TRƯỜNG ĐẠI HỌC BÁCH KHOA - ĐHQG TP.HCM
KHOA KHOA HỌC ỨNG DỤNG

XÁC SUẤT THỐNG KÊ (MT2013)

Bài tập lớn

Đề tài 4

Giảng viên: Nguyễn Đình Huy

Sinh viên: Trần Trọng Nhân - 2152209

TP. HỒ CHÍ MINH, 04 - 2023

Mục lục

Danh sách ký hiệu và các thuật ngữ viết tắt

- `ANOVA`: Phân tích phương sai (Analysis of Variance)
- `SST`: Sum of Squares Total
- `SSW`: Sum of Squares Within
- `SSG`: Sum of Squares between Groups
- `MSW`: Mean Square Within groups
- `MSG`: Mean Square between Groups

Danh sách bảng biểu

Danh sách hình ảnh

Lời cảm ơn

Đầu tiên, nhóm chúng em xin được gửi lời cảm ơn đến giảng viên Nguyễn Đình Huy vì đã hỗ trợ nhóm trong quá trình thực hiện bài tập lớn này. Nhờ sự giúp đỡ tận tình của quý thầy, chúng em đã vượt qua những khúc mắc, khó khăn trong suốt quá trình thực hiện bài tập, từ đó hoàn thành đúng tiến độ của môn học và cho ra sản phẩm chất lượng.

Ngoài ra, không thể không nhắc đến sự quan tâm giúp đỡ của các anh chị, các bạn sinh viên trong cộng đồng sinh viên trường Đại học Bách Khoa nói riêng và ĐHQG-HCM nói chung, những đóng góp to lớn của các anh, chị và các bạn đã giúp chúng em hoàn thành được bài tập lớn cho môn học Xác suất Thống kê.

Cuối cùng, nhóm chúng em xin gửi lời cảm ơn một lần nữa đến các tập thể, cá nhân đã giúp đỡ và truyền cảm hứng cho nhóm trong suốt quá trình thực hiện dự án bài tập lớn này.

1 Cơ sở lý thuyết

1.1 Giới thiệu mô hình hồi quy tuyến tính bội

Hồi quy tuyến tính là một phương pháp để dự đoán giá trị biến phụ thuộc (Y) dựa trên giá trị của biến độc lập (X). Nó có thể sử dụng cho các trường hợp chúng ta muốn dự đoán một số lượng liên tục. Ví dụ dự đoán thời gian người đọc dừng lại một trang nào đó hay số người đã truy cập vào một website,... Thông qua việc thu thập dữ liệu thực tế, chúng ta ước lượng hàm hồi quy của tổng thể, đó là ước lượng các tham số của tổng thể.

Hồi quy tuyến tính bội là phần mở rộng của hồi quy tuyến tinh đơn. Nó được sử dụng khi chúng ta muốn dự đoán giá trị của một biến phản hồi dựa trên giá trị của hai hoặc nhiều biến giải thích.

Mô hình hồi quy tuyến tính bội có dạng tổng quát như sau:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_n X_n + u \tag{1}$$

Trong đó:

- Y: Biến phụ thuộc
- X_i: Biến độc lập
- β_i: Hệ số hồi quy riêng
- β_0: Hệ số tự do (hệ số chặn)
- u: Hạng nhiễu ngẫu nhiên

1.1.1 Hàm hồi quy tổng thể (PRF - Population Regression Function)

Với Y là biến phụ thuộc và $X_1, X_2, \ldots, X_n$ là biến độc lập, Y là ngẫu nhiên và có một phân phối xác suất nào đó. Ta có:

$$F(X_1, X_2, \ldots, X_n) = E(Y \mid X_1, X_2, \ldots, X_n)$$

là hàm hồi quy tổng thể của Y theo X_1, X_2,...,X_n.

Nếu F (X) tuyến tính, ta có hàm hồi quy tổng thể có dạng tương tự phương trình (1):

$$F(X_1, X_2, \ldots, X_n) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_n X_n + u$$

1.1.2 Hàm hồi quy mẫu (SRF - Sample Regression Function)

Vì không biết tổng thể nên không biết giá trị trung bình tổng thể của biến phụ thuộc là đúng ở mức độ nào, do vậy chúng ta phải dựa vào dữ liệu mẫu để ước lượng.

Giả sử đã có các mẫu ngẫu nhiên $(Y_1, X_{1,1}, X_{2,1}, \ldots, X_{n,1})$, $(Y_2, X_{1,2}, X_{2,2}, \ldots, X_{n,2})$, ... , hàm hồi quy được xây dựng dựa trên mẫu này được gọi là **hàm hồi quy mẫu**.

Ta có hàm hồi quy mẫu tổng quát được viết dưới dạng như sau:

$$\hat{Y}_i = \hat{\beta}_1 + \hat{\beta}_2 x_{2,i} + \hat{\beta}_3 x_{3,i} + \ldots + \hat{\beta}_n x_{n,i} + \hat{u}_i$$

Trong đó $\hat{\beta}_m$ là ước lượng của β_m, $\hat{u}_i$ là ước lượng của u_i. Chúng ta mong đợi $\hat{\beta}_m$ là ước lượng không chênh lệch của β_m, hơn nữa phải là một ước lượng hiệu quả.

Ước lượng `SRF` giúp ta ước lượng các tham số của `F` qua việc tìm các tham số của $\hat{F}$ và lấy giá trị quan sát của các tham số này làm giá trị xấp xỉ cho tham số của `F`.

1.1.3 Phương pháp bình phương nhỏ nhất (Ordinary Least Squares)

Các giả thiết của phương pháp bình phương nhỏ nhất cho mô hình hồi quy tuyến tính bội như sau:

- Hàm hồi quy là tuyến tính theo các tham số.

  Điều này có nghĩa là quá trình thực hành hồi quy trên thực tế được miêu tả bởi mối quan hệ dưới dạng: $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \ldots + \beta_n x_n + u$ hoặc mối quan hệ thực tế có thể được viết lại, ví dụ như dưới dạng lấy logarit cả hai vế.

- Kỳ vọng của các yếu tố ngẫu nhiên: $u_i = 0$.

  Trung bình tổng thể sai số bằng `0`. Nghĩa là có một số giá trị sai số mang dấu dương và một số sai số mang dấu âm. Do hàm xem là đường trung bình nên giả định các sai số ngẫu nhiên trên sẽ loại trừ lẫn nhau ở mức trung bình trong tổng thể.

- $\text{Cov}(u_i, u_j) = 0$: Các sai số độc lập với nhau.

- $\text{Var}(u_i) = \sigma^2$: Các sai số có phương sai bằng nhau.

  Tất cả các giá trị u được phân phối giống nhau với cùng phương sai σ^2 sao cho $\text{Var}(u_i) = E(u_i^2) = \sigma^2$.

- Các sai số có phân phối chuẩn.

  Điều này rất quan trọng khi phát sinh khoảng tin cậy và thực hiện kiểm định giả thuyết trong những phạm vị mẫu là nhỏ. Nhưng nếu phạm vi mẫu lớn hơn, điều này trở nên không còn quan trọng.

1.1.4 Độ phù hợp của mô hình

Để có thể biết mô hình giải thích được như thế nào hay bao nhiêu % biến động của biến phụ thuộc, người ta sử dụng R^2.

Ta có:

- $\Sigma (y_i - \bar{y})^2$: `TSS` - Total Sum of Squares.
- $\Sigma (\hat{y}_i - \bar{y})^2$: `ESS` - Explained Sum of Squares.
- Σe_i^2: `RSS` - Residual Sum of Squares.

Có thể viết lại thành: `TSS = ESS + RSS`.

Ý nghĩa của các thành phần:

- `TSS` là tổng bình phương của tất cả các sai lệch giữa các giá trị quan sát Y_i và giá trị trung bình.
- `ESS` là tổng bình phương của tất cả các sai lệch giữa các giá trị của biến phụ thuộc `Y` nhận được từ hàm hồi quy mẫu và giá trị trung bình của chúng. Phần này đo độ chính xác của hàm hồi quy.
- `RSS` là tổng bình phương của tất cả các sai lệch giữa các giá trị quan sát `Y` và các giá trị nhận được từ hàm hồi quy.
- `TSS` được chia thành 2 phần: một phần do `ESS` và một phần do `RSS` gây ra.

R^2 được xác định theo công thức:

$$R^2 = \frac{ESS}{TSS} = 1 - \frac{RSS}{TSS}$$

Tỷ số giữa tổng biến thiên được giải thích bởi mô hình cho tổng bình phương cần được giải thích gọi là hệ số xác định hay là trị thống kê `good of fit`. Từ định nghĩa R^2 chúng ta thấy R^2 đo tỷ lệ hay số % của toàn bộ sai lệch `Y` với giá trị trung bình được giải thích bằng mô hình. Khi đó chúng ta sử dụng R^2 để đo sự phù hợp của hàm hồi quy:

- $0 \leq R^2 \leq 1$.
- R^2 cao nghĩa là mô hình ước lượng được giải thích được một mức độ cao biến động của biến phụ thuộc.
- Nếu $R^2 = 1$, nghĩa là đường hồi quy giải thích `100`% sự thay đổi của Y.
- Nếu $R^2 = 0$, nghĩa là mô hình không đưa ra thông tin nào về sự thay đổi của biến phụ thuộc Y.

1.1.5 Khoảng tin cậy và kiểm định các hệ số hồi quy

1.1.5.a Ước lượng khoảng tin cậy đối với các hệ số hồi quy

Với các giả thiết OLS, u_i có phân phối $N\left(0, \sigma^2\right)$. Các hệ số ước lượng tuân theo phân phối chuẩn:

$$\hat{\beta}_j \sim N\left(\hat{\beta}_j, Se\left(\hat{\beta}_j\right)\right)$$

$$\frac{\hat{\beta}_j - \beta_j}{Se\left(\hat{\beta}_j\right)} \sim T(n-k)$$

Ước lượng phương sai sai số dựa vào các phần dư bình phương tối thiểu. Trong đó k là hệ số có trong phương trình hồi quy đa biến:

$$\hat{\sigma}^2 = \frac{\sum e_i^2}{n-k}$$

Ước lượng 2 phía, tìm được $t_{\frac{\alpha}{2}}(n-k)$ thỏa mãn:

$$P\left(-t_{\frac{\alpha}{2}}(n-k)\right) \leq \frac{\hat{\beta}_j - \beta_j}{Se\left(\hat{\beta}_j\right)} \leq P\left(t_{\frac{\alpha}{2}}(n-k)\right) = 1-\alpha$$

Khoảng tin cậy $1-\alpha$ của β_j là:

$$\left[\hat{\beta}_j - t_{\frac{\alpha}{2}}(n-k).Se\left(\hat{\beta}_j\right);\ \hat{\beta}_j + t_{\frac{\alpha}{2}}(n-k).Se\left(\hat{\beta}_j\right)\right]$$

1.1.5.b Kiểm định giả thuyết đối với β_j

Kiểm định ý nghĩa thống kê của các hệ số hồi quy có ý nghĩa hay không: kiểm định rằng biến giải thích có thực sự ảnh hưởng đến biến phụ thuộc hay không. Nói cách khác là hệ số hồi quy có ý nghĩa thống kê hay không.

Có thể đưa ra giả thuyết nào đó đối với β_j, chẳng hạn $\beta_j = \beta_j^*$. Nếu giả thuyết này đúng thì:

$$T = \frac{\hat{\beta}_j - \beta_j}{Se\left(\hat{\beta}_j\right)} \sim T(n-k)$$

Ta có bảng sau:

Loại giả thuyết	Giả thuyết H_0	Giả thuyết đối H_1	Miền bác bỏ
Hai phía	$\beta_1 = \beta_i^*$	$\beta_i \neq \beta_i^*$	$\lvert t\rvert > t_{\alpha/2;\ n-k}$
Phía phải	$\beta_1 \leq \beta_i^*$	$\beta_i > \beta_i^*$	$t > t_{\alpha;\ n-k}$
Phía trái	$\beta_1 \geq \beta_i^*$	$\beta_i > \beta_i^*$	$t < -t_{\alpha;\ n-k}$

Bảng 1: Bảng tóm tắt giả thuyết và miền bác bỏ tương ứng

Ta có thể sử dụng giá trị P - value: P - value < mức ý nghĩa thì bác bỏ giả thuyết H_0.
Kiểm định β_j:

- Giả thuyết $H_0 : \beta_j = 0 \Leftrightarrow x_j$ không tác động.
- Giả thuyết $H_1 : \beta_j \neq 0 \Leftrightarrow x_j$ có tác động.
- $\beta_j < 0 \Leftrightarrow x_j$ có tác động ngược.
- $\beta_j > 0 \Leftrightarrow x_j$ có tác động thuận.

1.1.6 Kiểm định mức độ ý nghĩa chung của mô hình (trường hợp đặc biệt của kiểm định WALD)

1.1.6.a Khái quát về kiểm định WALD

Giả sử chúng ta có 2 mô hình sau:

$$(U) : Y = \beta_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_4 X_4 + u$$

$$(R) : Y = \beta_1 + \beta_2 X_2 + v$$

Mô hình U được gọi là mô hình không giới hạn (Unrestrict), và mô hình R được gọi là mô hình giới hạn (Restrict). Đó là do β_3 và β_4 buộc phải bằng 0 trong mô hình R. Ta có thể kiểm định giả thuyết liên kết $\beta_3 = \beta_4 = 0$ với giả thuyết đối là ít nhất một trong những hệ số này không bằng 0. Kiểm định giả thuyết liên kết này được gọi là kiểm định Wald, thủ tục như sau:
Đặt các mô hình giới hạn và không giới hạn là:

$$(U) : Y = \beta_1 + \beta_2 X_2 + \ldots + \beta_m X_m + \beta_{m+1} X_{m+1} + \ldots + \beta_k X_k + u$$

$$(R) : Y = \beta_1 + \beta_2 X_2 + \ldots + \beta_m X_m + v$$

Mô hình (R) có được bằng cách bỏ bớt một số biến ở mô hình (U), đó là: X_{m+1}, X_{m+2},... X_k.
Giả thuyết $H_0 : \beta_{m+1} = \ldots = \beta_k = 0$.
Giả thuyết H_1 : Các tham số không đồng thời bằng 0.

Lưu ý rằng (U) chứa k hệ số hồi quy chưa biết và (R) chứa m hệ số hồi quy chưa biết. Do đó, mô hình R có ít hơn thông số so với U. Câu hỏi chúng ta nêu ra là biến bị loại ra có ảnh hưởng ý nghĩa đối với Y hay không.

Trị thống kê kiểm định đối với giả thuyết là:

$$F_c = \frac{[RSS_R - RSS_U]/(k-m)}{RSS_U/(n-k)} \sim F(\alpha, k-m, n-k) = \frac{R_U^2 - R_R^2/(k-m)}{1 - R_U^2/(n-k)}$$

Với R^2 là số đo độ thích hợp không hiệu chỉnh.

Với giả thuyết không, F_c có phân phối F với $k-m$ bậc tự do đối với tử số và $n-k$ bậc tự do đối với mẫu số.

Ta bác bỏ giả thuyết H_0 khi:

$$F_c > F(\alpha, k-m, n-k)$$

Hoặc giá trị P - value của thống kê F nhỏ hơn mức ý nghĩa cho trước.

1.1.6.b Kiểm định ý nghĩa của mô hình

Trong mô hình hồi quy đa biến, giả thuyết "không" cho rằng mô hình không có ý nghĩa được hiểu là tất cả các hệ số hồi quy riêng đều bằng 0.

