**I. Ước lượng tham số (Parameter Estimation)**.

**1. Ước lượng tham số (Parameter Estimation)**

**1.1. Ước lượng điểm (Point Estimation)**

**Theory:**

* A point estimate gives a single value as an estimate of an unknown population parameter.
* Common estimators:
  + Mean: xˉ\bar{x}
  + Variance: s2s^2
  + Proportion: p^\hat{p}

**R Example:**

# Sample data

set.seed(123)

data <- rnorm(30, mean = 100, sd = 15)

# Point estimation

mean\_est <- mean(data)

var\_est <- var(data)

sd\_est <- sd(data)

mean\_est # Estimated mean

var\_est # Estimated variance

sd\_est # Estimated standard deviation

**1.2. Khoảng tin cậy (Confidence Interval)**

**1.2.1. Khoảng tin cậy cho mean (z.test, t.test)**

**Theory:**

* Use **t-distribution** when population SD is unknown and sample size is small.
* Use **z-distribution** when population SD is known or sample size is large.

**R Example:**

# t-test for confidence interval (population SD unknown)

t.test(data, conf.level = 0.95)

# If population SD known (e.g. sd = 15), use z-interval

library(BSDA)

z.test(data, sigma.x = 15, conf.level = 0.95)

**1.2.2. Khoảng tin cậy cho phương sai (Confidence Interval for Variance)**

**Theory:**

* Use Chi-squared distribution:

((n−1)s2χ1−α/22,(n−1)s2χα/22)\left( \frac{(n-1)s^2}{\chi^2\_{1-\alpha/2}}, \frac{(n-1)s^2}{\chi^2\_{\alpha/2}} \right)

**R Example:**

n <- length(data)

s2 <- var(data)

alpha <- 0.05

chi\_lower <- qchisq(1 - alpha/2, df = n - 1)

chi\_upper <- qchisq(alpha/2, df = n - 1)

ci\_lower <- (n - 1) \* s2 / chi\_lower

ci\_upper <- (n - 1) \* s2 / chi\_upper

c(ci\_lower, ci\_upper)

library(EnvStats)

varTest(x=data.c1.vector, conf.level= 0.95)

**1.2.3. Khoảng tin cậy cho tỷ lệ (Confidence Interval for Proportion)**

**Theory:**

* For large nn, the sampling distribution of p^\hat{p} is approximately normal.

**R Example:**

# Suppose 60 successes out of 100

x <- 60

n <- 100

prop.test(x, n, conf.level = 0.95, correct = FALSE)

1.2.4. Ước lượng

II. **Kiểm định giả thiết thống kê (Hypothesis Testing)**.

**2. Kiểm định giả thiết thống kê (Hypothesis Testing)**

**2.1. So sánh mean với scalar (One-sample test for mean)**

**Theory:**

* H₀: μ = μ₀ (e.g. mean = 100)
* Use **t-test** if population SD unknown; use **z-test** if known

**R Example:**

# Sample data

data <- c(102, 98, 101, 100, 99, 97, 103, 100)

# t-test

t.test(data, mu = 100)

# z-test (if known sigma, e.g. sigma = 5)

library(BSDA)

z.test(data, mu = 100, sigma.x = 5)

**2.2. So sánh proportion với scalar (One-sample test for proportion)**

**Theory:**

* H₀: p = p₀
* Use normal approximation for large nn

**R Example:**

# 60 successes in 100 trials, test if p = 0.5

prop.test(60, 100, p = 0.5, correct = FALSE)

**2.3. So sánh Standard Deviation với scalar (One-sample test for variance)**

**Theory:**

* Use **Chi-squared test**:

χ2=(n−1)s2σ02\chi^2 = \frac{(n-1)s^2}{\sigma\_0^2}

**R Example:**

# Test if variance = 25 (SD = 5)

n <- length(data)

s2 <- var(data)

sigma0 <- 5

chi\_stat <- (n - 1) \* s2 / sigma0^2

p\_value <- 2 \* min(pchisq(chi\_stat, df = n - 1), 1 - pchisq(chi\_stat, df = n - 1))

chi\_stat

p\_value

**2.4. So sánh 2 mean (Two-sample test for mean)**

**Theory:**

* H₀: μ₁ = μ₂
* Use:
  + **t-test (equal or unequal variance)**
  + **paired t-test** if data are paired

