

Thực hành xác suất thống kê

Chương I: Giới thiệu R

Chương II: Giới thiệu R(tt)

#Bài 1:

```
bai2.1<-function(n,i){  
  
  arr<-sample(n);  
  print('Vecto ngẫu nhiên', arr);  
  tong<-0;  
  for(k in 1:i) tong<-tong+arr[k];  
  print('Tong tích luy');  
  tong;  
}
```

#Bài 2:

```
bai2.2<-function(){  
  
  R<-c(3:20);  
  V<-c(4*pi*R^3/3);  
  TT<-data.frame(R,V);  
  TT  
}
```

#Bài 3:

```
bai2.3<-function()  
{  
  data<-read.csv("data/data01.csv",  
header=TRUE);  
  Index<-c();  
  for(i in 1:nrow(data))  
  {  
    if(data$Age[i] <= 60)  
      Index[i] <- 0  
    else if(data$Age[i] <= 70)  
      Index[i] <- 1  
    else if(data$Age[i] <= 80)  
      Index[i] <- 2  
    else Index[i] = 3;  
  }  
  #Add a column  
  data<-cbind(data, Index);  
  
  print(data);  
  data  
}  
bai2.3()
```

#Bài 5:

```
phanvi<-function(X, P)  
{  
  pv<-0;  
  
  sort(X);  
  
  i<-(P/100)*length(X);  
  
  if(is.integer(i))  
    pv<-(X[i]+X[i+1])/2  
  else  
    pv<-X[round(i,0)];  
  
  print(i);  
  pv  
}  
phanvi(sample(50),10)
```

#Bài 4:

```
bai2.4<-function(){  
  
  data<-data.frame(read.csv("Data/data11.csv", header=TRUE));  
  
  cat("Chieu cao cay nho nhat la:", data$a[which.min(data$n)], "-",  
      data$b[which.min(data$n)], "\n");  
  
  cat("Chieu cao cay lon nhat la: ", data$a[which.max(data$n)], "-",  
      data$b[which.max(data$n)], "\n");
```

```

tbm<-0;

for(i in 1:nrow(data)){
  mid<-(data$a[i]+data$b[i])/2.0;
  tbm<-tbm + mid*data$n[i];
}

tbm<-tbm/sum(data$n);

cat("Trung binh mau la: ", tbm,"\n");

psm<-0;

for(i in 1:nrow(data)){
  mid<-(data$a[i]+data$b[i])/2.0;
  psm<-psm+(mid-tbm)^2*data$n[i];
}

psm<-psm/sum(data$n);
cat("Phuong sai mau la: ", psm, "\n");
}
bai2.4()

```

Chương III: Biến ngẫu nhiên

#Bài 1:

```

bai3.1<-function(){
  f<-function(p){
    0.07*p^(-0.93);
  }
  F<-function(a){
    integrate(function(x) f(x),
              lower=0, upper=a)$value
  }
  F<-Vectorize(F);

  print('Xac suat P(P<0.2) la: ');
  print(F(0.2));

  print('Tich phan fp(p) tu 0->1: ');
  print(F(1));
}
bai3.1()

```

#Bài 2:

```

bai3.2<-function(){
  x<-sample(1:5,100,TRUE,
           c(0.1,0.2,0.4,0.2,0.1));

  print('Xac suat thuc nghiem: ');
  print(table(x)/100);

  plot(1:5, table(x)[1:5]/100,
       type="h");
}
bai3.2()