Ứng dụng kiểm định Wald (thường được gọi là kiểm định F) được tiến hành cụ thể như sau:

- Bước 1: Giả thuyết H_0: $\beta_2 = \beta_3 = \ldots = \beta_k = 0$.

  Giả thuyết H_1: Có ít nhất một trong những giá trị β khác không.

- Bước 2: Trước tiên hồi quy Y theo một số hạng không đổi và $X_2, X_3, \ldots, X_k$, sau đó tính tổng bình phương sai số RSS_U, RSS_R. Phân phối F là tỷ số của hai biến ngẫu nhiên phân phối khi bình phương độc lập. Điều này cho ta trị thống kê:

  $$F_c = \frac{[RSS_R - RSS_U]/(k-m)}{RSS_U/(n-k)} \sim F(\alpha, k-m, n-k)$$

  Vì $H_0 : \beta_2 = \beta_3 = \ldots = \beta_k = 0$, nhận thấy rằng trị thống kê kiểm định đối với giả thuyết này sẽ là:

  $$F_c = \frac{ESS/(k-1)}{RSS/(n-k)} \sim F(\alpha, k-1, n-k)$$

- Bước 3: Tra số liệu trong bảng F tương ứng với bậc tự do $(k-1)$ cho tử số và $(n-k)$ cho mẫu số, và với mức ý nghĩa α cho trước.

- Bước 4: Bác bỏ giả thuyết H_0 ở mức ý nghĩa α nếu $F_c > F(\alpha, k-1, n-k)$.

  Đối với phương pháp giá trị P - value, tính giá trị $p = P(F > F_c \mid H_0)$ và bác bỏ giả thuyết H_0 nếu p bé hơn mức ý nghĩa α.

1.2 Lý thuyết về ANOVA (Phân tích phương sai)

Mục tiêu của phân tích phương sai (Analysis of Variance - ANOVA) là so sánh trung bình của nhiều nhóm (tổng thể) dựa trên các trị trung bình của các mẫu quan sát từ các nhóm này và thông qua kiểm định giả thuyết của kết luận về sự bằng nhau của các trung bình tổng thể này.

Trong nghiên cứu, phân tích phương sai được dùng như một công cụ để xem xét ảnh hưởng của một yếu tố nguyên nhân (định tính) đến một yếu tố kết quả (định lượng). Ví dụ như khi nghiên cứu ảnh hưởng của thời gian tự học đến kết quả học tập của sinh viên. Nếu thời gian tự học của sinh viên được thu thập dạng dữ liệu định tính (dưới 9 giờ/ tuần, 9 - 18 giờ/ tuần, trên 18 giờ/ tuần); và kết quả học tập của sinh viên là dữ liệu định lượng (điểm trung bình học tập), thì phân tích phương sai là phương pháp phù hợp vì chúng ta có 3 nhóm cần so sánh trị trung bình.

Nếu chứng minh được 3 nhóm sinh viên có mức độ thời gian tự học khác nhau đều có kết quả điểm trung bình học tập bằng nhau, chúng ta kết luận được rằng ảnh hưởng của yếu tố thời gian tự học đến yếu tố kết quả học tập của những nhóm sinh viên có thời gian tự học khác nhau là như nhau. Nếu qua phân tích phương sai chúng ta thấy rằng 3 nhóm sinh viên có kết quả điểm trung bình khác nhau, trong đó nhóm có thời gian tự học nhiều (trên 18 giờ/ tuần) có kết quả học tập cao hơn 2 nhóm kia một cách có ý nghĩa thống kê, thì kết luận rút ra là thời gian tự học khác nhau sẽ có ảnh hưởng đến kết quả học tập.

Trong phần này chúng ta đề cập đến mô hình phân tích phương sai một yếu tố. Cụm từ yếu tố ở đây ám chỉ số lượng yếu tố nguyên nhân ảnh hưởng đến yếu tố kết quả đang nghiên cứu. Với ví dụ vừa nêu trên ta có một yếu tố nguyên nhân là thời gian tự học ảnh hưởng đến yếu tố kết quả học tập nên ta có loại phân tích phương sai một yếu tố.

1.2.1 Phân tích phương sai một yếu tố

Phân tích phương sai một yếu tố (`One way ANOVA`) là phân tích ảnh hưởng của một yếu tố nguyên nhân (dạng biến định tính) ảnh hưởng đến một yếu tố kết quả (dạng biến định lượng) đang nghiên cứu.

Ví dụ như xem xét ảnh hưởng của thời gian tự học của sinh viên đến kết quả học tập. Như đã phân tích ở trên, căn cứ vào thời gian tự học ta có `3` nhóm sinh viên cần so sánh về điểm trung bình học tập là nhóm dưới `9` giờ/ tuần, nhóm `9 - 18` giờ/ tuần, và nhóm trên `18` giờ/ tuần, cả `3` nhóm này thể hiện các cấp độ của một yếu tố đó là yếu tố thời gian tự học.

Xét rộng ra, `3` nhóm sinh viên này như mẫu đại diện của `3` tổng thể sinh viên với thời gian tự học khác nhau, mục đích của chúng ta là tìm hiểu xem điểm trung bình học tập của `3` tổng thể này thực ra giống hay khác nhau để kết luận liệu có hay không sự ảnh hưởng của yếu tố thời gian tự học đến kết quả học tập của sinh viên.

1.2.1.a Trường hợp k tổng thể có phân phối bình thường và phương sai bằng nhau

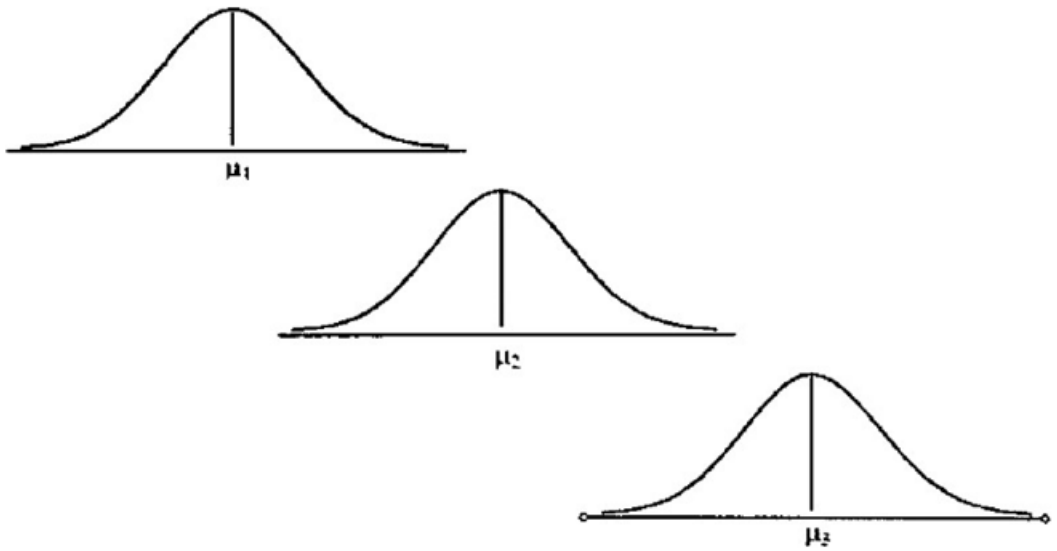
Giả sử rằng chúng ta muốn so sánh trung bình của `k` tổng thể (với ví dụ trên thì `k = 3`) dựa trên những mẫu ngẫu nhiên độc lập gồm $n_1, n_2, n_3, \ldots, n_k$ quan sát từ `k` tổng thể. Cần ghi nhớ ba giả định sau đây về các nhóm tổng thể được tiến hành phân tích `ANOVA`.

- Các tổng thể này có phân phối chuẩn.
- Các phương sai tổng thể bằng nhau.
- Các quan sát được lấy mẫu là độc lập nhau.

Nếu trung bình của các tổng thể được ký hiệu là μ_1, μ_2, ..., μ_k thì khi các giả định trên được đáp ứng, mô hình phân tích phương sai một yếu tố ảnh hưởng được mô tả dưới dạng kiểm định giả thuyết như sau: $H_0 = \mu_1 = \mu_2 = \mu_k$.

Giả thuyết H_0 cho rằng trung bình của `k` tổng thể đều bằng nhau (về mặt nghiên cứu liên hệ thì giả thuyết này cho rằng yếu tố nguyên nhân không có tác động gì đến vấn đề ta đang nghiên cứu). Và giả thuyết đối là H_1: Tồn tại ít nhất một cặp trung bình tổng thể khác nhau.

Hai giả định đầu tiên để tiến hành phân tích phương sai được mô tả như hình dưới đây, ta có thể thấy thấy ba tổng thể đều có phân phối chuẩn với mức độ phân tán tương đối giống nhau, nhưng ba vị trí chênh lệch của chúng cho thấy ba trị trung bình khác nhau. Rõ ràng nếu ta thực sự có các giá trị của `3` tổng thể và biểu diễn được phân phối của chúng như hình dưới thì ta có thể ngay lập tức kết luận bác bỏ H_0, hay `3` tổng thể này có trị trung bình khác nhau.



Hình 1: Mô hình phân phối của các tổng thể

Tuy vậy, ta chỉ có mẫu đại diện được quan sát, nên để kiểm định giả thuyết này, ta cần thực hiện các bước như sau:

Bước 1: Tính các trung bình mẫu của các nhóm (xem như đại diện của các tổng thể).

Trước hết ta xem cách tính các trung bình mẫu từ những quan sát của k mẫu ngẫu nhiên độc lập (ký hiệu $\overline{x_1}, \overline{x_2}, \ldots$) và trung bình chung của k mẫu quan sát (ký hiệu $\overline{x}$) từ trường hợp tổng quát như sau:

Tổng thể				
1	2	3	...	k
x_{11}	x_{21}	x_{31}	...	x_{k1}
x_{12}	x_{22}	x_{32}	...	x_{k2}
...	...	...	...	...

Bảng 2: Bảng số liệu tổng quát thực hiện phân tích phương sai

Tính trung bình mẫu của từng nhóm $\overline{x_1}, \overline{x_2}$ theo công thức:

$$\overline{x_{\mathrm{i}}} = \frac{\sum_{\mathrm{j}=1}^{\mathrm{n_i}} x_{\mathrm{ij}}, (\mathrm{i} = 1, 2, 3, \ldots, \mathrm{k})}{\mathrm{n_i}}$$

Và trung bình chung của k mẫu (trung bình chung của toàn bộ mẫu khảo sát):

$$\overline{x_{\mathrm{i}}} = \frac{\sum_{\mathrm{i}=1}^{\mathrm{k}} \mathrm{n_i}\overline{x_1}}{\sum_{\mathrm{i}=1}^{\mathrm{k}} \mathrm{n_i}}$$

Bước 2: Tính các tổng các chênh lệch bình phương (hay gọi tắt là tổng bình phương). Tính tổng các chênh lệch bình phương trong nội bộ nhóm - SSW và tổng các chênh lệch bình phương giữa các nhóm - SSG.

Tổng các chênh lệch bình phương trong nội bộ nhóm (SSW) được tính bằng cách cộng các chênh lệch bình phương giữa các giá trị quan sát với trung bình mẫu của từng nhóm, rồi sau đó lại tính tổng cộng kết quả tất cả các nhóm lại. SSW phản ánh phần biến thiên của yếu tố kết quả do ảnh hưởng của các yếu tố khác, chứ không phải do yếu tố nguyên nhân đang nghiên cứu (là yếu tố dùng để phân biệt các tổng thể/ nhóm đang so sánh)

Viết tổng quát theo công thức ta có:

$$\mathtt{SSW} = \sum_{\mathrm{i}=1}^{\mathrm{k}} \sum_{\mathrm{j}=1}^{\mathrm{n_i}} (x_{\mathrm{ij}} - \overline{x_{\mathrm{i}}})^2$$

Tổng các chênh lệch bình phương giữa các nhóm (SSG) được tính bằng cách cộng các chênh lệch được lấy bình phương giữa các trung bình mẫu của từng nhóm với trung bình chung của k nhóm (các chênh lệch này đều được nhân thêm với số quan sát tương ứng với từng nhóm). SSG phản ánh phần biến thiên của yếu tố kết quả do ảnh hưởng của yếu tố nguyên nhân đang nghiên cứu.

$$\mathtt{SSG} = \sum_{\mathrm{i}=1}^{\mathrm{k}} \mathrm{n_i}(\overline{x_{\mathrm{i}}} - \overline{x})^2$$

Tổng các chênh lệch bình phương toàn bộ (SST) được tính bằng cách cộng tổng các chênh lệch đã lấy bình phương giữa từng giá trị quan sát của toàn bộ mẫu nghiên cứu (x_{ij}) với trung bình toàn bộ (x). SST phản ánh biến thiên của yếu tố kết quả do ảnh hưởng của tất cả các nguyên nhân.

$$\mathtt{SST} = \sum_{i=1}^{k} \sum_{j=1}^{n} (x_{ij} - \overline{x})^2$$

Có thể dễ dàng chứng minh là tổng các chênh lệch bình phương toàn bộ bằng tổng cộng tổng các chênh lệch bình phương trong nội bộ các nhóm và tổng các chênh lệch bình phương giữa các nhóm.

$$\mathtt{SST = SSW + SSG}$$

Như vậy công thức trên cho thấy, SST là toàn bộ biến thiên của yếu tố kết quả đã được phân tích thành hai phần: phần biến thiên do yếu tố đang nghiên cứu tạo ra (SSG) và phần biến thiên còn lại do các yếu tố khác không nghiên cứu ở đây tạo ra (SSW). Nếu phần biến thiên do yếu tố nguyên nhân đang xét tạo ra càng **đáng kể** so với phần biến thiên do các yếu tố khác không cét tạo ra, thì chúng ta càng có cơ sở để bác bỏ H_0 và kết luận là yếu tố nguyên nhân đang nghiên cứu ảnh hưởng có ý nghĩa đến yếu tố kết quả.

Bước 3: Tính các phương sai (là trung bình của các chênh lệch bình phương).

Các phương sai được tính bằng cách lấy các tổng chênh lệch bình phương chia cho bậc tự do tương ứng. Tính phương sai trong nội bộ nhóm (MSW) bằng cách lấy tổng các chênh lệch bình phương trong nội bộ các nhóm (SSW) chia cho bậc tự do tương ứng là n-k (n là số quan sát, k là số nhóm so sánh). MSW là ước lượng phần biến thiên của yếu tố kết quả do các yếu tố khác gây ra.

$$\mathtt{MSW} = \frac{\mathtt{SSW}}{n-k}$$

Tính phương sai giữa các nhóm (MSG) bằng cách lấy tổng các chênh lệch bình phương giữa các nhóm chia cho bậc tự do tương ứng là k-1. MSG là ước lượng phần biến thiên của yếu tố kết quả do yếu tố nguyên nhân đang nghiên cứu gây ra.

$$\mathtt{MSG} = \frac{\mathtt{SSG}}{k-1}$$

Bước 4: Kiểm định giả thuyết. Giả thuyết về sự bằng nhau của k trung bình tổng thể được quyết đinh dựa trên tỉ số của hai phương sai: phương sai giữa các nhóm (MSG) và phương sai trong nội bộ nhóm (MSW). Tỉ số này gọi là tỉ số F vì nó tuân theo định luật Fisher - Snedecor với bậc tự do k-1 ở tử số và n-k ở mẫu số.

$$\mathtt{F} = \frac{\mathtt{MSG}}{\mathtt{MSW}}$$

Ta bác bỏ giả thuyết H_0 cho rằng trị trung bình của k tổng thể bằng nhau khi:

$$\mathtt{F} > \mathtt{F}_{(k-1;n-k;\alpha)}$$

F > $F_{(k-1;n-k;\alpha)}$ là giá trị giới hạn tra từ bảng Fisher với bậc tự do k-1 tra theo hàng đầu tiên và n-k tra theo cột đầu tiên, và cần chọn bảng với mức ý nghĩa phù hợp.