**Step: Kiểm định PPC, kiểm định phương sai bằng nhau**

**R Example:**

x <- rnorm(30, mean = 100, sd = 15)

y <- rnorm(30, mean = 105, sd = 15)

# Independent samples t-test

t.test(x, y, var.equal = TRUE) # Use var.equal = FALSE if unsure

# Paired t-test (if same subject before/after)

t.test(x, y, paired = TRUE)

**2.5. So sánh 2 proportion (Two-sample test for proportion)**

**R Example:**

# Group 1: 50 successes / 100

# Group 2: 60 successes / 120

prop.test(c(50, 60), c(100, 120), correct = FALSE)

**2.6. So sánh 2 Standard Deviation (Two-sample test for variance)**

**Theory:**

* Use **F-test**:

F=s12s22F = \frac{s\_1^2}{s\_2^2}

**R Example:**

var.test(x, y)

**2.7. So sánh nhiều mean (ANOVA)**

**Theory:**

* H₀: all means are equal
* Use **one-way ANOVA**

**R Example:**

group <- rep(c("A", "B", "C"), each = 10)

value <- c(rnorm(10, 100), rnorm(10, 105), rnorm(10, 102))

anova\_data <- data.frame(group, value)

anova\_result <- aov(value ~ group, data = anova\_data)

summary(anova\_result)

Step

* Kiểm tra phân phối chuẩn
* Đồng nhất phương sai

# Tukey HSD

TukeyHSD(aov\_sanluong)

# xem lwr và upr có chứa điểm 0 không

# Kết quả: mean thằng nào lớn hơn thằng nào

**2.8. So sánh nhiều proportion**

**Theory:**

* Use **Chi-squared test for proportions**

**R Example:**

# Rows: groups, columns: success/failure

# Example: A: 30/70, B: 40/60, C: 50/50

data <- matrix(c(30,70, 40,60, 50,50), nrow = 3, byrow = TRUE)

chisq.test(data)

**2.9. So sánh nhiều Standard Deviation**

**Theory:**

* No built-in R function like ANOVA for variance, but can use Bartlett or Levene test.

**R Example (Bartlett’s test):**

bartlett.test(value ~ group, data = anova\_data)

**2.10. Tiêu chuẩn phi tham số (Non-parametric tests)**

**Use when data is not normal.**

* Wilcoxon, Kruskal-Wallis, etc.

**R Example (Wilcoxon and Kruskal-Wallis):**

# Wilcoxon test (2 groups)

wilcox.test(x, y)

# Kruskal-Wallis (3+ groups)

kruskal.test(value ~ group, data = anova\_data)

**2.11. Tiêu chuẩn Chi squared (Chi-squared tests)**

**2.11.1. Kiểm tra sự phù hợp với phân bố (Goodness of Fit)**

# Observed: 50, 30, 20

# Expected proportions: 0.5, 0.3, 0.2

chisq.test(x = c(50,30,20), p = c(0.5,0.3,0.2))

**2.11.2. So sánh các phân bố xác suất (Compare distributions)**

# Similar to above, compare two empirical distributions

**2.11.3. So sánh nhiều tỷ lệ (Compare proportions across groups)**

# Similar to 2.8, already shown

**2.11.4. Kiểm tra tính độc lập (Test for independence)**

# Contingency table

tbl <- matrix(c(30, 10, 20, 40), nrow = 2)

chisq.test(tbl)

**Cách tạo data frame:**

# Nhập dữ liệu

ngo <- c(17, 18, 28, 28, 28, 22, 17)

dautuong <- c(17, 18, 19, 30, 20, 19, 18)

luami <- c(13, 23, 24, 25, 24, 13, 21)

# Gộp vào một data frame

sanluong <- c(ngo, dautuong, luami)

loai <- factor(rep(c("Ngo", "Dautuong", "Luami"), times = c(7,7,7)))

data <- data.frame(Sanluong = sanluong, Loai = loai)