```

Chương IV: Một số phân phối xác suất thông dụng

#Bài 1: <pre> bai4.1<-function(n, N, M) { print('Ham phan phoi: '); plot(0:n, pbinom(0:n,n,M/N),type="h", ylab="P(X<=x)", main=""); } bai4.1(15,100,25) </pre>	#Bài 2: <pre> bai4.2<-function(n, N, M){ #Cách 1 print(sum(dbinom(5:12, n,M/N))); #Cách 2 print(pbinom(12,n,M/N) - pbinom(4, n,M/N)); # Vì $F_X(x) = P(X \leq x)$ } bai4.2(15,100,25) </pre>
#Bài 3: <pre> bai4.3<-function() { curve(dexp(x,0.6),0,10); curve(dexp(x,0.3),0,10, add=T); #add=T: vẽ thêm print('Dien tich duoi ham 1: '); print(pexp(10,0.6)); # x chạy từ $-\infty \rightarrow 10$ print('Dien tich duoi ham 2: '); print(pexp(10,0.3)); # x chạy từ $-\infty \rightarrow 10$ } bai4.3() </pre>	#Bài 4: <pre> bai4.4<-function(lamda) { plot(0:8, dpois(0:8, lamda), type="h"); } bai4.4(1) </pre>
#Bài 5: <pre> bai4.5<-function(){ a <- c(1:10) b <- c(1:10) par(fig=c(0, (2/3), 0, 1)) par(new=TRUE) plot(a, b) par(fig=c((2/3), 1, 0, 1)) par(new=TRUE) plot(a, b) } bai4.5() </pre>	#Bài 6: <pre> bai4.6<-function() { dev.off(); plot(0:50, dbinom(0:50,50,0.08), ylim=c(0,0.25)); par(new = TRUE); plot(0:50, dpois(0:50, 4), color="red",ylim=c(0,0.25)); } bai4.6() </pre>
#Bài 7: <pre> bai4.7<-function() { dev.off(); plot(0:50, dbinom(0:50,50,0.4)); curve(dnorm(x,20,sqrt(12)), add=T); } bai4.7() </pre>	

Chương V: Phân phối mẫu

<pre>#Bài 1 (Only) Y<-function() { X=rnorm(2); return (sum(X^2)) } MauY=function(n) { replicate(n,Y()) } Layout(matrix(1:3))</pre>	<pre>n=100 hist(MauY(n),freq = 0,breaks=100) curve(dchisq(x,df=2),add=T) n=1000 hist(MauY(n),freq = 0,breaks=100) curve(dchisq(x,df=2),add=T) n=10000 hist(MauY(n),freq = 0,breaks=100) curve(dchisq(x,df=2),add=T)</pre>
---	---

Chương VI: Lý thuyết mẫu

<pre>#Bài 1:</pre>	<pre>#Bài 2: #a) data=read.csv('data01.csv', header=TRUE) mean(data\$FPSA); #b) plot(data\$FPSA,type='l') boxplot(data\$FPSA, data = InsectSprays, col = "lightgray") #c) data\$FPSA[data\$K==0] data\$FPSA[data\$K==1] #d) data2=read.csv('data02.csv',header=TRUE) data=merge(data,data2,by='K') #e) tPSA=data\$Age tPSA[data\$Age<=30]=0 tPSA[data\$Age>30& data\$Age<=50]=1 tPSA[data\$Age>50]=2 tPSA summary(tPSA)</pre>
--------------------	---

Chương VII: Ước lượng tham số thống kê

#Bài 1: <pre> n=35 a=rnorm(n,10,5) alpha=0.05 epsilon=qnorm(1-alpha/2)*5/sqrt(n) mean=sum(a)/n cat(mean-epsilon); cat(mean+epsilon); </pre>	#bai 2 <pre> df=read.csv("data31.csv",header = T) profit=df[,2] ci.mean=function(x,alpha) { xm=mean(x) n=length(x) s=sd(x) e=qt(df=n-1,1-alpha/2)*s/sqrt(n) c(xm-e,xm+e) } ci.mean(profit,1-0.95) ci.mean(profit,1-0.99) </pre>
#Bai 03 <pre> data=read.csv("Data/data32.csv", header =T); time = data[,1] ci.mean(time, 1 - 0.95) ci.prop <- function(f, n, alpha) { ti_le = f / n; dung_sai = qnorm(1-alpha/2)* sqrt(ti_le*(1-ti_le)/ length(x)); cat(ti_le - dung_sai, ti_le + dung_sai); } cnt = length(time[time > 5]) ci.prop(cnt, length(time), 1- 0.9); </pre>	#Bai 05 <pre> ktc.tb=function(mean, adj,s,n,alpha) { #Da biet DLC cua tong the if (adj == T) { epsilon=qnorm(1-alpha/2)*s/sqrt(n) } else { if (n >= 30) epsilon=qnorm(1-alpha/2)*s/ sqrt(n); else epsilon = qt(1-alpha/2, df=n-1)*s/sqrt(n) } cat(mean - epsilon, mean + epsilon) } </pre>
#Bai 04 <pre> x <- rep(1.3, 6) x <- c(x, rep(1.5, 34)); x <- c(x, rep(1.7, 31)); x <- c(x, rep(1.9, 42)); x <- c(x, rep(2.1, 12)); ci.mean(x, 1 - 0.95) </pre>	#Bai 06 <pre> ktc.tb.mau <- function(x, adj, s, alpha) { n = length(x) mean = mean(x) if (adj == F) # Chua biet DLC tong the s = sd(x) ktc.tb(mean, adj, s, n, alpha) } </pre>
#Bai 07 <pre> x <- rep(12, 2) x <- c(x, rep(12.05, 3)) x <- c(x, rep(12.10, 7)) x <- c(x, rep(12.15, 9)) x <- c(x, rep(12.20, 10)) </pre>	<pre> x <- c(x, rep(12.25, 8)) x <- c(x, rep(12.30, 6)) x <- c(x, rep(12.35, 5)) x <- c(x, rep(12.40, 3)) ktc.tb.mau(x, F, 0, 1 - 0.95) </pre>