Nguồn biến thiên	Tổng chênh lệch bình phương	Bậc tự do	Phương sai	Tỉ số F
Giữa các nhóm	SSG	k-1	MSG = $\frac{SSG}{k-1}$	$\frac{MSG}{MSW}$
Trong nội bộ các nhóm	SSW	n-k	MSW = $\frac{SSW}{n-k}$	
Toàn bộ	SST	n-1		

Bảng 3: Bảng kết quả tổng quát của ANOVA

1.2.1.b Kiểm tra các giả định của phân tích phương sai

Chúng ta có thể kiểm tra nhanh các giả định này bằng đồ thị. Histogram là phương pháp tốt nhất để kiểm tra giả định về phân phối bình thường của dữ liệu nhưng nó đòi hỏi một số lượng quan sát khá lớn. Biểu đồ thân lá hay biểu đồ box and whiskers là một thay thế tốt trong tình huống số quan sát ít hơn. Nếu công cụ đồ thị cho thấy tập dữ liệu mẫu khá phù hợp với phân phối bình thường thì ta có thể xem giả định phân phối bình thường đã thỏa mãn.

Một phương pháp kiểm định tham số chắc chắn hơn cho giả định phương sai bằng nhau là kiểm định Levene về phương sai của các tổng thể. Kiểm định này xuất phát từ giả thuyết sau.

$$H_0 = {\delta_1}^2 = {\delta_2}^2 = {\delta_k}^2$$

H_1: Không phải tất cả các phương sai bằng nhau.

Để quyết định chấp nhận hay bác bỏ H_0 ta tính toán giá trị kiểm định F theo công thức:

$$F_{max} = \frac{{S_{max}}^2}{{S_{min}}^2}$$

Trong đó ${S_{max}}^2$ là phương sai lớn nhất trong các nhóm nghiên cứu và ${S_{min}}^2$ là phương sai nhỏ nhất trong các nhóm nghiên cứu. Giá trị F tính được được đem so sánh với giá trị $F_{(k;df;\alpha)}$ tra được từ bảng phân phối Hartley F_{max}.

Quy tắc quyết định: $F_{(k;df;\alpha)}$ thì bác bỏ giả thuyết H_0 cho rằng phương sai bằng nhau và ngược lại.

Nếu ta không chắc chắn về các giả định hoặc nếu kết quả kiểm định cho thấy các giả định không được thỏa mãn thì một phương pháp kiểm định thay thế cho ANOVA là phương pháp kiểm định phi tham số Kruskal - Wallis sẽ được áp dụng.

1.2.1.c Phân tích sau ANOVA

Mục đích của phân tích phương sai là kiểm định giả thuyết H_0 rằng trung bình của tổng thể bằng nhau. Sau khi phân tích và kết luận, có hai trường hợp xảy ra là chấp thuận giả thuyết H_0 hoặc bác bỏ giả thuyết H_0.

Nếu chấp nhận giả thuyết H_0 thì phân tích kết thúc.

Nếu bác bỏ giả thuyết H_0, ta kết luận trung bình của các tổng thể không bằng nhau. Vì vậy, vấn đề tiếp theo là phân tích sâu hơn để xác minh nhóm (tổng thể) nào khác nhóm nào, nhóm nào có trung bình lớn hơn hay nhỏ hơn.

Có nhiều phương pháp để tiếp tục phân tích sâu `ANOVA` khi bác bỏ giả thuyết H_0. Trong phần này chỉ đề cập đến một phương pháp thông dụng đó là phương pháp `Tukey`, phương pháp này còn được gọi là kiểm định `HSD` (`Honestly Significant Differences`). Nội dung của phương pháp này là so sánh từng cặp các trung bình nhóm ở mức ý nghĩa nào đó cho tất cả các cặp kiểm định có thể để phát hiện ra những nhóm khác nhau. Nếu có `k` nhóm nghiên cứu và chúng ta so sánh tất cả các cặp nhóm thì số lượng cặp cần phải so sánh là tổ hợp chập `2` của `k` nhóm.

$$c_k^2 = \frac{k!}{2!(k-2)!} = \frac{k(k-1)}{2}$$

Giá trị giới hạn `Tukey` được tính theo công thức:

$$T = q_{\alpha,k,n-k}\sqrt{\frac{MSW}{n_i}}$$

Trong đó:

- $q_{\alpha,k,n-k}$ là giá trị tra bảng phân phối kiểm định `Tukey` ở mức ý nghĩa α, với bậc tự do `k` và `n-k`, với `n` là tổng số quan sát mẫu ($n = \sum n_i$).
- `MSW` là phương sai trong nội bộ nhóm.
- n_i là số quan sát trong một nhóm (tổng thể), trong trường hợp mỗi nhóm có số quan sát n_i khác nhau, sử dụng giá trị n_i nhỏ nhất.

Tiêu chuẩn quyết định là bác bỏ giả thuyết H_0 khi độ lệch tuyệt đối giữa các cặp trung bình mẫu lớn hơn hay bằng `T` giới hạn.

Bên cạnh việc kiểm định để phát hiện ra những nhóm khác biệt, ta có thể tìm khoảng ước lượng cho chênh lệch giữa các nhóm có khác biệt có ý nghĩa thống kê. Ước lượng khoảng chênh lệch giữa hai trung bình nhóm có khác biệt tính theo công thức:

$$\mu_1 - \mu_2 = (\overline{x_1} - \overline{x_2} \pm (t_{n-k}, \frac{\alpha}{2}\sqrt{\frac{2 * MSW}{n_i}})$$

Trong đó, `t` là giá trị được tra từ bảng phân phối `Student t` với `n-k` bậc tự do.

Phân tích phương sai với kiểm định `F` chỉ có thể áp dụng khi các nhóm so sánh có phân phối bình thường và phương sai bằng nhau. Trong trường hợp không thỏa điều kiện này, ta có thể chuyển đổi dữ liệu của yếu tố kết quả từ dạng định lượng về dạng định tính (dữ liệu thứ bậc) và áp dụng một kiểm định phi tham số phù hợp tên là `Kruskal - Wallis`.

1.2.1.d Phương pháp phân tích phương sai một yếu tố Kruskal - Wallis bằng thứ hạng

Khi các nhóm so sánh không thỏa mãn các điều kiện có phân phối chuẩn và phương sai bằng nhau, ta không thể sử dụng phương pháp `ANOVA` thông thường cũng như phương pháp `Tukey`. Ở đây ta sẽ đề xuất một phương pháp thay thế là phương pháp `Kruskal - Wallis`.

Trong phương pháp `Kruskal - Wallis`, mỗi quan sát trong tổng số `N` quan sát được thay thế bởi một số điểm để xếp hạng, với điểm thấp nhất có hạng `1`, ..., điểm cao nhất có hạng `N`. Tổng của hạng sẽ được tính cho từng nhóm. Phương pháp `Kruskal - Wallis` giúp xác định rằng liệu tổng này có khác biệt đáng kể đến mức chúng không thể được lấy từ chung một nhóm.

Ta chứng minh được rằng nếu `k` nhóm được lấy từ chung `1` tổng thể, tức là giả thiết H_0 đúng, vậy hàm `H` được định nghĩa bởi dưới đây sẽ phân phối theo chi - bình phương (chi - square) với `df = k - 1`. Ta cần lưu ý rằng kích thước của các nhóm này không được quá nhỏ.

$$\mathtt{H} = \frac{12}{\mathrm{N}(\mathrm{N}+1)} \sum_{\mathrm{j}=1}^{\mathrm{k}} \frac{\mathrm{R}_\mathrm{j}^2}{\mathrm{n}_\mathrm{j}} - 3(\mathrm{N}+1)$$

Trong đó:

- k: Số nhóm.
- n_j: Kích thước của nhóm thứ `j`.
- N: Tổng n_j, tổng kích thước các nhóm.
- R_j: Tổng hạng của nhóm thứ `j`.

Khi giá trị quan sát được của `H` lớn hơn hoặc bằng giá trị của chi - bình phương với mức ý nghĩa cho trước và giá trị quan sát của `df = k - 1`, vậy ta có thể bác bỏ giả thiết H_0.

2 Ngôn ngữ lập trình R

2.1 Giới thiệu ngôn ngữ lập trình R

R là một ngôn ngữ lập trình và môi trường phần mềm dành cho tính toán thống kê và vẽ biểu đồ.

Trong nhiều năm trước đây, khi nhắc đến thống kê, người ta thường nghĩ ngay đến SAS, SPSS, Stata, Statistica, và S-Plus. Chúng đều là các công cụ chuyên nghiệp và có khả năng tính toán mạnh mẽ, tuy nhiên lại rất đắt tiền, có khi chi phí đến hàng trăm nghìn USD một năm. Do đó, vào năm 1997, R được phát triển bởi hai nhà thống kê học Ross Ihaka và Robert Gentleman. R có mã nguồn mở và hoàn toàn miễn phí. Từ khi ra đời đến nay, R ngày càng hoàn thiện và dần trở thành một trong những công cụ có thể sánh ngang với các ngôn ngữ tính toán thống kê khác.

2.2 Phân tích số liệu và biểu đồ trong R

2.2.1 Các cấu trúc dữ liệu cơ bản

Vector

Vector đơn giản là một danh sách các phần tử cùng loại. Cách đơn giản để tạo một vector là sử dụng hàm `c()` để gộp các thành phần riêng lẻ vào thành một vector.

```
fruits <- c("banana", "apple", "orange")
fruits
```

OUTPUT

```
> fruits
[1] "banana" "apple" "orange"
```

Muốn truy cập phần tử của vector, ta chỉ ra vị trí của chúng trong `[]`. Lưu ý, trong ngôn ngữ R, chỉ số được bắt đầu từ `1`(`1 - indexed`).

```
fruits[2]
fruits[c(1,3)]
fruits[2:3]
```

OUTPUT

```
> fruits[2]
[1] "apple"

> fruits[c(1,3)]
[1] "banana" "orange"

> fruits[2:3]
[1] "apple" "orange"
```

Matrix

Ma trận cũng giống như vector, tuy nhiên vector là một chiều, còn ma trận là hai chiều với nhiều dòng và nhiều cột. Một số câu lệnh thông dụng liên quan đến ma trận trong R như sau:

- `rbind()` : Ghép các vector hàng lại với nhau.
- `cbind()` : Ghép các vector cột lại theo với nhau.
- `nrow(),ncol()` : Cho biết số hàng, số cột của ma trận.
- `colnames(), rownames()` : Đặt tên cho các cột, hàng của ma trận.

Để chọn phần tử của ma trận, ta cần chỉ số dòng và cột hoặc tên hàng và cột tương ứng. Cho dữ liệu như sau:

	1	2
1	banana	apple
2	pineapple	melon

```
fruits[1,2]
fruits[1,1:2]
fruits[,2]
```

```
OUTPUT
> fruits[1,2]
[1] "apple"

> fruits[1,1:2]
     1      2
1 banana apple

> fruits[,2]
[1] "apple" "melon"
```

2.2.2 Tidyverse

Tidyverse là một gói lý tưởng cho việc phân tích dữ liệu, bao gồm các gói nhỏ khác của R.

Khi làm việc trong Tidyverse cũng như phân tích dữ liệu, `Data frame` (hay còn gọi là khung dữ liệu) là cấu trúc phổ biến mà chúng ta sẽ làm việc cùng. `Data frame` có thể chứa nhiều loại dữ liệu khác nhau, trong đó tất cả các phần tử của khung dữ liệu là các vector có độ dài bằng nhau. Khi làm việc với R, ta sẽ nhập số liệu vào R và lưu nó như một khung dữ liệu thay vì dùng các hàm cơ bản của R.

Để cài đặt một thư viện trong R, ta nhập vào console hàm `install.packages()`, chẳng hạn như `install.packages(tidyverse)`. Sau khi cài đặt xong, ta có thể sử dụng ở các lần làm việc tiếp theo với một lệnh đơn giản `library() : library(tidyverse)`.

2.2.2.a readr

Thư viện `readr` được sử dụng trong bài tập lớn lần này để nhập vào bộ dữ liệu.

```
library(readr)
dataset <- read.csv("test.csv")
```

2.2.2.b dplyr

Thư viện được thiết kế cụ thể cho việc phân tích, chỉnh sửa các dữ liệu trong data frame. Các hàm sẽ được sử dụng thường xuyên trong bài tập lớn lần này:

- `filter()`: Chọn các hàng thỏa mãn điều kiện đề ra.
- `count()`: Đếm số hàng có cùng chung dữ liệu theo điều kiện đề ra.
- `group_by()`: Tạo ra một bản sao của dữ liệu gốc, xử lý các hàm khác theo từng nhóm dữ liệu được cho trong hàm một cách riêng rẽ, cuối cùng ghép lại thành dữ liệu hoàn chỉnh.
- `distinct()`: Loại bỏ các hàng có dữ liệu giống nhau.
- `summary()`: Rút gọn dữ liệu lại thành một bản tổng hợp dữ liệu.

2.2.2.c ggplot2

Thư viện ggplot2 sẽ là công cụ chính giúp chúng ta tạo nên các biểu đồ từ các dữ liệu, từ đó giúp chúng ta mô hình dữ liệu một cách trực quan.

Các bước cơ bản tạo biểu đồ:

- Tạo một biểu đổ và định nghĩa các biến cần ánh xạ đến đồ thị:
  `ggplot(data = data_frame, aes(x = variable_1, y = variable_2))`
- Vẽ đồ thị đường:
  `ggplot(data = data_frame, aes(x = variable_1, y = variable_2)) + geom_line()`
- Gán tựa đề cho biểu đồ và các trục tọa độ:
  `ggplot(data = data_frame, aes(x = variable_1, y = variable_2)) + geom_line() + labs(title = "Title of Graph", x = "new x label", y = "new y label")`

Đồng thời ta cũng có thể sử dụng một số loại đồ thị khác:

- `geom_point()`
- `geom_line()`
- `geom_bar()`
- `geom_boxplot()`
- `geom_histogram()`

2.2.3 Toán tử ống (%>%)

Toán tử ống - pipe operator giúp chúng ta có thể viết chuỗi các xử lý với nhau, làm cho code dễ hiểu, dễ viết hơn. Ta cũng có thể hiểu `%>%` = `then`.

```
x %>% f(y) -> f(x,y)
```

2.2.4 Một số lệnh cơ bản khác

Ngoài các thư viện, các cấu trức dữ liệu như trên, ta còn có các câu lệnh cơ bản khác:

- `view()`: Biểu diễn lại dữ liệu theo kiểu bảng tính.
- `unique()`: Loại bỏ các hàng, cột bị trùng dữ liệu.
- `min(),max()`: Trả về giá trị nhỏ nhất, giá trị lớn nhất của một nhóm dữ liệu.
- `which()`: Trả về vị trí các hàng, cột thỏa mãn điều kiện đề ra.
- `duplicated()`: Xác định các phần tử bị trùng dữ liệu.
- `abs()`: Trả về giá trị tuyệt đối.
- `is.na()`: Xác định xem dữ liệu có mang giá trị `NA` hay không.
- ...

