wt", add = "reg.line", conf.int = TRUE, cor.coef = TRUE, cor.method = "pearson", xlab = "Miles/(US) gallon", ylab = "Weight (1000 lbs)")

# Shapiro-Wilk normality test for mpg

shapiro.test(my\_data$mpg)

# Shapiro-Wilk normality test for wt

shapiro.test(my\_data$wt)

result <- cor.test(my\_data$wt, my\_data$mpg, method = "pearson")

result

res2 <- cor.test(my\_data$wt, my\_data$mpg, method="kendall")

res2

res3 <-cor.test(my\_data$wt, my\_data$mpg, method = "spearman")

res3

# Load data

data("mtcars")

my\_data <- mtcars[, c(1,3,4,5,6,7)]

# print the first 6 rows

head(my\_data, 6)

res <- cor(my\_data)

round(res, 2)

cor(my\_data, use = "complete.obs")

# Install Hmisc package:

install.packages("Hmisc")

library("Hmisc")

res2 <- rcorr(as.matrix(my\_data))

res2

install.packages("corrplot")

library(corrplot)

corrplot(res, type = "upper", tl.col = "black", tl.srt = 45)

install.packages("PerformanceAnalytics")

library("PerformanceAnalytics")

my\_data <- mtcars[, c(1,3,4,5,6,7)]

chart.Correlation(my\_data, histogram=TRUE, pch=19)

# Part Two: Comparing Means

t.test(x, mu = 0, alternative = "two.sided")

set.seed(1234)

my\_data <- data.frame(

name = paste0(rep("M\_", 10), 1:10),

weight = round(rnorm(10, 20, 2), 1))

# Print the data

my\_data

# Statistical summaries of weight

summary(my\_data$weight)

library(ggpubr)

ggboxplot(my\_data$weight,

ylab = "Weight (g)", xlab = FALSE,

ggtheme = theme\_minimal())

shapiro.test(my\_data$weight)

# One-sample t-test

mean(my\_data$weight)

# two sided test (Ho: the mean is equal to 25g)

res <- t.test(my\_data$weight, mu = 25)

# Printing the results

res

# one-sided test (Ho: the mean is greater than or equal to to 25g)

t.test(my\_data$weight, mu = 25,

alternative = "less")

wilcox.test(x, mu = 0, alternative = "two.sided")

# One-sample wilcoxon test

res <- wilcox.test(my\_data$weight, mu = 25)

# Printing the results

res

# Part Three: Comparing Means of two Independent Groups

# R function to compute unpaired two-samples t-test

t.test(x, y, alternative = "two.sided", var.equal = FALSE)

# Data in two numeric vectors

women\_weight <- c(38.9, 61.2, 73.3, 21.8, 63.4, 64.6, 48.4, 48.8, 48.5)

men\_weight <- c(67.8, 60, 63.4, 76, 89.4, 73.3, 67.3, 61.3, 62.4)

# Create a data frame

my\_data <- data.frame(

group = rep(c("Woman", "Man"), each = 9),

weight = c(women\_weight, men\_weight)

)

print(my\_data)

# Computing summary statistics by groups

library(dplyr)

group\_by(my\_data, group) %>%

summarise(

count = n(),

mean = mean(weight, na.rm = TRUE),

sd = sd(weight, na.rm = TRUE)

)

# Plot weight by group and color by group

library("ggpubr")

ggboxplot(my\_data, x = "group", y = "weight",

color = "group", palette = c("#00AFBB", "#E7B800"),

ylab = "Weight", xlab = "Groups")

# Shapiro-Wilk normality test for Men's weights

with(my\_data, shapiro.test(weight[group == "Man"]))

# Shapiro-Wilk normality test for Women's weights

with(my\_data, shapiro.test(weight[group == "Woman"]))

res.ftest <- var.test(weight ~ group, data = my\_data)

res.ftest

t.test(women\_weight, men\_weight, var.equal = TRUE, alternative="less")

t.test(weight ~ group, data = my\_data, var.equal = TRUE, alternative = "greater")

wilcox.test(x, y, alternative = "two.sided")

library(dplyr)

group\_by(my\_data, group) %>%

summarise(

count = n(),

median = median(weight, na.rm = TRUE),

IQR = IQR(weight, na.rm = TRUE)

)

res <- wilcox.test(women\_weight, men\_weight)

res