Chương VIII: Kiểm định giả thuyết thống kê

#Bai 1

```
# x[,1] ⇔ x$profit ⇔ attach(x) profit
#a
x=read.csv(file.choose(),header=T)
hist(x[,1])

#b
a=x[,1][x[,1]>=65]
s=sd(a)
z=qt(1-0.01/2,df=length(a)-1)
e=z*s/sqrt(length(a))
print(mean(a)-e)
print(mean(a)+e)

#c
t.test(x,mu=60,conf.level = 0.99,
       alternative = "greater")
```

#Bai 01(Cường giải)

```
data1 = read.csv("Data/profit.csv",
header = T)
#data1 = read.csv(file.choose(), header
= T)

x1 = data1[,1]
hist(x1)

x2 = x1[x1 > 65]
ci.mean = function(x, alpha){
  x.mean = mean(x)
  n = length (x)
  s = sd(x)
  epsilon = qt(df = n - 1,1 -
alpha / 2) * s / sqrt(n)
  #Khi n >= 30 thì qt xap xi
  qnorm
  c(x.mean - epsilon, x.mean +
epsilon)
}
ci.mean(x2, 0.01)

#n nho dung binom.test, n lon dung
prop.test
t.test(x1, alternative = "less", mu =
60, conf.level=0.99)
#p-value nho hon 0.05 nen bac bo H0
```

Chương IX: Kiểm định hai mẫu

#Bai 1

#a)

#H1: $\mu_1 \neq \mu_2$, $\alpha = 0.05$

```
data=read.csv(file.choose(),header=T)
```

```
attach(data)
```

```
t.test(machine1,machine2, alternative = "two.side",conf.level = 1-0.05)
```

#p-value lớn hơn alpha không có căn cứ bác bỏ H_0

#d

```
test.leq.oneside = function(x,y,mu0,sigma1,sigma2,alpha)
```

```
{  
  z0=(mean(x)-mean(y))/sqrt(sigma1^2/length(x)+sigma2^2/length(y));
```

```
  p.value=pnorm(z0);
```

```
  kl=ifelse(p.value<alpha,"Bac bo H0","Khong bac bo")
```

```
  return (list(kl,p.value));
```

```
}  
test.leq.oneside(machine1, machine2, 1,0.002, 0.0025, 0.05)
```

#e

```
test.geq.oneside = function(x, y, mu0, sig1, sig2, alpha) {  
  z0 = (mean(x) - mean(y)) / sqrt( sig1 ^ 2 / length(x) + sig2 ^ 2 / length(y))  
  p.value = 1 - dnorm(z0)  
  res = ifelse(p.value < alpha, "Bac bo H0", "Khong co can cu Bac bo H0")  
  cat(res)
```

```
}  
test.geq.oneside(machine1, machine2, 1,0.002, 0.0025, 0.05)
```


1. Hàm nào sau đây sẽ cho kết quả mô phỏng 100 giá trị ngẫu nhiên có phân phối $U([-10,10])$?
 - i. `runif(100, -10, 10)`
 - ii. `runif(-10, 10, 100)`
 - iii. `runif(n=100, min = -10, max = 10)`

A. i. và iii. B. i., ii. và iii. C. i và ii. D. ii. và iii.
2. Hệ thống R chính có mặt trên _____

A. CRAN B. GNU C. CRWO D. Tất cả đều đúng
3. Hàm nào sau đây không phải là hàm được định nghĩa sẵn liên quan đến phân phối nhị thức?