3 Hoạt động 1

3.1 Yêu cầu

Tập tin `flights.rda` cung cấp thông tin về `162049` chuyến bay đã khởi hành từ hai sân bay lớn của vùng Tây bắc Thái Bình Dương của Mỹ, `SEA` ở Seattle và `PDX` ở Portland trong năm 2014. Dữ liệu cung cấp bởi Văn phòng Thống kê Vận tải, Mỹ (https://www.transtats.bts.gov/).

Dữ liệu này được dùng để phân tích các nguyên nhân gây ra sự khởi hành trễ hoặc hoãn các chuyến bay. Chi tiết về bộ dữ liệu như sau:

Các biến chính trong bộ dữ liệu:

- `year, month, day`: Ngày khởi hành của mỗi chuyến bay.
- `carrier`: Tên của hãng hàng không, được mã hóa bằng 2 chữ cái in hoa. Ví dụ: UA = United Air Lines, AA = American Airlines, DL = Delta Airlines, v.v.
- `origin` và `dest`: Tên sân bay đi và đến. Đối với sân bay đi, ta chỉ có hai giá trị SEA (Seattle) và PDX (Portland).
- `dep_time` và `arr_time`: Thời gian cất cánh và hạ cánh (theo lịch dự kiến).
- `dep_delay` và `arr_delay`: Chênh lệch (phút) giữa thời gian cất cánh/hạ cánh thực tế với thời gian cất cánh/hạ cánh in trong vé.
- `distance`: Khoảng cách giữa hai sân bay (dặm).

Các bước thực hiện:

- Đọc dữ liệu (`Import data`): `flights.rda`.
- Làm sạch dữ liệu (`Data cleaning`): `NA` (dữ liệu khuyết).
- Làm rõ dữ liệu (`Data visualization`):
  - Chuyển đổi biến (nếu cần thiết).
  - Thống kê mô tả: Dùng thống kê mẫu và dùng đồ thị.
- `ANOVA` một nhân tố: Đánh giá sự khác biệt trong việc lệch giờ bay (`dep_delay`) giữa các hãng bay.
- Mô hình hồi quy tuyến tính: Sử dụng một mô hình hồi quy phù hợp để phân tích các yếu tố ảnh hưởng đến việc lệch giờ đến (`arr_delay`) của các chuyến bay.

3.2 Đọc dữ liệu (Import data)

```
library(tidyverse)
library(nortest)
library(car)
library(pgirmess)

rm(list = ls()) #Clear environments

load("flights.rda")
dataset <- flights[,c("year", "month", "day", "carrier", "origin", "dest",
"dep_time", "arr_time", "dep_delay", "arr_delay", "distance")]
head(dataset, 3) #Get first three lines of the dataset
```

Ta sử dụng đoạn code trên để đọc vào dữ liệu từ bộ dữ liệu `flights.rda`, lấy các cột dữ liệu chứa các biến chính, và xuất mẫu 3 dòng đầu tiên của bộ dữ liệu.

```
  year month day carrier origin dest dep_time arr_time dep_delay arr_delay distance
1 2014     1   1      AS    PDX  ANC        1      235        96        70     1542
2 2014     1   1      US    SEA  CLT        4      738        -6       -23     2279
3 2014     1   1      UA    PDX  IAH        8      548        13        -4     1825
```

Hình 2: Bộ dữ liệu được đọc từ `flights.rda`

3.3 Làm sạch dữ liệu (Data cleaning)

Kiểm tra dữ liệu khuyết trong tệp tin `dataset`.

```
apply(is.na(dataset),2,sum)
```

```
    year      month        day    carrier       dest  dep_time dep_delay
       0          0          0          0          0        857       857
arr_time arr_delay  distance
     988       1301          0
```

Hình 3: Thống kê số lượng giá trị khuyết đối với từng biến

```
apply(is.na(dataset),2,mean)
```

```
       year       month         day     carrier        dest
0.000000000 0.000000000 0.000000000 0.000000000 0.000000000
   dep_time   dep_delay    arr_time   arr_delay    distance
0.005288524 0.005288524 0.006096921 0.008028436 0.000000000
```

Hình 4: Thống kê tỷ lệ giá trị khuyết đối với từng biến

Nhận xét: Từ bảng thống kê trên ta thu được số lượng và tỉ lệ khuyết của từng biến, ta nhận thấy có nhiều giá trị khuyết tại các biến `dep_time, dep_delay, arr_time, arr_delay`. Vì tỉ lệ giá trị khuyết đối với từng biến là thấp (dưới `1%`) nên để làm sạch dữ liệu ta lựa chọn phương pháp xóa các giá trị khuyết trong bộ dữ liệu `dataset`.

```
dataset <- na.omit(dataset)
```

Ta kiểm tra số lượng và tỉ lệ dữ liệu khuyết đã xóa:

```
#So quan sat tu du lieu goc
nrow(flights)
#So quan sat sau khi lam sach
nrow(dataset)
#Thong ke so quan sat da xoa
nrow(flights)-nrow(dataset)
#Thong ke ty le quan sat da xoa
(nrow(flights)-nrow(dataset))/nrow(flights)
```

```
> nrow(dataset)
[1] 160748
> nrow(flights)
[1] 162049
> nrow(flights)-nrow(dataset)
[1] 1301
> (nrow(flights)-nrow(dataset))/nrow(flights)
[1] 0.008028436
```

Hình 5: Kiểm tra số lượng và tỉ lệ dữ liệu khuyết đã xóa

Nhận xét: Ta thấy số lượng dữ liệu đã xóa là `1301`, chiếm tỉ lệ `0.8%` so với dữ liệu ban đầu, có thể thấy việc xóa các dữ liệu có giá trị khuyết trong bộ dữ liệu `dataset` không làm ảnh hưởng nhiều đến kết quả của dữ liệu.

3.4 Làm rõ dữ liệu (Data visualization)

3.4.1 Tính các thông số thống kê đặc trưng với hai biến `dep_delay` và `arr_delay`

Ta tính các giá trị thống kê mô tả đặc trưng bao gồm kích thước mẫu, trung bình, độ lệch chuẩn, min, max, các điểm tứ phân vị của chênh lệch giữa thời gian cất/hạ cánh thực tế và thời gian cất/hạ cánh in trong vé (`dep_delay` và `arr_delay`) của từng hãng hàng không (`carrier`).

```
length = tapply(dataset$dep_delay, dataset$carrier,length)
mean = tapply(dataset$dep_delay, dataset$carrier,mean)
sd = tapply(dataset$dep_delay, dataset$carrier,sd)
min = tapply(dataset$dep_delay, dataset$carrier,min)
max = tapply(dataset$dep_delay, dataset$carrier,max)
Q1 = tapply(dataset$dep_delay, dataset$carrier,quantile,probs = 0.25)
Q2 = tapply(dataset$dep_delay, dataset$carrier,quantile,probs = 0.5)
Q3 = tapply(dataset$dep_delay, dataset$carrier,quantile,probs = 0.75)
t(data.frame(length,mean,sd,min,max,Q1,Q2,Q3))
```

	AA	AS	B6	DL	F9	HA	OO
length	7474.00000	62189.000000	3493.000000	16637.000000	2683.00000	1092.000000	18368.000000
mean	10.58269	2.746933	8.389064	4.778806	10.14983	2.589744	4.386378
sd	52.06113	20.335588	31.445321	29.249610	41.03238	47.239858	28.673300
min	-18.00000	-25.000000	-20.000000	-19.000000	-20.00000	-17.000000	-37.000000
max	1553.00000	866.000000	365.000000	886.000000	815.00000	878.000000	677.000000
Q1	-6.00000	-5.000000	-6.000000	-4.000000	-6.00000	-7.000000	-7.000000
Q2	-2.00000	-2.000000	-2.000000	-2.000000	-2.00000	-4.000000	-4.000000
Q3	7.00000	2.000000	9.000000	4.000000	11.00000	-1.000000	0.000000

	UA	US	VX	WN
length	16452.000000	5876.000000	3266.000000	23218.00000
mean	9.802516	2.739278	7.855175	13.33668
sd	33.773763	26.025057	32.859469	30.28554
min	-19.000000	-26.000000	-21.000000	-11.00000
max	580.000000	711.000000	358.000000	712.00000
Q1	-5.000000	-6.000000	-5.000000	-2.00000
Q2	-1.000000	-3.000000	-2.000000	3.00000
Q3	8.000000	1.000000	3.000000	17.00000

Hình 6: Kết quả khi tính các giá trị thông kê mô tả cho biến `dep_delay` theo từng biến `carrier`

```
length = tapply(dataset$arr_delay, dataset$carrier,length)
mean = tapply(dataset$arr_delay, dataset$carrier,mean)
sd = tapply(dataset$arr_delay, dataset$carrier,sd)
min = tapply(dataset$arr_delay, dataset$carrier,min)
max = tapply(dataset$arr_delay, dataset$carrier,max)
Q1 = tapply(dataset$arr_delay, dataset$carrier,quantile,probs = 0.25)
Q2 = tapply(dataset$arr_delay, dataset$carrier,quantile,probs = 0.5)
Q3 = tapply(dataset$arr_delay, dataset$carrier,quantile,probs = 0.75)
t(data.frame(length,mean,sd,min,max,Q1,Q2,Q3))
```

	AA	AS	B6	DL	F9	HA	OO
length	7474.000000	6.218900e+04	3493.000000	16637.0000000	2683.000000	1092.000000	18368.000000
mean	5.780974	9.093248e-02	3.810764	-0.3414077	9.300037	2.059524	2.786749
sd	53.731487	2.379759e+01	33.824212	31.5367870	41.424466	49.734005	30.122616
min	-52.000000	-6.700000e+01	-51.000000	-62.0000000	-35.000000	-49.000000	-40.000000
max	1539.000000	8.440000e+02	357.000000	900.0000000	804.000000	866.000000	671.000000
Q1	-14.000000	-1.200000e+01	-14.000000	-13.0000000	-8.000000	-17.000000	-11.000000
Q2	-4.000000	-4.000000e+00	-5.000000	-6.0000000	-1.000000	-5.000000	-4.000000
Q3	9.000000	6.000000e+00	9.000000	4.0000000	12.000000	10.000000	5.000000

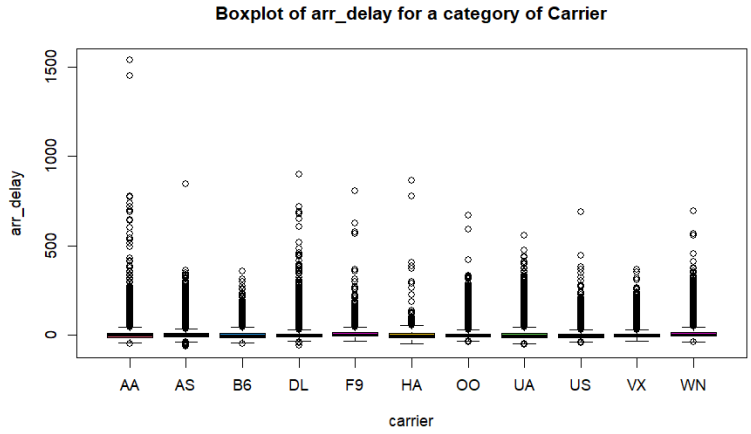
	UA	US	VX	WN
length	16452.000000	5876.000000	3266.000000	23218.000000
mean	2.377219	-1.327093	3.694121	7.836894
sd	36.426044	28.355372	35.721877	31.419566
min	-54.000000	-47.000000	-34.000000	-40.000000
max	557.000000	690.000000	366.000000	694.000000
Q1	-16.000000	-14.000000	-13.000000	-8.000000
Q2	-6.000000	-6.000000	-7.000000	0.000000
Q3	7.000000	3.000000	4.000000	13.000000

Hình 7: Kết quả khi tính các giá trị thông kê mô tả cho biến `arr_delay` theo từng biến `carrier`

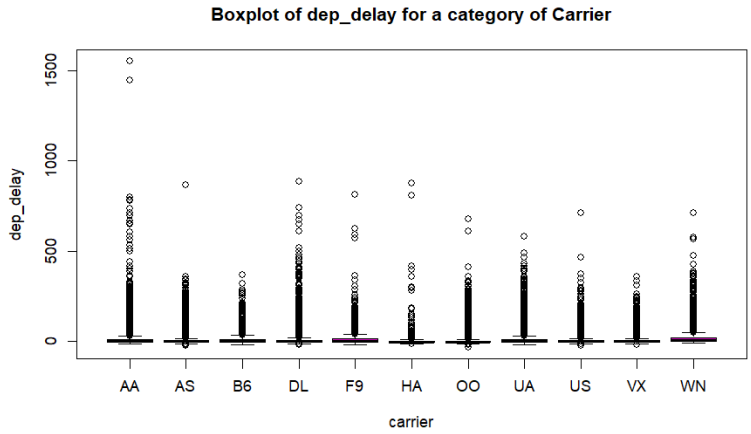
3.4.2 Vẽ đồ thị phân tán thể hiện phân phối của biến `arr_delay` và `dep_delay` theo từng hãng hàng không

Ta vẽ biểu đồ boxplot thực hiện phân phối của biến `arr_delay` và `dep_delay` theo từng hãng hàng không (`carrier`):

```
boxplot(arr_delay~carrier, xlab = "carrier", ylab = "arr_delay", main =
"Boxplot of arr_delay for each Carrier", data = dataset, col = 2:7)
boxplot(dep_delay~carrier, xlab = "carrier", ylab = "dep_delay", main =
"Boxplot of Dep_delay for each Carrier", data = dataset, col = 2:7)
```



Hình 8: Kết quả vẽ biểu đồ hộp thực hiện phân phối của biến `arr_delay` theo `carrier`



Hình 9: Kết quả vẽ biểu đồ hộp thực hiện phân phối của biến `dep_delay` theo `carrier`

Nhận xét: Qua biểu đồ trên ta thấy rằng có rất nhiều điểm ngoại lai (outliers) ở cả hai biến, điều này có thể là nguyên nhân ảnh hưởng đến kết quả phân tích phía sau. Do đó, ta sử dụng khoảng tứ phân vị (interquartile range) để loại bỏ các điểm outlier.