**Tạo data frame 2 chiều**

data.c4.tren\_trung\_binh <- data.frame(

hoc\_luc = c("tren\_trung\_binh", "tren\_trung\_binh", "tren\_trung\_binh"),

y\_kien = c("tot", "vua", "xau"),

so\_luong = c(175, 124, 92)

)

data.c4.trung\_binh <- data.frame(

hoc\_luc = c("trung\_binh", "trung\_binh", "trung\_binh"),

y\_kien = c("tot", "vua", "xau"),

so\_luong = c(118, 110, 126)

)

data.c4.duoi\_trung\_binh <- data.frame(

hoc\_luc = c("duoi\_trung\_binh", "duoi\_trung\_binh", "duoi\_trung\_binh"),

y\_kien = c("tot", "vua", "xau"),

so\_luong = c(127, 82, 147)

)

data.c4 <- rbind(data.c4.tren\_trung\_binh, data.c4.trung\_binh, data.c4.duoi\_trung\_binh)

**Kiểm tra phân phối chuẩn**

shapiro.test(data$Sanluong[data$Loai == "Ngo"])

# Nếu p-value > 0.05 => không bác bỏ H0 => phân phối chuẩn

**Kiểm tra phương sai các nhóm phải đồng nhất**

# install.packages("car")

library(car)

leveneTest(data = data, Sanluong ~ Loai)

# Nếu p-value > 0.05 => không bác bỏ H0 => phương sai đồng nhất

# Non-parametric

kruskal.test(SanLuong ~ Loai, data = data)

**Bài tập mẫu về**

Hãy kiểm chứng ở mức ý nghĩa 5% xem tỷ lệ học sinh “đánh giá phương pháp giáo dục mới là tốt” có như nhau trong 3 nhóm học sinh trên hay không? Nhóm nào có tỷ lệ đánh giá này là cao nhất.

# Nhập data

data.c4.tren\_trung\_binh <- data.frame(

hoc\_luc = c("tren\_trung\_binh", "tren\_trung\_binh", "tren\_trung\_binh"),

y\_kien = c("tot", "vua", "xau"),

so\_luong = c(175, 124, 92)

)

data.c4.trung\_binh <- data.frame(

hoc\_luc = c("trung\_binh", "trung\_binh", "trung\_binh"),

y\_kien = c("tot", "vua", "xau"),

so\_luong = c(118, 110, 126)

)

data.c4.duoi\_trung\_binh <- data.frame(

hoc\_luc = c("duoi\_trung\_binh", "duoi\_trung\_binh", "duoi\_trung\_binh"),

y\_kien = c("tot", "vua", "xau"),

so\_luong = c(127, 82, 147)

)

data.c4 <- rbind(data.c4.tren\_trung\_binh, data.c4.trung\_binh, data.c4.duoi\_trung\_binh)

library(dplyr)

data.c4.updated <-

data.c4 %>%

group\_by(hoc\_luc) %>%

mutate(tong\_y\_kien = sum(so\_luong)) %>%

mutate(ty\_le = so\_luong / tong\_y\_kien)

data.c4.tot <- data.c4.updated %>%

filter(y\_kien == 'tot')

p <- sum(data.c4.tot$so\_luong) / sum(data.c4.tot$tong\_y\_kien)

q = 1-p

# Kiểm tra giả thiết về xấp xỉ phân phối chuẩn: với từng mẫu A và B:

# nA \* p > 5 , nA \* q > 5. Trong đó nA là số quan sát trong mẫu A,

# p là tỷ lệ thành công trên TỔNG QUAN SÁT (tỷ lệ trên cả A và B).

# q là tỷ lệ thất bại trên tổng quan sát. Tương tự với mẫu B.

n.duoi <- data.c4.tot %>%

filter(hoc\_luc == 'duoi\_trung\_binh') %>%

pull(tong\_y\_kien)

n.tb <- data.c4.tot %>%

filter(hoc\_luc == 'trung\_binh') %>%

pull(tong\_y\_kien)

n.tren <- data.c4.tot %>%

filter(hoc\_luc == 'tren\_trung\_binh') %>%

pull(tong\_y\_kien)

n.duoi.success <- data.c4.tot %>%

filter(hoc\_luc == 'duoi\_trung\_binh') %>%

pull(so\_luong)