A. `rbinom()` B. `qbinom()` C. `pbinom()` D. `dbinom()` E. `tbinom()`
4. Lệnh nào sau đây sẽ in ra, '1 1 2 1 2 3 1 2 3 4'

A. `for(x in 4:1) for(x in x:1) print(x)`

B. `for(x in 1:4) for(y in 1:x) print(y)`

C. `for(x in 1:16) print(1:x)`

D. `print(1:4) print(1:3)`
5. Lệnh nào sau đây **không** in ra một danh sách tất cả các số lẻ từ 1 đến 1000?

A. `for(x in seq(1^3, 10^3, 1^3)) print(x)` B. `print ((1:500)*2-1)`

C. `x <-1; while (x < 1000) print (x <- x + 2)` D. `for (x in 1:500) print(2*x-1)`
6. Hàm nào sau đây sẽ cho kết quả mô phỏng 100 giá trị ngẫu nhiên có phân phối $B(10,0,1)$
 - i. `rbinom(n = 100, size = 10, prob = 0.1)`
 - ii. `rbinom(n = 100, size = 10, prob = 1)`
 - iii. `dbinom(n = 100, size = 10, prob = 0.1)`
 - iv. `rbinom(100, 10, 0.1)`
 - v. `dbinom(100, 10, 0.1)`

A. i. B. i. và iv. C. iii. và v. D. ii.

Từ câu 7 đến câu 9 sử dụng đề bài sau

Cho kết quả chạy trong R như sau

```
One Sample t-test

data: heights

= 0.77842, df = 124, p-value = 0.4378

alternative hypothesis: true mean is not equal to 162

95 percent confidence interval:

160,6941 164,9990

sample estimates:
```

mean of x:

162.8465

7. Đây là kết quả khi thực hiện kiểm định so sánh

- A. trung bình với 1 số
- B. tỷ lệ với 1 số
- C. hay tỷ lệ với nhau
- D. hai trung bình với nhau

8. Giả thuyết và đối thuyết trong kiểm định này là:

- A. $H_0 : \mu \neq 162, H_1 : \mu = 162$
- B. $H_0 : \mu \neq 162, H_1 : \mu < 162$
- C. $H_0 : \mu = 162, H_1 : \mu \neq 162$
- D. $H_0 : \mu = 162, H_1 : \mu > 162$

9. Lệnh nào sau đây sẽ cho kết quả trên?

- A. `t.test(heights, mu = 162, conf.level = 0.95)`
- B. `t.test(data, mu = 162, conf.level = 0.90)`
- C. `t.test(data, mu = 162,8465, conf.level = 0.4378)`
- D. `t.test(heights, mu = 162,8465, conf.level = 43.78)`

10. Điều nào sau đây được dùng để tìm phương sai của mọi giá trị?

- A. `mean()`
- B. `sd()`
- C. `var()`
- D. Tất cả đáp án trên

Từ câu 11 đến câu 13 sử dụng đề bài sau:

Cho đoạn code sau trong R:

```
mu <- 2
sigma <- 2
Y <- function() rnorm(1,mu,sigma)
vecY <- function(n) replicate(n,Y())
n <- 4
MeanY <- function() mean(vecY(n))
MeanY()
SampleMeanY <- function(m) replicate(m, MeanY())
m <- 10000
hist(SampleMeanY(m), freq=0,breaks=40)
curve(dnorm(x,mu,sigma/sqrt(n)),col="blue",lty=1,lwd=2,add=TRUE)
```

