**TỔNG HỢP CÁC CÂU LỆNH TRONG R**

**Các lệnh cơ bản**

sample(x, size, replace = FALSE, prob = NULL): dùng để **mô phỏng**. Trong đó x là vector dữ liệu, size là cỡ mẫu, replace là các giá trị được phép giống nhau hay không, prob là xác suất từng phần tử trong x xuất hiện.

data.frame(x1, x2): tạo dataframe từ x1 và x2 rồi lưu vào biến data.

read.table(“\*.txt”, header = TRUE): đọc dữ liệu file .txt

names(): liệt kê các cột và tên cột.

read.csv(“\*csv”, header = TRUE): đọc dữ liệu file .csv

seq(from, to, by): tạo dãy số từ from đến to, bước nhảy là by.

rep(x, times): lặp lại biến x times lần.

solve(): tìm nghịch đảo của ma trận.

subset(data, <condition>): tách data thành các data frame dựa vào điều kiện.

source(‘script.name.r’): gọi lại script.

function.name <- function(parameters) {<statements>}: khai báo và định nghĩa hàm.

replicate(n, function): lặp lại function n lần.

**Một số lệnh khác**

setwd(‘path’)

getwd()

list.files(), dir(): liệt kê tất cả file trong thư mục

save.image(‘name.rda’): lưu workspace

save(x, file=’name.rda’): lưu biến x

load(‘name.rda’): load file .rda

rm(x): xóa biến x

rm(list=ls()): xóa tất cả biến

ls(): liệt kê tất cả biến

str(x): xem thông tin biến x

**Đồ thị và một số thứ liên quan**

plot(x, y, type = “h”, xlab = “…”, ylab = “…”, add = TRUE): vẽ đồ thị y = f(x), type = “h” là vẽ cột dọc, “p” là điểm, “l” là đường, xlab ylab là tên trục hoành trục tung sau khi vẽ, add = TRUE là vẽ chồng lên đồ thị đã có.

curve(expr, from, to, xlab, ylab, add = TRUE): vẽ đồ thị **dạng đường cong liên tục (dùng cho BNN liên tục)**, expr thường là một hàm hoặc một lời gọi tới cái gì đó chứa x, from to là giới hạn trên và dưới của trục hoành đồ thị.

plot(stepfun(x, f(x)), xlab = “…”, ylab = “…”): vẽ đồ thị **dạng bậc thang (dùng cho BNN rời rạc)**, trong đó f(x) phải dài hơn x một đơn vị.

hist(x, freq, break): vẽ **biểu đồ tần suất**.

boxplot(x):

**Các loại phân phối**

binom(x, n, p): phân phối nhị thức.

geom(x, p): phân phối hình học.

pois(x, lambda): phân phối poison.

unif(x, a, b): phân phối đều.

exp(x, lambda): phân phối mũ.

norm(x, my, sigma): phân phối chuẩn.

chisq(x, df = n – 1): phân phối chi bình phương.

Đặt d-p-q-r trước các hàm theo quy tắc sau: d mật độ, p phân phối, q phân vị, r mô phỏng. VD: rpois(n, lambda) là mô phỏng phân phối poison, dnorm(x, mu, sigma) là mật độ phân phối chuẩn…

**Ước lượng**

mean(x): trung bình mẫu.

length(x): cỡ mẫu.

sd(x): độ lệch chuẩn mẫu.

qnorm(1 – alpha/2): tìm phân vị trong phân phối chuẩn.

pt(t, n – 1, lower.tail = FALSE): tìm phân vị P(t(n – 1) > t) trong phân phối Student, lower.tail mặc định là TRUE. Nếu là TRUE, hàm sẽ tìm phân vị P(t(n – 1) < t).

**Kiểm định một mẫu**

t.test(x, alternative = “…”, mu, conf.level): kiểm định **kỳ vọng**, x là vector dữ liệu, alternative = “two.sides” là kiểm định 2 phía, “less” và “greater” lần lượt là H1 nhỏ hơn và H1 lớn hơn, nếu alternative trống thì mặc định là “two.sides”, mu = mu0 là giá trị cần kiểm định, conf.level là độ tin cậy (nếu để trống mặc định 0.95).

result$statistic

result[['statistic']]

unname(result[['statistic']])): cả 3 dùng để rút ra thống kê kiểm định với result = t.test(…).

prop.test(y, n, p = p0, alternative = “two.sides”, conf.level = 0.95): y là số phần tử thỏa, n là cỡ mẫu, p0 là giá trị cần kiểm định.

**Kiểm định hai mẫu**

var.test(x1, x2): kiểm định phương sai, nếu p-value < alpha thì phương sai khác nhau.

t.test(x1, x2, alternative = “two.sides”, var.equal, conf.level = 0.95): kiểm định **kỳ vọng cho hai mẫu độc lập**, x1 x2 là hai mẫu dữ liệu, var.equal = FALSE nếu phương sai hai mẫu khác nhau, TRUE nếu như nhau, nếu để trống thì mặc định là FALSE.

t.test(data ~ group, alternative = “two.sides”, var.equal, conf.level = 0.95): tương tự như trên nhưng trong trường hợp này, các cặp giá trị được chứa trong cùng 1 biến data và được phân biệt bởi biến group, thay vì x1 x2.

t.test(x1, x2, alternative = “two.sides”, paired = TRUE, var.equal, conf.level = 0.95) hoặc

t.test(data ~ group, alternative = “two.sides”, paired = TRUE, var.equal, conf.level = 0.95): kiểm định **kỳ vọng** **hai mẫu không độc lập**.

prop.test(y = c(y1, y2), n = c(n1, n2), alternative = “…”, conf.level, correct = TRUE): kiểm định tỷ lệ hai mẫu, y và n phải **là các vector**, y1 y2 là số phần tử thỏa ở mẫu 1 và 2, n1 n2 là cỡ mẫu ở mẫu 1 và 2.

**LƯU Ý:** khi dùng các hàm t.test, prop.test, ta thường so sánh p-value với alpha để kết luận, nếu p-value < alpha thì bác bỏ H0.

**Hồi quy**

abline(lm(y~x): vẽ đường thẳng hồi quy.

lm(y~x): các giá trị cơ bản của phân tích hồi quy.

GIẢI THỬ ĐỀ CUỐI KỲ 2016 – 2017

A A E B A B A C A C D C D B A D A D C D A B A D D A D B

Đúng: 24/28