Ý tưởng cho bài toán: Chuyển các outliers của hai biến `arr_delay` và `dep_delay` ở từng hãng hàng không sang `NA`, từ đó đề xuất các phương pháp xử lý các `NA` đó. Ta tạo function xác định outliers, chuyển các outliers thành dạng `NA`. Để tối ưu code, ta sẽ viết function xử lý.

```
rm.out <- function(x, na.rm = TRUE, ...){
  qnt <- quantile(x, probs = c(.25,.75), na.rm = na.rm, ...)
  H <- 1.5 * IQR(x, na.rm = na.rm)
  y <- x
  y[x < (qnt[1] - H)] <- NA
  y[x > (qnt[2] + H)] <- NA
  y
}
```

Lọc các outliers tương ứng với từng hãng và chuyển thành `NA`.

```
AA = subset(dataset,dataset$carrier =="AA")
AA$dep_delay = rm.out(AA$dep_delay)
AA$arr_delay = rm.out(AA$arr_delay)
AS = subset(dataset,dataset$carrier =="AS")
AS$dep_delay = rm.out(AS$dep_delay)
AS$arr_delay = rm.out(AS$arr_delay)
B6 = subset(dataset,dataset$carrier =="B6")
B6$dep_delay = rm.out(B6$dep_delay)
B6$arr_delay = rm.out(B6$arr_delay)
DL = subset(dataset,dataset$carrier =="DL")
DL$dep_delay = rm.out(DL$dep_delay)
DL$arr_delay = rm.out(DL$arr_delay)
F9 = subset(dataset,dataset$carrier =="F9")
F9$dep_delay = rm.out(F9$dep_delay)
F9$arr_delay = rm.out(F9$arr_delay)
HA = subset(dataset,dataset$carrier =="HA")
HA$dep_delay = rm.out(HA$dep_delay)
HA$arr_delay = rm.out(HA$arr_delay)
OO = subset(dataset,dataset$carrier =="OO")
OO$dep_delay = rm.out(OO$dep_delay)
OO$arr_delay = rm.out(OO$arr_delay)
UA = subset(dataset,dataset$carrier =="UA")
UA$dep_delay = rm.out(UA$dep_delay)
UA$arr_delay = rm.out(UA$arr_delay)
US = subset(dataset,dataset$carrier =="US")
US$dep_delay = rm.out(US$dep_delay)
US$arr_delay = rm.out(US$arr_delay)
VX = subset(dataset,dataset$carrier =="VX")
VX$dep_delay = rm.out(VX$dep_delay)
VX$arr_delay = rm.out(VX$arr_delay)
WN = subset(dataset,dataset$carrier =="WN")
```

```
WN$dep_delay = rm.out(WN$dep_delay)
WN$arr_delay = rm.out(WN$arr_delay)
```

Ghép các dữ liệu với nhau và lưu vào new_dataset.

```
new_dataset <- rbind(AA,AS,B6,DL,F9,HA,OO,UA,US,VX,WN)
```

Kiểm tra lại NA trong data **new_dataset** sau khi xử lý NA.

```
apply(is.na(new_dataset),2,sum)
apply(is.na(new_dataset),2,mean)
```

```
> apply(is.na(new_dataset),2,sum)
     year     month       day   carrier    origin      dest  dep_time dep_delay  arr_time arr_delay
        0         0         0         0         0         0         0     18732         0     11520
 distance
        0

> apply(is.na(new_dataset),2,mean)
      year      month        day    carrier     origin       dest   dep_time  dep_delay   arr_time
0.00000000 0.00000000 0.00000000 0.00000000 0.00000000 0.00000000 0.00000000 0.11653022 0.00000000
 arr_delay   distance
0.07166497 0.00000000
```

Hình 10: Kết quả khi kiểm tra tổng NA và tỷ lệ NA trong tệp tin new_dataset.

Nhận xét: Với **dep_delay** số lượng NA = 18732 và chiếm tỉ lệ 11.65302% lượng quan sát của dữ liệu, với **arr_delay** số lượng NA = 11520 và chiếm tỉ lệ 7,16650% lượng quan sát của dữ liệu. Trong trường hợp này, ta không chọn phương pháp xoá các NA, vì lượng NA tương đối nhiều (>5% dữ liệu). Do đó, ta sẽ xử lý bằng phương pháp thay thế các NA bằng các giá trị trung bình tương ứng với từng hãng hàng không.

```
AA$dep_delay[is.na(AA$dep_delay)] = mean(AA$dep_delay, na.rm = T)
AS$dep_delay[is.na(AS$dep_delay)] = mean(AS$dep_delay, na.rm = T)
B6$dep_delay[is.na(B6$dep_delay)] = mean(B6$dep_delay, na.rm = T)
DL$dep_delay[is.na(DL$dep_delay)] = mean(DL$dep_delay, na.rm = T)
F9$dep_delay[is.na(F9$dep_delay)] = mean(F9$dep_delay, na.rm = T)
HA$dep_delay[is.na(HA$dep_delay)] = mean(HA$dep_delay, na.rm = T)
OO$dep_delay[is.na(OO$dep_delay)] = mean(OO$dep_delay, na.rm = T)
UA$dep_delay[is.na(UA$dep_delay)] = mean(UA$dep_delay, na.rm = T)
US$dep_delay[is.na(US$dep_delay)] = mean(US$dep_delay, na.rm = T)
VX$dep_delay[is.na(VX$dep_delay)] = mean(VX$dep_delay, na.rm = T)
WN$dep_delay[is.na(WN$dep_delay)] = mean(WN$dep_delay, na.rm = T)
AA$arr_delay[is.na(AA$arr_delay)] = mean(AA$arr_delay, na.rm = T)
AS$arr_delay[is.na(AS$arr_delay)] = mean(AS$arr_delay, na.rm = T)
B6$arr_delay[is.na(B6$arr_delay)] = mean(B6$arr_delay, na.rm = T)
DL$arr_delay[is.na(DL$arr_delay)] = mean(DL$arr_delay, na.rm = T)
F9$arr_delay[is.na(F9$arr_delay)] = mean(F9$arr_delay, na.rm = T)
HA$arr_delay[is.na(HA$arr_delay)] = mean(HA$arr_delay, na.rm = T)
OO$arr_delay[is.na(OO$arr_delay)] = mean(OO$arr_delay, na.rm = T)
```

```
UA$arr_delay[is.na(UA$arr_delay)] = mean(UA$arr_delay, na.rm = T)
US$arr_delay[is.na(US$arr_delay)] = mean(US$arr_delay, na.rm = T)
VX$arr_delay[is.na(VX$arr_delay)] = mean(VX$arr_delay, na.rm = T)
WN$arr_delay[is.na(WN$arr_delay)] = mean(WN$arr_delay, na.rm = T)
```

Ghép các dữ liệu lại với nhau và kiểm tra lại `NA` trong data `new_dataset` sau khi xử lý `NA`.

```
apply(is.na(new_dataset), 2, which)
```

```
integer(0)
```

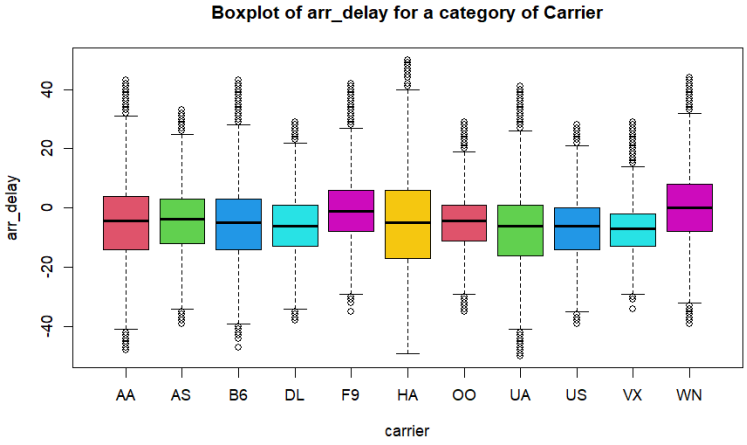
Hình 11: Kết quả kiểm tra lại `NA` trong data `new_dataset` sau khi xử lý `NA`

Nhận xét: Sau khi thay thế các `NA` bằng các giá trị trung bình, dữ liệu đã không còn `NA`. Ta tính lại các giá trị mô tả thống kê và vẽ lại biểu đồ boxplot cho hai biến theo `carrier`:

	AA	AS	B6	DL	F9	HA	OO
length	7474.000000	62189.000000	3493.000000	16637.000000	2683.0000000	1092.000000	18368.000000
mean	-4.237595	-3.808254	-3.947745	-5.401035	0.1303813	-4.201528	-4.318732
sd	15.151738	12.410553	14.915370	11.819556	12.6086115	17.787020	10.300434
min	-48.000000	-39.000000	-47.000000	-38.000000	-35.0000000	-49.000000	-35.000000
max	43.000000	33.000000	43.000000	29.000000	42.0000000	50.000000	29.000000
Q1	-14.000000	-12.000000	-14.000000	-13.000000	-8.0000000	-17.000000	-11.000000
Q2	-4.237595	-3.808254	-5.000000	-6.000000	-1.0000000	-5.000000	-4.318732
Q3	4.000000	3.000000	3.000000	1.000000	6.0000000	6.000000	1.000000

	UA	US	VX	WN
length	16452.000000	5876.00000	3266.000000	23218.000000
mean	-6.000663	-6.12416	-6.165354	1.107677
sd	14.641179	11.71164	10.277663	13.931702
min	-50.000000	-39.00000	-34.000000	-39.000000
max	41.000000	28.00000	29.000000	44.000000
Q1	-16.000000	-14.00000	-13.000000	-8.000000
Q2	-6.000663	-6.12416	-7.000000	0.000000
Q3	1.000000	0.00000	-2.000000	8.000000

Hình 12: Kết quả khi tính lại các giá trị thống kê mô tả cho biến `arr_delay` của từng `carrier`.



Hình 13: Kết quả khi vẽ lại biểu đồ hộp thực hiện phân phối biến `arr_delay` của từng `carrier`.

Nhận xét:
Hãng hàng không AA:

- Min = -48: Thời gian hạ cánh sớm nhất: sớm hơn 48 phút so với thời điểm dự kiến.
- Max = 43: Thời gian hạ cánh trễ nhất: trễ hơn 43 phút so với thời điểm dự kiến.
- Q1 = -14: 25% chuyến bay có thời gian hạ cánh sớm hơn ít nhất 14 phút so với thời điểm dự kiến.
- Q2 = -4: 50% chuyến bay có thời gian hạ cánh sớm hơn ít nhất 4 phút so với thời điểm dự kiến.
- Q3 = 4: 75% chuyến bay có thời gian hạ cánh trễ hơn nhiều nhất 4 phút so với thời điểm dự kiến.

Hãng hàng không AS:

- Min = -39: Thời gian hạ cánh sớm nhất: sớm hơn 39 phút so với thời điểm dự kiến.
- Max = 33: Thời gian hạ cánh trễ nhất: trễ hơn 33 phút so với thời điểm dự kiến.
- Q1 = -12: 25% chuyến bay có thời gian hạ cánh sớm hơn ít nhất 12 phút so với thời điểm dự kiến.
- Q2 = -3.808254: 50% chuyến bay có thời gian hạ cánh sớm hơn ít nhất 3.808254 phút so với thời điểm dự kiến.
- Q3 = 3: 75% chuyến bay có thời gian hạ cánh trễ hơn nhiều nhất 3 phút so với thời điểm dự kiến.

Hãng hàng không B6:

- Min = -47: Thời gian hạ cánh sớm nhất: sớm hơn 47 phút so với thời điểm dự kiến.
- Max = 43: Thời gian hạ cánh trễ nhất: trễ hơn 43 phút so với thời điểm dự kiến.
- Q1 = -14: 25% chuyến bay có thời gian hạ cánh sớm hơn ít nhất 14 phút so với thời điểm dự kiến.
- Q2 = -5: 50% chuyến bay có thời gian hạ cánh sớm hơn ít nhất 5 phút so với thời điểm dự kiến.
- Q3 = 3: 75% chuyến bay có thời gian hạ cánh trễ hơn nhiều nhất 3 phút so với thời điểm dự kiến.

Hãng hàng không DL:

- Min = -38: Thời gian hạ cánh sớm nhất: sớm hơn 38 phút so với thời điểm dự kiến.
- Max = 29: Thời gian hạ cánh trễ nhất: trễ hơn 29 phút so với thời điểm dự kiến.
- Q1 = -13: 25% chuyến bay có thời gian hạ cánh sớm hơn ít nhất 13 phút so với thời điểm dự kiến.
- Q2 = -6: 50% chuyến bay có thời gian hạ cánh sớm hơn ít nhất 6 phút so với thời điểm dự kiến.
- Q3 = 1: 75% chuyến bay có thời gian hạ cánh trễ hơn nhiều nhất 1 phút so với thời điểm dự kiến.

Hãng hàng không F9:

- Min = -35: Thời gian hạ cánh sớm nhất: sớm hơn 35 phút so với thời điểm dự kiến.
- Max = 42: Thời gian hạ cánh trễ nhất: trễ hơn 42 phút so với thời điểm dự kiến.
- Q1 = -8: 25% chuyến bay có thời gian hạ cánh sớm hơn ít nhất 8 phút so với thời điểm dự kiến.
- Q2 = -1: 50% chuyến bay có thời gian hạ cánh sớm hơn ít nhất 1 phút so với thời điểm dự kiến.
- Q3 = 6: 75% chuyến bay có thời gian hạ cánh trễ hơn nhiều nhất 6 phút so với thời điểm dự kiến.

Hãng hàng không HA:

- Min = -49: Thời gian hạ cánh sớm nhất: sớm hơn 49 phút so với thời điểm dự kiến.
- Max = 50: Thời gian hạ cánh trễ nhất: trễ hơn 50 phút so với thời điểm dự kiến.
- Q1 = -17: 25% chuyến bay có thời gian hạ cánh sớm hơn ít nhất 17 phút so với thời điểm dự kiến.
- Q2 = -5: 50% chuyến bay có thời gian hạ cánh sớm hơn ít nhất 5 phút so với thời điểm dự kiến.
- Q3 = 6: 75% chuyến bay có thời gian hạ cánh trễ hơn nhiều nhất 6 phút so với thời điểm dự kiến.

Hãng hàng không OO:

- Min = -35: Thời gian hạ cánh sớm nhất: sớm hơn 35 phút so với thời điểm dự kiến.
- Max = 29: Thời gian hạ cánh trễ nhất: trễ hơn 29 phút so với thời điểm dự kiến.
- Q1 = -11: 25% chuyến bay có thời gian hạ cánh sớm hơn ít nhất 11 phút so với thời điểm dự kiến.
- Q2 = -4: 50% chuyến bay có thời gian hạ cánh sớm hơn ít nhất 4 phút so với thời điểm dự kiến.
- Q3 = 1: 75% chuyến bay có thời gian hạ cánh trễ hơn nhiều nhất 1 phút so với thời điểm dự kiến.

Hãng hàng không UA:

- Min = -50: Thời gian hạ cánh sớm nhất: sớm hơn 50 phút so với thời điểm dự kiến.
- Max = 41: Thời gian hạ cánh trễ nhất: trễ hơn 41 phút so với thời điểm dự kiến.
- Q1 = -16: 25% chuyến bay có thời gian hạ cánh sớm hơn ít nhất 16 phút so với thời điểm dự kiến.
- Q2 = -6: 50% chuyến bay có thời gian hạ cánh sớm hơn ít nhất 6 phút so với thời điểm dự kiến.
- Q3 = 1: 75% chuyến bay có thời gian hạ cánh trễ hơn nhiều nhất 1 phút so với thời điểm dự kiến.

Hãng hàng không US:

- Min = -39: Thời gian hạ cánh sớm nhất: sớm hơn 39 phút so với thời điểm dự kiến.
- Max = 28: Thời gian hạ cánh trễ nhất: trễ hơn 28 phút so với thời điểm dự kiến.
- Q1 = -14: 25% chuyến bay có thời gian hạ cánh sớm hơn ít nhất 14 phút so với thời điểm dự kiến.
- Q2 = -6.12416: 50% chuyến bay có thời gian hạ cánh sớm hơn ít nhất 6.12416 phút so với thời điểm dự kiến.
- Q3 = 0: 75% chuyến bay có thời gian khởi hành cùng lúc hoặc sớm hơn so với thời điểm dự kiến.

Hãng hàng không `VX`:

- `Min = -39`: Thời gian hạ cánh sớm nhất: sớm hơn `39` phút so với thời điểm dự kiến.
- `Max = 29`: Thời gian hạ cánh trễ nhất: trễ hơn `29` phút so với thời điểm dự kiến.
- `Q1 = -13`: `25%` chuyến bay có thời gian hạ cánh sớm hơn ít nhất `13` phút so với thời điểm dự kiến.
- `Q2 = -7`: `50%` chuyến bay có thời gian hạ cánh sớm hơn ít nhất `7` phút so với thời điểm dự kiến.
- `Q3 = -2`: `75%` chuyến bay có thời gian hạ cánh sớm hơn ít nhất `2` phút so với thời điểm dự kiến.