11. Lệnh Y() cho kết quả

- A. Một giá trị ngẫu nhiên từ phân phối $N(2,2)$,
- B. Một giá trị ngẫu nhiên từ phân phối $U(2,2)$,

- C. Một giá trị ngẫu nhiên từ phân phối $N(0,1)$,
- D. Một giá trị ngẫu nhiên từ phân phối $N(2,2^2)$

12. Để phát sinh 10 giá trị ngẫu nhiên có phân phối chuẩn ta dùng:

- A. `vecY()^10`
- B. Gọi hàm `vecY` 10 lần
- C. `vecY(10)`
- D. `vecY(1:10)`

13. Trong đoạn code trên, lệnh

`curve(dnorm(x, mu, sigma/sqrt(n)), col="blue", lty=1, lwd=2, add=TRUE)` dùng để

- A. vẽ đồ thị histogram của trung bình mẫu
- B. vẽ đồ thị hàm phân phối xác suất của phân phối chuẩn
- C. vẽ đồ thị histogram của mẫu ngẫu nhiên các trung bình mẫu
- D. vẽ đồ thị hàm mật độ của phân phối chuẩn $N(2,1)$

14. Đáp án nào sau đây là cú pháp hợp lệ cho câu lệnh `if else` trong R?

- A. `if (<condition>){
 ## do something
}
elseif {
 ## do something else
}`
- B. `if (<condition>){
 ## do something
}
else {
 ## do something else
}`
- C. `if (<condition>){
 ## do something
} else if {
 ## do something else
}`
- D. Tất cả đáp án trên

15. Phát biểu nào sau đây là một cách khác cho `?solve`

- A. `help(solve)`
- B. `man(solve)`
- C. `hel(solve)`
- D. Tất cả đều đúng

Từ câu 16 đến câu 18 sử dụng đề bài sau:

Hàm sau đây được viết trong R

```
ci.mean <- function(x, alpha) {  
  x.bar <- mean(x)  
  s.sd <- sd(x)
```

```

n      <- length(x)
q      <- qt(1-alpha/2,n-1)
eps    <- q*s.sd/sqrt(n)
result <- c(x.bar - eps, x.bar + eps)
result
}

```

16. Mục đích của hàm này là gì?

- A. Kiểm định giả thuyết so sánh tỷ lệ với 1 số
- B. Tìm khoảng tin cậy cho tỷ lệ tổng thể
- C. Kiểm định giả thuyết so sánh trung bình với 1 số
- D. Tìm khoảng tin cậy cho trung bình tổng thể

17. Lệnh `eps <- q*s.sd/sqrt(n)` để tính

- A. dung sai (sai số) của khoảng tin cậy
- B. khoảng tin cậy
- C. độ dài của khoảng tin cậy
- D. độ tin cậy của KTC

18. Lệnh nào sau đây áp dụng đúng hàm được viết ở trên?

- A. `ci.mean(dnorm(10,5), 0.01)`
- B. `ci.mean(0.01, rnorm(5,10))`
- C. `ci.mean(pnorm(10,5), 0.03)`
- D. `ci.mean(alpha = 0.05, x = rnorm(10,5))`

19. Kết quả của code sau là gì?

```

> x <- rnorm(10)
> summary(x)

```

A.

```

[1] 22.20356 21.51156 19.52353 21.97489 21.48278 20.17869 18.09011
[8] 19.60970 21.85104 20.96596

```

B.

```

[1] 0.01874617 -0.18425254 -1.37133055 -0.59916772 0.29454513
[6] 0.38979430 -1.20807618 -0.36367602 -1.62667268 -0.25647839

```

C.

