

Notebook wee5

Bùi Khánh Duy

2023-03-23

Bài 1

Sinh ngẫu nhiên 1000 giá trị của biến X_1 biết $X_1 \sim N(0, 1)$, 1000 giá trị của X_2 biết $X_2 \sim N(0, 4)$. Vẽ ellip

```
X1 <- rnorm(1000, 0, sqrt(1)) # Sinh ra nn 1000 gt của x1
X2 <- rnorm(1000, 0, sqrt(1)) # Sinh ra nn 1000 gt của x1
```

Để vẽ ellipse -> phải cài đặt thư viện

```
# install.packages("ellipse")
library(ellipse)
```

```
##
## Attaching package: 'ellipse'

## The following object is masked from 'package:graphics':
##
##      pairs
```

Gộp X_1 và X_2 thành 1 data frame.

```
X <- data.frame(X1, X2)
head(X)
```

```
##           X1           X2
## 1 -2.6994942 -0.673359509
## 2  1.6077648  0.455655144
## 3 -0.6020787  0.478066501
## 4 -0.7487627 -0.008553207
## 5 -0.3886144 -1.050067767
## 6 -1.0116103  0.363042463
```

```
S = cov(X) # Ma trận hiệp phương sai
S
```

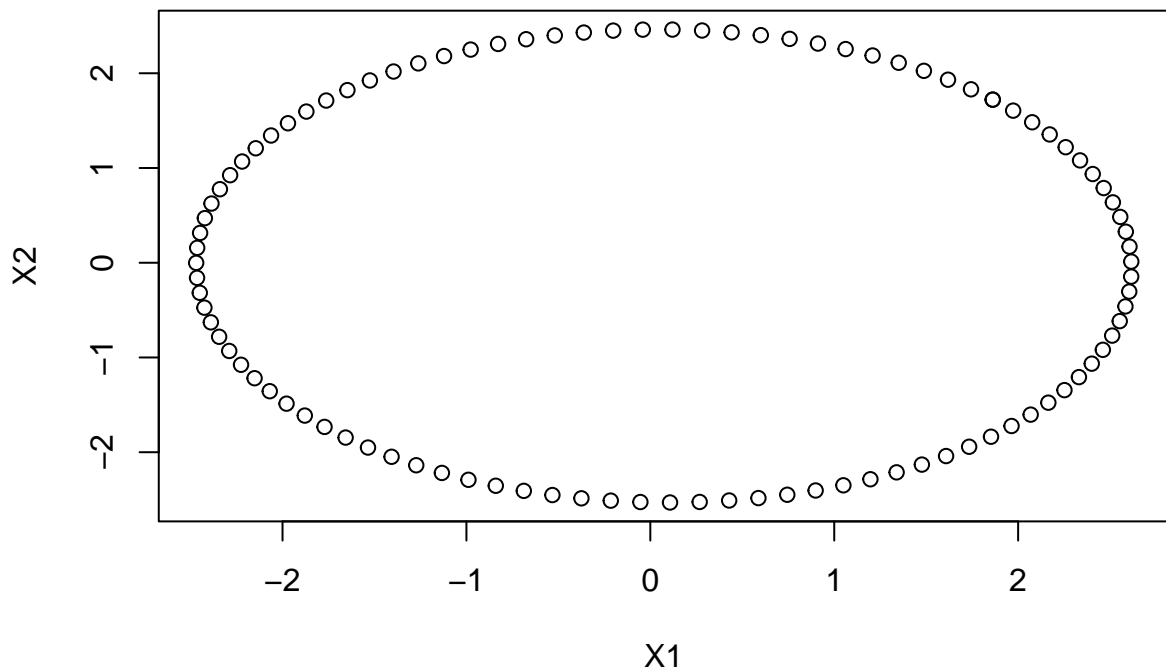
```
##           X1           X2
## X1  1.07938460 -0.01128066
## X2 -0.01128066  1.04003211
```

```
cm = colMeans(X)
cm
```

```
##           X1           X2
## 0.07340610 -0.03417673
```

Vẽ đồ thị

```
elip = ellipse(S, centre = cm)
plot(elip)
```



tâm của ellipse là kì vọng mẫu

Toạ độ

Bài 2:

Cho trước ma trận hiệp phương sai và kỳ vọng mẫu. Sinh ngẫu nhiên 1000 giá trị của vectơ ngẫu nhiên tương ứng. Kiểm định xem từng biến trong vectơ ngẫu nhiên có phân bố chuẩn 1-chiều không?

```
S = matrix(c(2,0,0,3), nrow=2)
S
```

```
##      [,1] [,2]
## [1,]    2    0
## [2,]    0    3
```

```
cm = c(0,1)
```