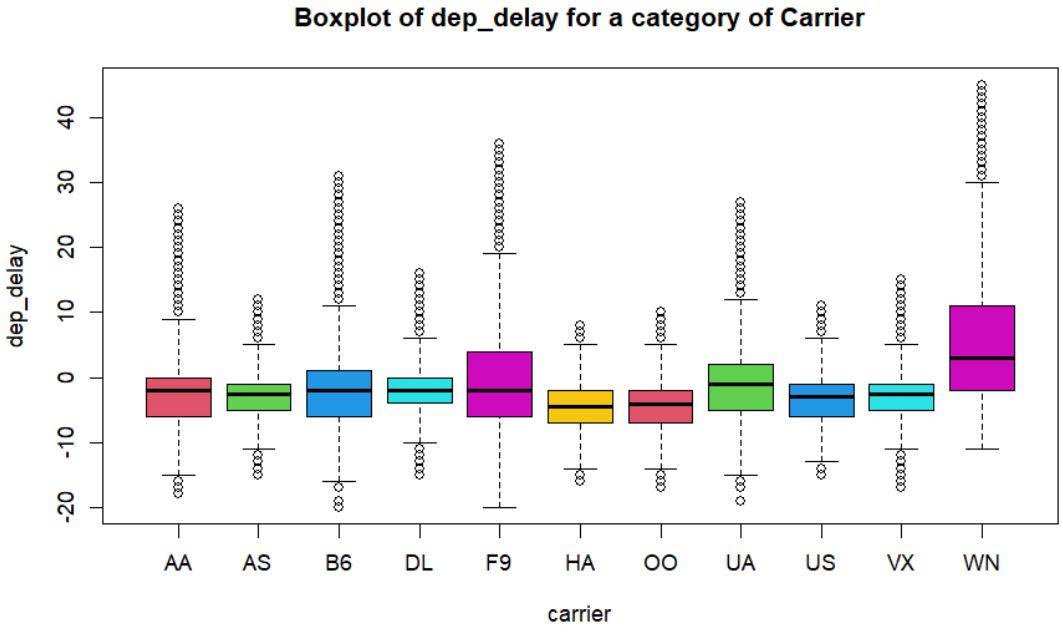
Hãng hàng không `WN`:

- `Min = -39`: Thời gian hạ cánh sớm nhất: sớm hơn `39` phút so với thời điểm dự kiến.
- `Max = 44`: Thời gian hạ cánh trễ nhất: trễ hơn `44` phút so với thời điểm dự kiến.
- `Q1 = -8`: `25%` chuyến bay có thời gian hạ cánh sớm hơn ít nhất `8` phút so với thời điểm dự kiến.
- `Q2 = 0`: `50%` chuyến bay có thời gian khởi hành cùng lúc hoặc sớm hơn so với thời điểm dự kiến.
- `Q3 = 8`: `75%` chuyến bay có thời gian hạ cánh trễ hơn nhiều nhất `8` phút so với thời điểm dự kiến.

	AA	AS	B6	DL	F9	HA	OO
length	7474.0000000	62189.000000	3493.000000	16637.000000	2683.0000000	1092.000000	18368.000000
mean	-0.9703442	-2.556315	-0.324333	-1.037923	0.5559252	-4.505550	-4.113604
sd	7.2149732	4.437033	8.807274	4.977870	9.9261578	3.957811	4.071091
min	-18.0000000	-15.000000	-20.000000	-15.000000	-20.0000000	-16.000000	-17.000000
max	26.0000000	12.000000	31.000000	16.000000	36.0000000	8.000000	10.000000
Q1	-6.0000000	-5.000000	-6.000000	-4.000000	-6.0000000	-7.000000	-7.000000
Q2	-2.0000000	-2.556315	-2.000000	-2.000000	-2.0000000	-4.505550	-4.113604
Q3	0.0000000	-1.000000	1.000000	0.000000	4.0000000	-2.000000	-2.000000

	UA	US	VX	WN
length	16452.000000	5876.000000	3266.000000	23218.000000
mean	0.140357	-3.008452	-2.627174	6.428162
sd	7.300819	4.116384	4.813692	11.337814
min	-19.000000	-15.000000	-17.000000	-11.000000
max	27.000000	11.000000	15.000000	45.000000
Q1	-5.000000	-6.000000	-5.000000	-2.000000
Q2	-1.000000	-3.008452	-2.627174	3.000000
Q3	2.000000	-1.000000	-1.000000	11.000000

Hình 14: Kết quả khi tính lại các giá trị thống kê mô tả cho biến `dep_delay` của từng `carrier`.



Hình 15: Kết quả khi vẽ lại biểu đồ hộp thực hiện phân phối biến `dep_delay` của từng `carrier`.

Nhận xét:
Hãng hàng không `AA`:

- `Min = -18`: Thời gian khởi hành sớm nhất: sớm hơn `18` phút so với thời điểm dự kiến.
- `Max = 26`: Thời gian khởi hành trễ nhất: trễ hơn `26` phút so với thời điểm dự kiến.
- `Q1 = -6`: `25%` chuyến bay có thời gian khởi hành sớm hơn ít nhất `6` phút so với thời điểm dự kiến.
- `Q2 = -2`: `50%` chuyến bay có thời gian khởi hành sớm hơn ít nhất `2` phút so với thời điểm dự kiến.
- `Q3 = 0`: `75%` chuyến bay có thời gian khởi hành cùng lúc hoặc sớm hơn so với thời điểm dự kiến.

Hãng hàng không `AS`:

- `Min = -15`: Thời gian khởi hành sớm nhất: sớm hơn `15` phút so với thời điểm dự kiến.
- `Max = 12`: Thời gian khởi hành trễ nhất: trễ hơn `12` phút so với thời điểm dự kiến.
- `Q1 = -5`: `25%` chuyến bay có thời gian khởi hành sớm hơn ít nhất `5` phút so với thời điểm dự kiến.

- `Q2 = -2.556315`: `50%` chuyến bay có thời gian khởi hành sớm hơn ít nhất `2.556315` phút so với thời điểm dự kiến.
- `Q3 = -1`: `75%` chuyến bay có thời gian khởi hành sớm hơn ít nhất `1` phút so với thời điểm dự kiến.

Hãng hàng không `B6`:

- `Min = -20`: Thời gian khởi hành sớm nhất: sớm hơn `20` phút so với thời điểm dự kiến.
- `Max = 31`: Thời gian khởi hành trễ nhất: trễ hơn `31` phút so với thời điểm dự kiến.
- `Q1 = -6`: `25%` chuyến bay có thời gian khởi hành sớm hơn ít nhất `6` phút so với thời điểm dự kiến.
- `Q2 = -2`: `50%` chuyến bay có thời gian khởi hành sớm hơn ít nhất `2` phút so với thời điểm dự kiến.
- `Q3 = 1`: `75%` chuyến bay có thời gian khởi hành trễ hơn nhiều nhất `1` phút so với thời điểm dự kiến.

Hãng hàng không `DL`:

- `Min = -15`: Thời gian khởi hành sớm nhất: sớm hơn `15` phút so với thời điểm dự kiến.
- `Max = 16`: Thời gian khởi hành trễ nhất: trễ hơn `16` phút so với thời điểm dự kiến.
- `Q1 = -4`: `25%` chuyến bay có thời gian khởi hành sớm hơn ít nhất `4` phút so với thời điểm dự kiến.
- `Q2 = -2`: `50%` chuyến bay có thời gian khởi hành sớm hơn ít nhất `2` phút so với thời điểm dự kiến.
- `Q3 = 0`: `75%` chuyến bay có thời gian khởi hành cùng lúc hoặc sớm hơn so với thời điểm dự kiến.

Hãng hàng không `F9`:

- `Min = -20`: Thời gian khởi hành sớm nhất: sớm hơn `20` phút so với thời điểm dự kiến.
- `Max = 36`: Thời gian khởi hành trễ nhất: trễ hơn `36` phút so với thời điểm dự kiến.
- `Q1 = -6`: `25%` chuyến bay có thời gian khởi hành sớm hơn ít nhất `6` phút so với thời điểm dự kiến.
- `Q2 = -2`: `50%` chuyến bay có thời gian khởi hành sớm hơn ít nhất `2` phút so với thời điểm dự kiến.
- `Q3 = 4`: `75%` chuyến bay có thời gian khởi hành trễ hơn nhiều nhất `4` phút so với thời điểm dự kiến.

Hãng hàng không HA:

- Min = -16: Thời gian khởi hành sớm nhất: sớm hơn 16 phút so với thời điểm dự kiến.
- Max = 8: Thời gian khởi hành trễ nhất: trễ hơn 8 phút so với thời điểm dự kiến.
- Q1 = -7: 25% chuyến bay có thời gian khởi hành sớm hơn ít nhất 7 phút so với thời điểm dự kiến.
- Q2 = -4.505550: 50% chuyến bay có thời gian khởi hành sớm hơn ít nhất 4.505550 phút so với thời điểm dự kiến.
- Q3 = -2: 75% chuyến bay có thời gian khởi hành sớm hơn ít nhất 2 phút so với thời điểm dự kiến.

Hãng hàng không OO:

- Min = -17: Thời gian khởi hành sớm nhất: sớm hơn 17 phút so với thời điểm dự kiến.
- Max = 10: Thời gian khởi hành trễ nhất: trễ hơn 10 phút so với thời điểm dự kiến.
- Q1 = -7: 25% chuyến bay có thời gian khởi hành sớm hơn ít nhất 7 phút so với thời điểm dự kiến.
- Q2 = -4.113604: 50% chuyến bay có thời gian khởi hành sớm hơn ít nhất 4.113604 phút so với thời điểm dự kiến.
- Q3 = -2: 75% chuyến bay có thời gian khởi hành sớm hơn ít nhất 2 phút so với thời điểm dự kiến.

Hãng hàng không UA:

- Min = -19: Thời gian khởi hành sớm nhất: sớm hơn 19 phút so với thời điểm dự kiến.
- Max = 27: Thời gian khởi hành trễ nhất: trễ hơn 27 phút so với thời điểm dự kiến.
- Q1 = -5: 25% chuyến bay có thời gian khởi hành sớm hơn ít nhất 5 phút so với thời điểm dự kiến.
- Q2 = -1: 50% chuyến bay có thời gian khởi hành sớm hơn ít nhất 1 phút so với thời điểm dự kiến.
- Q3 = 2: 75% chuyến bay có thời gian khởi hành trễ hơn nhiều nhất 2 phút so với thời điểm dự kiến.

Hãng hàng không US:

- Min = -15: Thời gian khởi hành sớm nhất: sớm hơn 15 phút so với thời điểm dự kiến.
- Max = 11: Thời gian khởi hành trễ nhất: trễ hơn 11 phút so với thời điểm dự kiến.
- Q1 = -6: 25% chuyến bay có thời gian khởi hành sớm hơn ít nhất 6 phút so với thời điểm dự kiến.
- Q2 = -3.008452: 50% chuyến bay có thời gian khởi hành sớm hơn ít nhất 3.008452 phút so với thời điểm dự kiến.
- Q3 = -1: 75% chuyến bay có thời gian khởi hành sớm hơn ít nhất 1 phút so với thời điểm dự kiến.

Hãng hàng không VX:

- Min = -17: Thời gian khởi hành sớm nhất: sớm hơn 17 phút so với thời điểm dự kiến.
- Max = 15: Thời gian khởi hành trễ nhất: trễ hơn 15 phút so với thời điểm dự kiến.
- Q1 = -5: 25% chuyến bay có thời gian khởi hành sớm hơn ít nhất 5 phút so với thời điểm dự kiến.
- Q2 = -2.627174: 50% chuyến bay có thời gian khởi hành sớm hơn ít nhất 2.627174 phút so với thời điểm dự kiến.
- Q3 = -1: 75% chuyến bay có thời gian khởi hành sớm hơn ít nhất 1 phút so với thời điểm dự kiến.

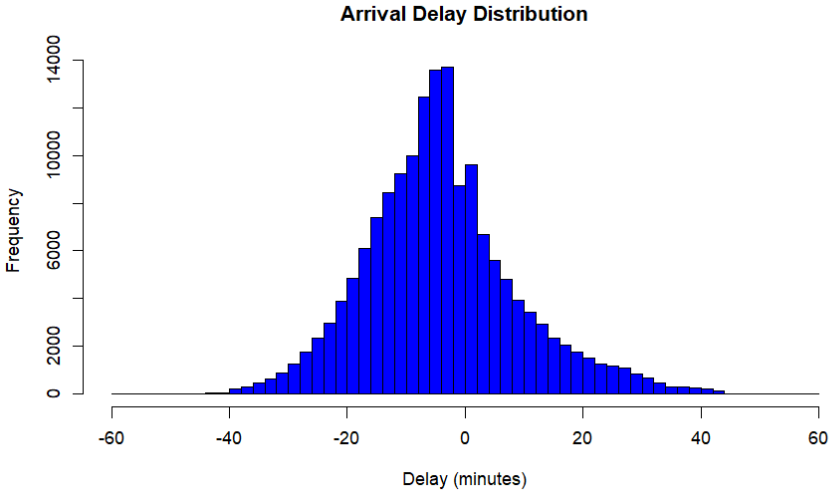
Hãng hàng không WN:

- Min = -11: Thời gian khởi hành sớm nhất: sớm hơn 11 phút so với thời điểm dự kiến.
- Max = 45: Thời gian khởi hành trễ nhất: trễ hơn 45 phút so với thời điểm dự kiến.
- Q1 = -2: 25% chuyến bay có thời gian khởi hành sớm hơn ít nhất 2 phút so với thời điểm dự kiến.
- Q2 = 3: 50% chuyến bay có thời gian khởi hành trễ hơn nhiều nhất 3 phút so với thời điểm dự kiến.
- Q3 = 11: 75% chuyến bay có thời gian khởi hành trễ hơn nhiều nhất 11 phút so với thời điểm dự kiến.