<i>Min.</i>	<i>1stQu.</i>	<i>Median</i>	<i>Mean</i>	<i>3rdQu.</i>	<i>Max.</i>
18.09	19.75	21.32	20.74	21.77	22.20

D. Tất cả các đáp án trên

20. Kết quả của code sau là gì?

```
> rpois(10,1)
```

A. [1] 0 0 1 1 2 1 1 4 1 2

B. [1] 0 8 1 1 2 1 1 4 1 2

C. [1] 7 0 1 1 2 1 1 4 1 2

D. All of the mentioned

21. Chỉ ra phát biểu đúng:

- A. NaN cũng có thể được xem như một giá trị khuyết
- B. Số Inf trình bày vô cùng trong R
- C. Giá trị NaN trình bày giá trị không được xác định
- D. Không có đáp án ở trên

Từ câu 22 đến câu 23 sử dụng đề bài sau:

Hàm sau đây được viết trong R:

```
f = function(x, mu = 0, sigma = 1){  
  1/sqrt(2*pi*sigma^2)*exp(-(x-mu)^2/(2*sigma^2))  
}  
  
integrate(function(x) f(x,0,1), lower = -Inf, upper = Inf)
```

22. Lệnh nào sau đây để tính giá trị của hàm mật độ xác suất tại $x = 3$ của phân phối $N(1, 2^2)$?

A. $f(3,1,4)$

B. $f(3,1,2)$

C. $f(3,2,1)$

D. $f(2,3,4)$

23. Hàm integrate trên dùng để

- A. kiểm tra xem hàm f có thỏa tính chất $\int_{-\infty}^{\infty} f(x)dx = 1$ hay không
- B. kiểm tra xem hàm f có sai số hay không
- C. kiểm tra xem hàm f có không âm hay không
- D. tìm hàm phân phối xác suất từ hàm mật độ xác suất f

24. Cho hàm sau đây trong R

```
p = function(k) choose(8,k)*0.3^k*0.7^(8-k)
```

Hàm p trên tính

A. $P(X \geq k)$ với $X \sim B(8, 0.7)$

B. $P(X \leq k)$ với $X \sim B(8, 0.3)$

C. $P(X = k)$ với $X \sim B(8, 0.7)$

D. $P(X = k)$ với $X \sim B(8, 0.3)$

Từ câu 25 đến câu 27 sử dụng đề bài sau

Cho đoạn code trong R sau đây

```
> t.test(score$midterm, score$final, var.equal=FALSE)
```

```
Welch Two Sample t-test
```

```
data: score$midterm and score$final
```

```
t = -0.7354, df = 35.656, p-value = 0.4669
```

alternative hypothesis: true difference in means is not equal to 0

95 percent confidence interval:

-8.858370 4.145037

sample estimates

mean of x mean of y

77.56000 79.91667

25. Hàm `t.test` ở trên dùng để kiểm định so sánh

- A. theo cặp hai trung bình (kỳ vọng) của hai tổng thể
- B. trung bình với một số
- C. hai phương sai
- D. hai trung bình (kỳ vọng) của hai tổng thể độc lập

26. Trong hàm `t.test` ở trên có dùng tham số `var.equal=FALSE`. Điều này có ý nghĩa gì?

- A. Ta giả sử phương sai của hai tổng thể khác nhau
- B. Kết quả kiểm định có thể sai
- C. Ta đang kiểm định xem phương sai của hai tổng thể có bằng nhau không
- D. Ta giả sử phương sai của hai tổng thể bằng nhau

27. Biết rằng `score$midterm` và `score$final` là điểm giữa kỳ và cuối kỳ. Kết luận của bàn toàn kiểm định ở trên là

- A. Với mức ý nghĩa 95%, ta kết luận điểm thi giữa kỳ không khác với điểm thi cuối kỳ
- B. Điểm thi giữa kỳ cao hơn điểm thi cuối kỳ
- C. Với mức ý nghĩa 5% ta kết luận điểm thi giữa kỳ khác với điểm thi cuối kỳ
- D. Không thể kết luận

28. Hàm nào sau đây sẽ cho kết quả mô phỏng 1000 giá trị ngẫu nhiên có phân phối $N(10,100)$?

- | | |
|--|--|
| A. <code>x <- rnorm(1000)</code> | B. <code>x <- rnorm(1000, 10, 10)</code> |
| C. <code>x <- rnorm(1000, 100, 10)</code> | D. <code>x <- rnorm(1000, 10, 100)</code> |

ĐÁP ÁN

1	2	3	4	5	6	7	8	9	10	11	12	13	14
A	A	E	B	A	B	A	C	A	C	D	C	D	B
15	16	17	18	19	20	21	22	23	24	25	26	27	28
A	D	A	D	C	D	A	B	A	D	D	A	D	B

Câu 1: (Bài 4)

Trong R, để “mô phỏng”, tức tạo một mẫu ngẫu nhiên có k phần tử theo phân phối đều $X \sim U(a, b)$ ta dùng lệnh: `runif(n = k, min = a, max = b)` (giá trị mặc định của `min` là 0, `max` là 1)

Câu 2: (Bài 1)

GNU (GPL) là giấy phép phần mềm tự do, không phải là... một cái gì đó mà R có thể ở trên. CRWO là cái gì không rõ.