Dùng data có sẵn từ thư viện MASS

```
# install.packages("MASS")
library(MASS)
```

```
X = mvrnorm(1000, cm, S)
head(X)
```

```
##           [,1]      [,2]
## [1,]  1.4112726  0.1739184
## [2,]  0.7342582 -0.7384864
## [3,]  0.6468461 -0.6176199
## [4,] -1.9820850 -0.1357026
## [5,] -2.1162564  0.8190989
## [6,] -0.4197841 -0.4086836
```

```
shapiro.test(X[,1])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  X[, 1]
## W = 0.99852, p-value = 0.5701
```

Mô hình hồi quy tuyến tính

$$Y = a_0 + \varepsilon$$

$$Y = a_0 + a_i X_i + \varepsilon$$

$$Y = a_0 + a_i X_i + a_j X_j + \varepsilon$$

...

$$Y = a_0 + a_i X_i + a_j X_j + \dots + a_k X_k + \varepsilon (\leftarrow \text{Stepbackward})$$

Sử dụng chỉ số AIC để xét mô hình phù hợp. AIC nhỏ nhất \Leftrightarrow Mô hình phù hợp nhất Ví dụ ở bước 2 thứ đã thoả mãn AIC min \rightarrow dừng lại.

\Rightarrow

- Đưa ra mô hình HQT/UL của các hệ số hồi quy
- Kiểm định hệ số: $H_0 : a_1 = 0$ vs $H_1 : a_1 \neq 0$
- Tìm khoảng tin cậy cho Hệ số hồi quy
- Kiểm định phần dư $\varepsilon = y - \hat{y} \sim N(0, \sigma_2^2)$

Ví dụ

```
library("datasets")
only = lm(mpg ~ 1, data = mtcars) # Biểu diễn HQT của mpg vs chính nó
only
```

```
##
## Call:
## lm(formula = mpg ~ 1, data = mtcars)
```

```
##
## Coefficients:
## (Intercept)
##      20.09
```

```
# mpg = 20.09
two = lm(mpg ~ cyl, data=mtcars) # Biểu diễn HQT của mpg theo cyl
two
```

```
##
## Call:
## lm(formula = mpg ~ cyl, data = mtcars)
##
## Coefficients:
## (Intercept)      cyl
##      37.885      -2.876
```

```
two2 = lm(mpg ~ cyl + 0, data=mtcars) # Biểu diễn HQT của mpg theo cyl mất hệ số tự do
two2
```

```
##
## Call:
## lm(formula = mpg ~ cyl + 0, data = mtcars)
##
## Coefficients:
##      cyl
##      2.79
```

```
# mpg = 2.79 * cyl
```

```
thr = lm(mpg ~ cyl + disp, data = mtcars) # Biểu diễn HQT của mpg theo cyl và disp
thr$coefficients
```

```
## (Intercept)      cyl      disp
## 34.66099474 -1.58727681 -0.02058363
```

```
# PTHQT: mpg = 34.66099474 -1.58727681*cyl -0.02058363*disp
```

```
ff = lm(mpg ~ cyl + disp + hp, data = mtcars) # Biểu diễn HQT của mpg theo cyl, disp và hp
summary(ff)
```

```
##
## Call:
## lm(formula = mpg ~ cyl + disp + hp, data = mtcars)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -4.0889 -2.0845 -0.7745  1.3972  6.9183
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept) 34.18492    2.59078   13.195 1.54e-13 ***
## cyl         -1.22742    0.79728   -1.540  0.1349
## disp        -0.01884    0.01040   -1.811  0.0809 .
## hp          -0.01468    0.01465   -1.002  0.3250
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.055 on 28 degrees of freedom
## Multiple R-squared:  0.7679, Adjusted R-squared:  0.743
## F-statistic: 30.88 on 3 and 28 DF,  p-value: 5.054e-09
```

```
# mpg = a0 + a1*cyl + a2*disp + a3*hp + e
# Ước lượng cho các hệ số trong mô hình hồi quy:
# _h aka \hat
# a0_h = 34.18492, a1_h = -1.22742, a2_h = -0.01884, a3_h = -0.01468
# y_h[i] = mpg_h[i] = a0_h + a1_h*cyl[i] + a2_h*disp[i] + a3_h*hp[i]
# Có sẵn trong quan sát: y[i] = mpg[i]
# Phần dư e[i] = y[i] - y_h[i]
e = ff$residuals
```