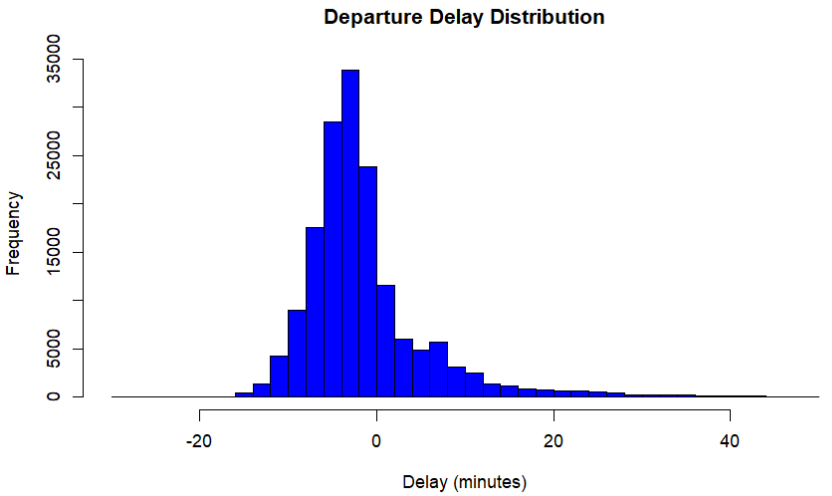
Kết luận: *Dựa vào kết quả trên, nhìn chung có sự khác biệt về phân phối của thời gian lệch giờ bay ở các hãng hàng không. Từ kết quả trên, ta có thể dự đoán được hãng WN có thời gian khởi hành trễ nhất (so với thời gian dự kiến).*

3.4.3 Đồ thị Histogram cho dep_delay và arr_delay

```
hist(new_dataset$arr_delay, main = "Arrival Delay Distribution", xlab = "Delay (minutes)",
ylab = "Frequency", breaks = seq(-60, 60, by = 2), col = "blue")
hist(new_dataset$dep_delay, main = "Departure Delay Distribution", xlab = "Delay (minutes)",
ylab = "Frequency", breaks = seq(-30, 50, by = 2), col = "blue")
```



Hình 16: Đồ thị Histogram của arr_delay

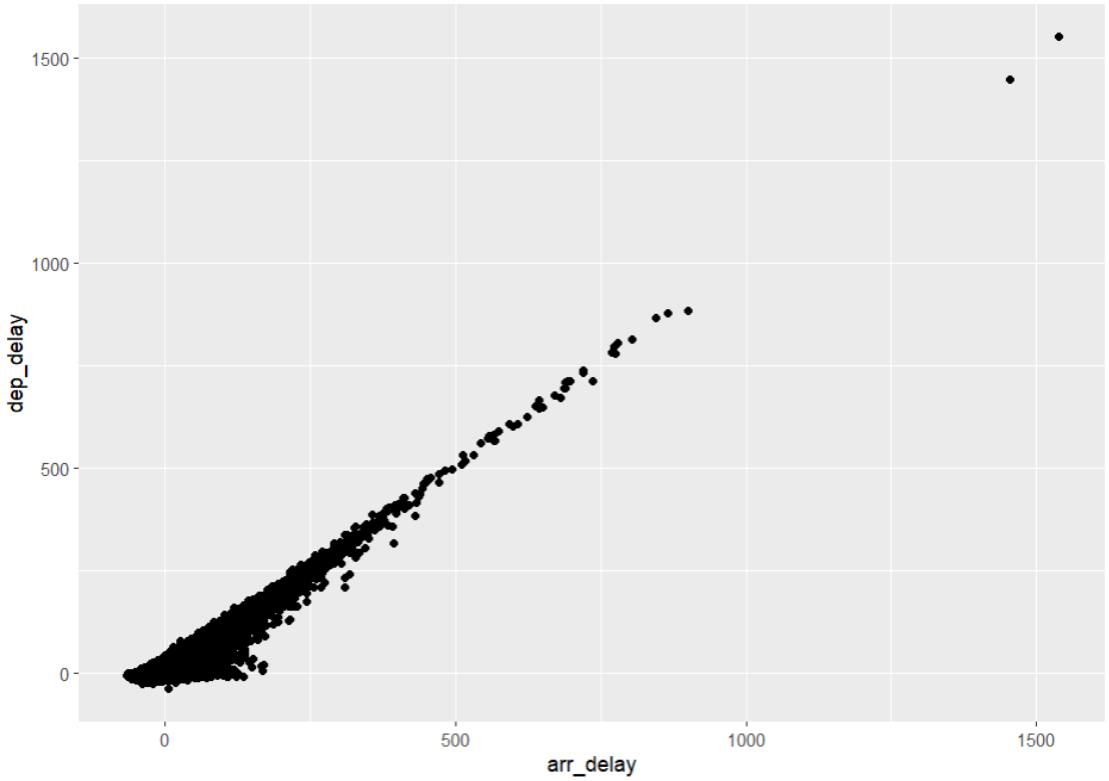


Hình 17: Đồ thị Histogram của dep_delay

Nhận xét: Biểu đồ cho `arr_delay` có dạng phân phối chuẩn, tập trung phần lớn từ `-10` đến `0` phút. Trong khi đó, biểu đồ cho `dep_delay` cũng có dạng phân phối chuẩn tuy nhiên lại lệch về phía trái, tập trung phần lớn từ `-10` đến `0` phút. Có thể thấy phần lớn máy bay sẽ khởi hành và hạ cánh sớm hơn dự kiến.

3.4.4 Thực hiện vẽ đồ thị phân tán thể hiện phân phối của `arr_delay` theo biến `dep_delay`

```
ggplot()+geom_point(data=dataset,aes(x=arr_delay,y=dep_delay))
```



Hình 18: Đồ thị phân tán của hai biến `arr_delay` và `dep_delay`

Nhận xét: Nhìn vào đồ thị trên, ta nhận thấy biến `arr_delay` có mối quan hệ tuyến tính với biến `dep_delay`.

3.5 ANOVA một nhân tố

Ta sẽ thực hiện kiểm định rằng liệu có sự khác biệt về việc lệch giờ bay trung bình giữa các hãng hàng không hay không.

3.5.1 Tại sao lại dùng ANOVA một nhân tố?

Trong bộ dữ liệu mà đề bài đã cho có đến 11 hãng hàng không cung cấp chuyến bay. Việc thực hiện so sánh trung bình của nhiều nhóm, phương pháp tối ưu nhất là dùng phân tích phương sai. Nếu chỉ so sánh 2 trung bình của 2 nhóm, ta có thể dùng `t-test`. Xét cho đề bài này, nếu dùng `t-test`, ta phải thực hiện kiểm định rất nhiều lần. Trong khi đó, phương pháp phân tích phương sai lại cho sự bằng nhau hoặc khác nhau giữa các nhóm so sánh thông qua một phép kiểm định duy nhất nên tối ưu hơn.

Vì vậy, ta sử dụng mô hình `ANOVA` một nhân tố để đánh giá sự khác biệt trong việc lệch giờ bay (`dep_delay`) giữa các hãng bay.

Trong đó:

- Biến phụ thuộc: `dep_delay`.
- Các nhân tố (hay biến độc lập): `carrier`.

3.5.2 Hiện thực

Đặt giả thuyết

- Giả thuyết H_0: $\mu_1 = \mu_2 = \ldots$: Việc lệch giờ bay trung bình giữa các hãng hàng không bằng nhau.
- Đối thuyết H_1: $\exists i \neq j, \mu_i \neq \mu_j$: Có ít nhất 2 hãng hàng không có việc lệch giờ bay trung bình khác nhau.

Các giả định cần kiểm tra trong ANOVA một nhân tố

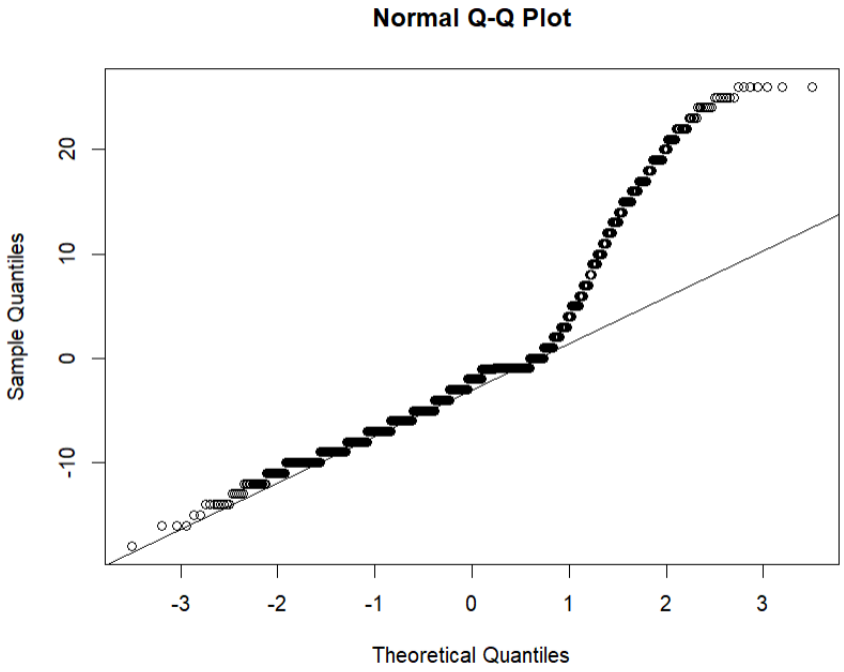
- Giả định phân phối chuẩn: Việc lệch giờ bay của các hãng hàng không tuân theo phân phối chuẩn.
- Tính đồng nhất của các phương sai: Phương sai giữa lệch giờ bay ở các hãng hàng không bằng nhau.

Kiểm tra giả định phân phối chuẩn

- Giả thuyết H_0: Việc lệch giờ bay ở các hãng hàng không tuân theo phân phối chuẩn.
- Giả thuyết đối H_1: Việc lệch giờ bay ở các hãng hàng không không tuân theo phân phối chuẩn.

Để kiểm định giả định phân phối chuẩn cho biến `dep_delay` ở các hãng hàng không, ta sẽ sử dụng thư viện `nortest` và `Anderson - Darling Normality test` để kiểm định.

```
AA= subset(new_dataset , new_dataset$carrier == "AA")
qqnorm(AA$dep_delay)
qqline(AA$dep_delay)
ad.test(AA$dep_delay)
```



Hình 19: Đồ thị kiểm tra phân phối chuẩn cho `AA`

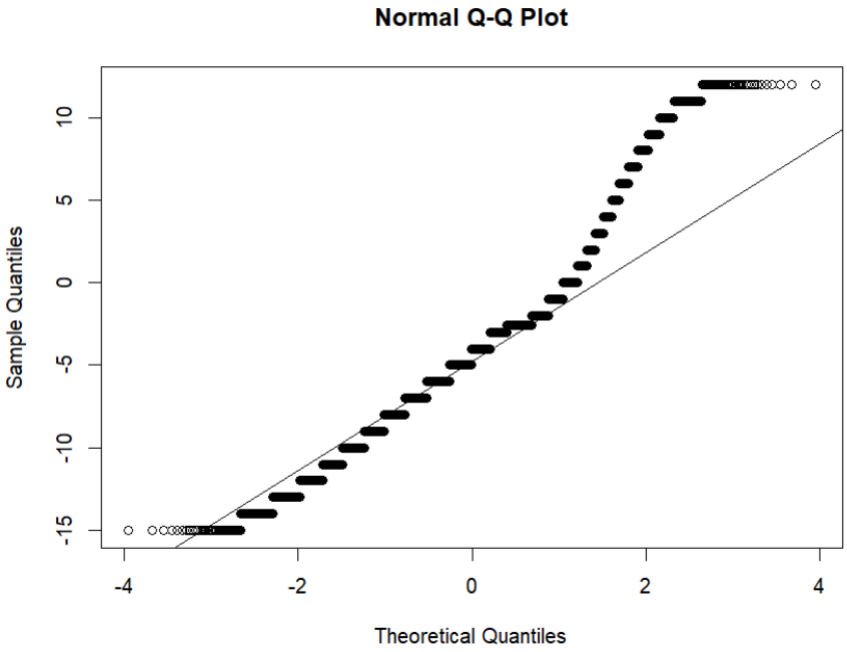
```
        Anderson-Darling normality test

data:  AA$dep_delay
A = 333.64, p-value < 2.2e-16
```

Hình 20: Kết quả kiểm định giả định phân phối chuẩn cho biến `dep_delay` ở hãng hàng không `AA`

Nhận xét: Xét biểu đồ `QQ-plot`, ta nhận thấy có nhiều giá trị quan sát không nằm trên đường thẳng kì vọng của phân phối chuẩn do đó, biến `dep_delay` ở hãng hàng không `AA` không tuân theo phân phối chuẩn.

Ngoài ra, `p_value` ở các kiểm định `ad.test` bé hơn rất nhiều so với mức ý nghĩa $\alpha = 0.05$, nên ta bác bỏ giả thuyết H_0, đồng thời đưa ra kết luận là biến `dep_delay` ở hàng hàng không `AA` không tuân theo phân phối chuẩn.



Hình 21: Đồ thị kiểm tra phân phối chuẩn cho `AS`

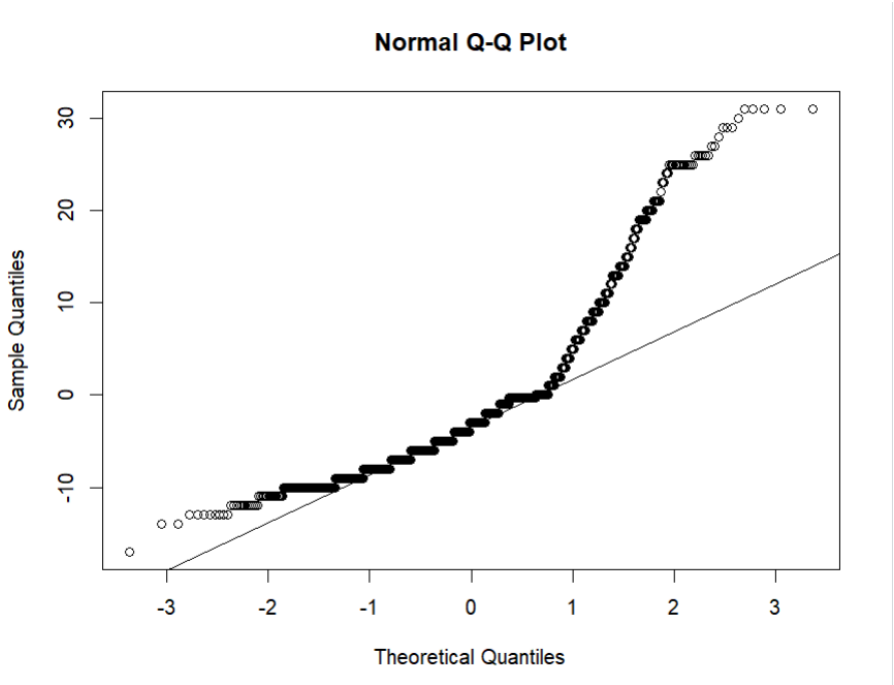
```
        Anderson-Darling normality test

data:  AS$dep_delay
A = 1073.8, p-value < 2.2e-16
```

Hình 22: Kết quả kiểm định giả định phân phối chuẩn cho biến `dep_delay` ở hãng hàng không `AS`

Nhận xét: Xét biểu đồ `QQ-plot`, ta nhận thấy có nhiều giá trị quan sát không nằm trên đường thẳng kì vọng của phân phối chuẩn do đó, biến `dep_delay` ở hãng hàng không `AS` không tuân theo phân phối chuẩn.

Ngoài ra, `p_value` ở các kiểm định `ad.test` bé hơn rất nhiều so với mức ý nghĩa $\alpha = 0.05$, nên ta bác bỏ giả thuyết H_0, đồng thời đưa ra kết luận là biến `dep_delay` ở hàng hàng không `AS` không tuân theo phân phối chuẩn.



Hình 23: Đồ thị kiểm tra phân phối chuẩn cho `B6`

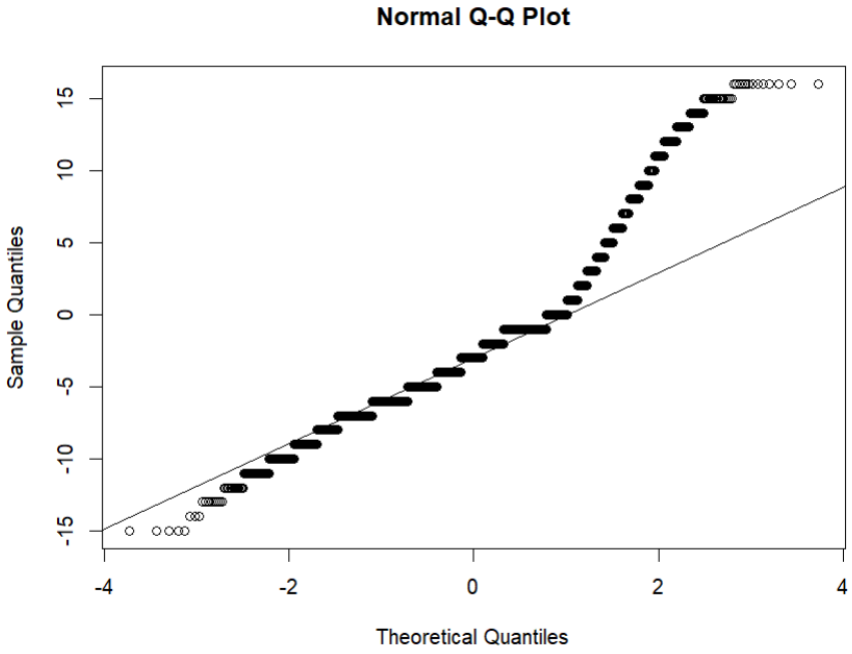
```
        Anderson-Darling normality test

data:  B6$dep_delay
A = 159.29, p-value < 2.2e-16
```

Hình 24: Kết quả kiểm định giả định phân phối chuẩn cho biến `dep_delay` ở hãng hàng không `B6`

Nhận xét: Xét biểu đồ `QQ-plot`, ta nhận thấy có nhiều giá trị quan sát không nằm trên đường thẳng kì vọng của phân phối chuẩn do đó, biến `dep_delay` ở hãng hàng không `B6` không tuân theo phân phối chuẩn.