Câu 3: (Bài 4)

Đối với một phân phối nào đó, r là để mô phỏng, q là để tìm phân vị, p là để tìm giá trị hàm phân phối, d là để tìm giá trị hàm mật độ, còn t vô nghĩa.

Câu 4:

A -> 4 3 2 1 3 2 1 2 1 1 (lưu ý kể cả khi trong vế sau là `x in x:1` nhưng nó vẫn hiểu và chạy đúng, thật kỳ diệu...)

B -> 1 1 2 1 2 3 1 2 3 4 (dễ hiểu, với mỗi x từ 1 -> 4, in lần lượt từ 1 -> x)

C -> 1 1 2 1 2 3 1 2 3 4 1 2 3 4 5 ... 1 2 3 ... 15 16 (lưu ý ở trên, mỗi số in trên một dòng, nhưng ở đây thì mỗi cụm 1 2 ... x sẽ in trên một dòng – lệnh `print` in mỗi thứ trên 1 dòng)

D -> 1 2 3 4 / 1 2 3 (thật ra nếu viết trên cùng 1 dòng thì không chạy được, phải có ; ngăn cách)

Câu 5:

A -> 1 2 3 ... 999 1000 (lệnh `seq(from = start, to = end, by = step)` in các số trong đoạn $[start, end]$ với bước nhảy là `step`)

B -> 1:500 cho vector `c(1,2,...,500)`, *2 biến nó thành `c(2,4,...,1000)`, -1 biến nó thành `c(1,3,...,999)`

C -> khởi đầu `x <- 1`, lặp lại việc cộng 2 và in ra kết quả `print(x <- x + 2)` khi x vẫn còn chưa tới 1000 (lưu ý nó sẽ cộng trước rồi in sau, nên có lẽ thầy đã có chút nhầm lẫn ở đây, nó sẽ in: 3 5 ... 1001) (ngoài ra, nếu ghi `print(x = x + 2)` sẽ bị lỗi lặp vô hạn in ra 3)

D -> dễ hiểu, với mỗi x từ 1 đến 500, in ra $2x - 1$

Câu 6: (Bài 4)

Cú pháp: `rbinom(n = k, size = n, prob = p)` (nếu không ghi tên tham số - `n`, `size`, `prob`; thì R tự động hiểu theo thứ tự đó) (`size` và `prob` không có mặc định)

Câu 7: (Bài 8)**Câu 8: (Bài 8)**

alternative hypothesis chính là đối thuyết -> is not equal là khác

Câu 9: (Bài 8)

Cú pháp: `t.test(data = X, mu = μ_0 , conf.level = 1 - α)`

Kết quả có ghi sẵn: 95 percent confidence interval (khoảng tin cậy 95%)

Câu 10: (Bài 2)**Câu 11: (Bài 5)**

Cú pháp: `rnorm(n = k, mean = mu, sd = sigma)` (mặc định mean = 0, sd = 1)

Lưu ý kí hiệu của phân phối chuẩn là $N(\mu, \sigma^2)$ (tham số sau là phương sai), nhưng trong cú pháp thì tham số sau là sd (σ) tức độ lệch chuẩn

Câu 12: (Bài 5)

Hàm `vecY(n)` sẽ cho ra n giá trị ngẫu nhiên có phân phối chuẩn.

Hàm `replicate(n, func())` sẽ thực hiện `func()` n lần, lưu các giá trị kết quả vào chung 1 vector.