n.tb.success <- data.c4.tot %>%

filter(hoc\_luc == 'trung\_binh') %>%

pull(so\_luong)

n.tren.success <- data.c4.tot %>%

filter(hoc\_luc == 'tren\_trung\_binh') %>%

pull(so\_luong)

n.duoi \* p > 5 # TRUE

n.tb \* p > 5 # TRUE

n.tren \* p > 5 # TRUE

n.duoi \* q > 5 # TRUE

n.tb \* q > 5 # TRUE

n.tren \* q > 5 # TRUE

# Điều kiện được thỏa mãn. Ta có thể sử dụng proportion test

# Kiểm định xem tỷ lệ ý kiến tốt giữa các nhóm học lực có khác nhau không

# H0: p.duoi = p.tb = p.tren

# H1: Có ít nhất 1 nhóm học lực có tỷ lệ khác nhau

prop.test(

x=c(n.duoi.success, n.tb.success, n.tren.success),

n=c(n.duoi, n.tb, n.tren),

conf.level = 1 - 0.05

)

# Kiểm tra từng cặp

# Dưới TB và TB

# H0: p.duoi = p.tb

# H1: p.duoi > p.tb

prop.test(

x=c(n.duoi.success, n.tb.success),

n=c(n.duoi, n.tb),

alternative="greater",

conf.level = 1 - 0.05

)

**Bài tập mẫu về**

Trọng lượng của các gói hàng được đóng bằng máy tự động là ĐLNN có phân bố chuẩn, cân ngẫu nhiên một số gói được kết quả như sau:

Trọng lượng 0,8 0,9 1,0 1,1 1,2

Số gói 1 2 5 3 2

a/Tìm ước lượng hợp lý cực đại của giá trị trung bình và phương sai của trọng lượng gói hàng.

b/Tìm ước lượng các giá trị trung bình và phương sai với độ tin cậy 95%.

data.c1 <- data.frame(

trong\_luong = c(0.8, 0.9, 1, 1.1, 1.2),

so\_goi = c(1, 2, 5, 3, 2)

)

data.c1.vector <- rep(data.c1$trong\_luong, data.c1$so\_goi)

# a/ Tìm ước lượng hợp lý cực đại của giá trị trung bình và phương sai

# của trọng lượng gói hàng.

# Có 2 cách như dưới, cần xác nhận với cô nên dùng cách nào.

# Cách 1: dùng luôn công thức toán học. Tham khảo trang 216

# sách cô gửi (Introduction to Probability and Statistic in R)

n <- length(data.c1.vector)

mu\_hat <- mean(data.c1.vector)

var\_hat <- var(data.c1.vector) \* (n-1) / n

# Cách 2: tính máy bằng hàm tối ưu mle.

# Logic tính tương tự như câu 1b đề thi deminhhoa. Tính hàm nll, rồi đưa vào

# hàm mle

library(stats4)

mu <- mean(data.c1.vector)

sigma2 <- var(data.c1.vector)

minuslogL <- function(mu, sigma2){

-sum(dnorm(data.c1.vector, mean = mu, sd = sqrt(sigma2), log = TRUE))

}

mle(minuslogl=minuslogL, start=list(mu=mu, sigma2=sigma2))

# mu: 1.02308039 , sigma2: 0.01254663

**Bài tập mẫu về**

b/ Hãy ước lượng tỷ lệ đạt chuẩn của công ty theo mẫu trên với α=5%.

data.c2.vector <- c(30.1, 32.7, 22.5, 27.5, 27.7, 29.8, 28.9, 31.4, 31.2, 24.3, 26.4, 22.8, 29.1, 33.4, 32.5, 21.7)

# Kiểm tra giả định: n\*p0 > 5 và n\*q > 5.

n.c2 <- length(data.c2.vector)

n.c2.success <- length(data.c2.vector[data.c2.vector >= 30])

p0 = n.c2.success / n.c2

q = 1 - p0

n.c2 \* p0 > 5 # TRUE

n.c2 \* q > 5 # TRUE

# Tính khoảng tin cậy

prop.test(x=n.c2.success, n=n.c2, correct=FALSE) # [0.1848123, 0.6135896]