Câu 13: (Bài 5)

Hàm `curve` dùng để vẽ đường cong (không phải histogram), các tham số có ý nghĩa:

`dnorm(x, mu, sigma/sqrt(n))`: hàm mật độ của phân phối chuẩn $N(\mu, \sigma^2)$ ở đây $\mu = mu = 2$ và $\sigma^2 = \left(\frac{sigma}{sqrt(n)}\right)^2 = \left(\frac{2}{2}\right)^2 = 1$

`col`: định màu (blue)

`lty`, `lwd`: định dạng (line type) và độ dày (line width) đường

`add`: vẽ thêm vào đồ thị cũ (TRUE) hay vẽ mới (FALSE)

Câu 14: (Bài 2)**Câu 15: (Bài 1)****Câu 16: (Bài 7)**

Trả về kết quả là $[\bar{x} - \varepsilon, \bar{x} + \varepsilon]$ thì cả khoảng tiêu chuẩn chứ là gì (trả về bằng cách để lệnh gọi ở cuối, hoặc `return()`, xem bài 1)

Câu 17: (Bài 7)

Công thức dung sai: $\varepsilon = q \frac{s.sd}{\sqrt{n}}$ với q là phân vị 1-alpha/2 của phân phối Student bậc tự do n - 1 (cú pháp: `qt(q, df)`). Lưu ý là bình thường với n lớn ta dùng phân phối chuẩn, nhưng bản chất của phép tính này là dùng phân phối Student, lý do dùng phân phối chuẩn là khi n lớn thì hai giá trị xấp xỉ nhau, và chuẩn thì dễ tra hơn. Do đó không cần xét các trường hợp để làm gì, dùng thẳng `qt` luôn.

Câu 18: (Bài 7)

Các hàm `dnorm` và `pnorm` trả về một (hay nhiều) giá trị hàm mật độ/phân phối, không đúng ý nghĩa mình cần (x cần là một mẫu ngẫu nhiên).

Khi gọi hàm trong R, nếu không nêu rõ tên tham số thì hàm sẽ ghép các tham số truyền vào theo thứ tự hàm được cài đặt. Nhưng nếu nêu rõ tên rồi thì không cần.

Một cách áp dụng đúng khác là: `ci.mean(rnorm(10,5), 0.05)`

Câu 19: (Bài 1)

Hàm `summary(x)` với `x` là một vector trả về các thống kê cơ bản của mẫu `x` (`min`, `median`, `mean`, `max`,...)

Câu 20: (Bài 4)

Cú pháp: `rpois(n = k, lambda = 1)` (không có mặc định)

Miễn sao nó trả về 10 phần tử nguyên không âm là được. Cái nào cũng có xác suất hết, nên không thể nói là “khó xảy ra” được.

Câu 21: (Bài 1)

Nếu chạy `is.na(NaN)` thì sẽ cho `TRUE`

Câu 22:

Như đã nói, $2^2 = 4$ ở đây là phương sai, khác với tham số thứ 2 trong hàm là độ lệch chuẩn.

Câu 23:

Hàm `intergrate` sẽ nhận một hàm theo `x` và tính tích phân từ lower đến upper của hàm đó.

Câu 24: (Bài 3)

`choose(n, k) = C_n^k`, hàm trên rõ ràng tính $P(X = k)$ của phân phối nhị thức

Câu 25: (Bài 9)

Lệnh `t.test` kiểm tra liên quan tới trung bình (`prop.test` kiểm tra tỷ lệ)

Câu 26: (Bài 9)

Tham số `var.equal` để giả sử xem phương sai (`var`) của hai mẫu giống (`TRUE`) hay khác (`FALSE`) nhau.

Câu 27: (Bài 9)

$p\text{-value}$ (p giá trị) $> \alpha$ (0.05) nên ta không đủ cơ sở bác bỏ $H_0: \mu_1 - \mu_2 = 0$

Đừng bị dụ đánh câu A, bởi vì nó là mức ý nghĩa 5% chứ không phải 95% (độ tin cậy trong khoảng tin cậy mới 95%). Câu c sai vì trái với tính toán ở trên.

Câu 28: (Bài 4)

Cú pháp: `rnorm(n = k, mean = mu, sd = sigma)` (lưu ý lần nữa: `sd` là độ lệch chuẩn)