Ngoài ra, `p_value` ở các kiểm định `ad.test` bé hơn rất nhiều so với mức ý nghĩa $\alpha = 0.05$, nên ta bác bỏ giả thuyết H_0, đồng thời đưa ra kết luận là biến `dep_delay` ở hàng hàng không `B6` không tuân theo phân phối chuẩn.



Hình 25: Đồ thị kiểm tra phân phối chuẩn cho DL

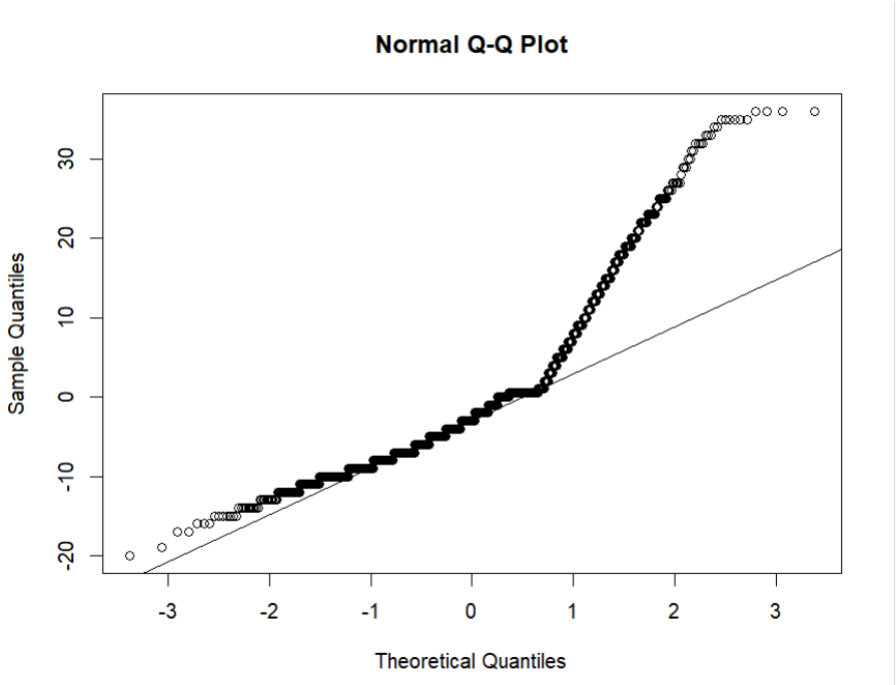
```
        Anderson-Darling normality test

data:  DL$dep_delay
A = 534.15, p-value < 2.2e-16
```

Hình 26: Kết quả kiểm định giả định phân phối chuẩn cho biến `dep_delay` ở hãng hàng không DL

Nhận xét: Xét biểu đồ `QQ-plot`, ta nhận thấy có nhiều giá trị quan sát không nằm trên đường thẳng kì vọng của phân phối chuẩn do đó, biến `dep_delay` ở hãng hàng không DL không tuân theo phân phối chuẩn.

Ngoài ra, `p_value` ở các kiểm định `ad.test` bé hơn rất nhiều so với mức ý nghĩa $\alpha = 0.05$, nên ta bác bỏ giả thuyết H_0, đồng thời đưa ra kết luận là biến `dep_delay` ở hàng hàng không DL không tuân theo phân phối chuẩn.



Hình 27: Đồ thị kiểm tra phân phối chuẩn cho `F9`

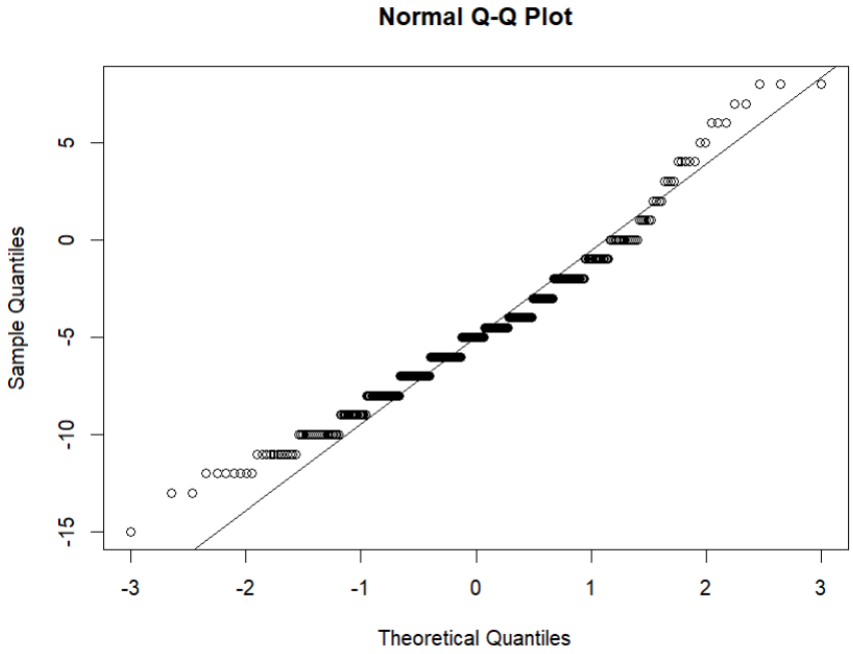
```
        Anderson-Darling normality test

data:  F9$dep_delay
A = 98.334, p-value < 2.2e-16
```

Hình 28: Kết quả kiểm định giả định phân phối chuẩn cho biến `dep_delay` ở hãng hàng không `F9`

Nhận xét: Xét biểu đồ `QQ-plot`, ta nhận thấy có nhiều giá trị quan sát không nằm trên đường thẳng kì vọng của phân phối chuẩn do đó, biến `dep_delay` ở hãng hàng không `F9` không tuân theo phân phối chuẩn.

Ngoài ra, `p_value` ở các kiểm định `ad.test` bé hơn rất nhiều so với mức ý nghĩa $\alpha = 0.05$, nên ta bác bỏ giả thuyết H_0, đồng thời đưa ra kết luận là biến `dep_delay` ở hàng hàng không `F9` không tuân theo phân phối chuẩn.



Hình 29: Đồ thị kiểm tra phân phối chuẩn cho `HA`

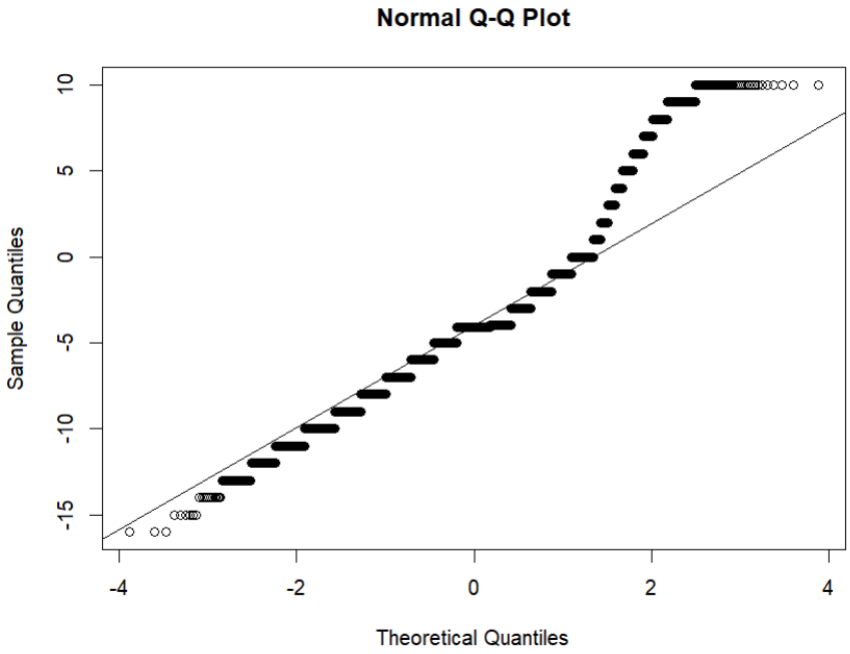
```
        Anderson-Darling normality test

data:  HA$dep_delay
A = 6.7273, p-value < 2.2e-16
```

Hình 30: Kết quả kiểm định giả định phân phối chuẩn cho biến `dep_delay` ở hãng hàng không `HA`

Nhận xét: Xét biểu đồ `QQ-plot`, ta nhận thấy có nhiều giá trị quan sát không nằm trên đường thẳng kì vọng của phân phối chuẩn do đó, biến `dep_delay` ở hãng hàng không `HA` không tuân theo phân phối chuẩn.

Ngoài ra, `p_value` ở các kiểm định `ad.test` bé hơn rất nhiều so với mức ý nghĩa $\alpha = 0.05$, nên ta bác bỏ giả thuyết H_0, đồng thời đưa ra kết luận là biến `dep_delay` ở hàng hàng không `HA` không tuân theo phân phối chuẩn.



Hình 31: Đồ thị kiểm tra phân phối chuẩn cho OO

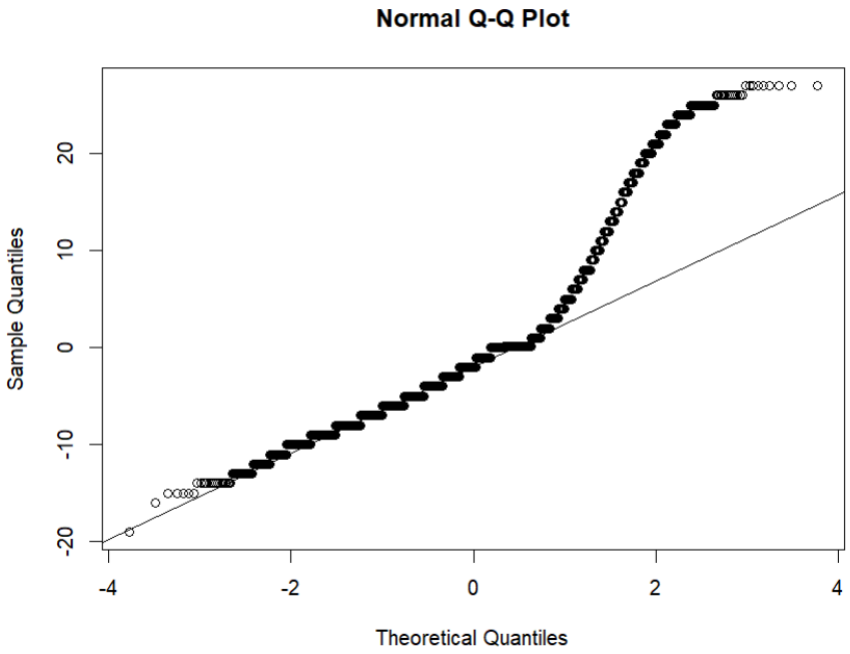
```
        Anderson-Darling normality test

data:  OO$dep_delay
A = 354.77, p-value < 2.2e-16
```

Hình 32: Kết quả kiểm định giả định phân phối chuẩn cho biến `dep_delay` ở hãng hàng không OO

Nhận xét: Xét biểu đồ `QQ-plot`, ta nhận thấy có nhiều giá trị quan sát không nằm trên đường thẳng kì vọng của phân phối chuẩn do đó, biến `dep_delay` ở hãng hàng không OO không tuân theo phân phối chuẩn.

Ngoài ra, `p_value` ở các kiểm định `ad.test` bé hơn rất nhiều so với mức ý nghĩa $\alpha = 0.05$, nên ta bác bỏ giả thuyết H_0, đồng thời đưa ra kết luận là biến `dep_delay` ở hàng hàng không OO không tuân theo phân phối chuẩn.



Hình 33: Đồ thị kiểm tra phân phối chuẩn cho `UA`

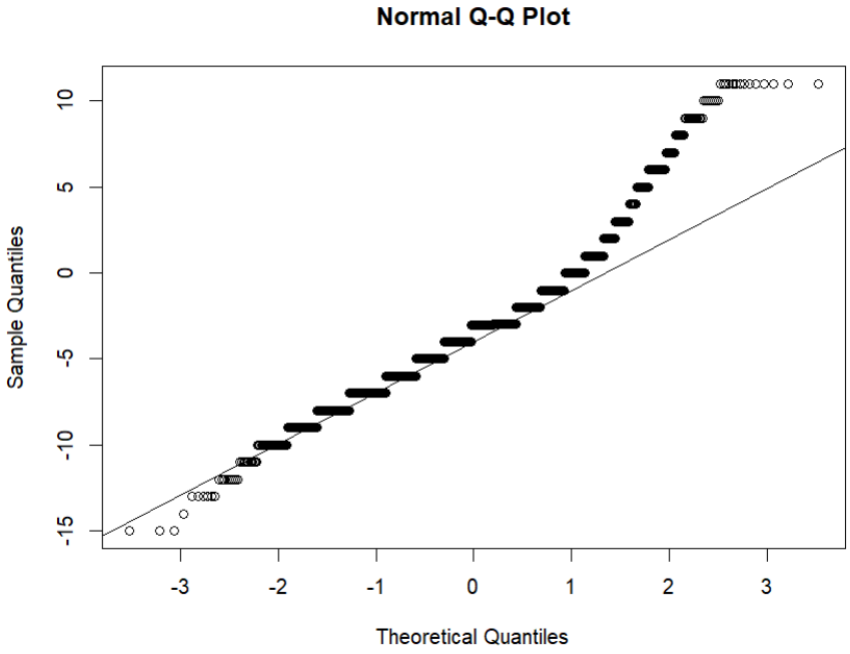
```
        Anderson-Darling normality test

data:  UA$dep_delay
A = 588.4, p-value < 2.2e-16
```

Hình 34: Kết quả kiểm định giả định phân phối chuẩn cho biến `dep_delay` ở hãng hàng không `UA`

Nhận xét: Xét biểu đồ `QQ-plot`, ta nhận thấy có nhiều giá trị quan sát không nằm trên đường thẳng kì vọng của phân phối chuẩn do đó, biến `dep_delay` ở hãng hàng không `UA` không tuân theo phân phối chuẩn.

Ngoài ra, `p_value` ở các kiểm định `ad.test` bé hơn rất nhiều so với mức ý nghĩa $\alpha = 0.05$, nên ta bác bỏ giả thuyết H_0, đồng thời đưa ra kết luận là biến `dep_delay` ở hàng hàng không `UA` không tuân theo phân phối chuẩn.



Hình 35: Đồ thị kiểm tra phân phối chuẩn cho `US`

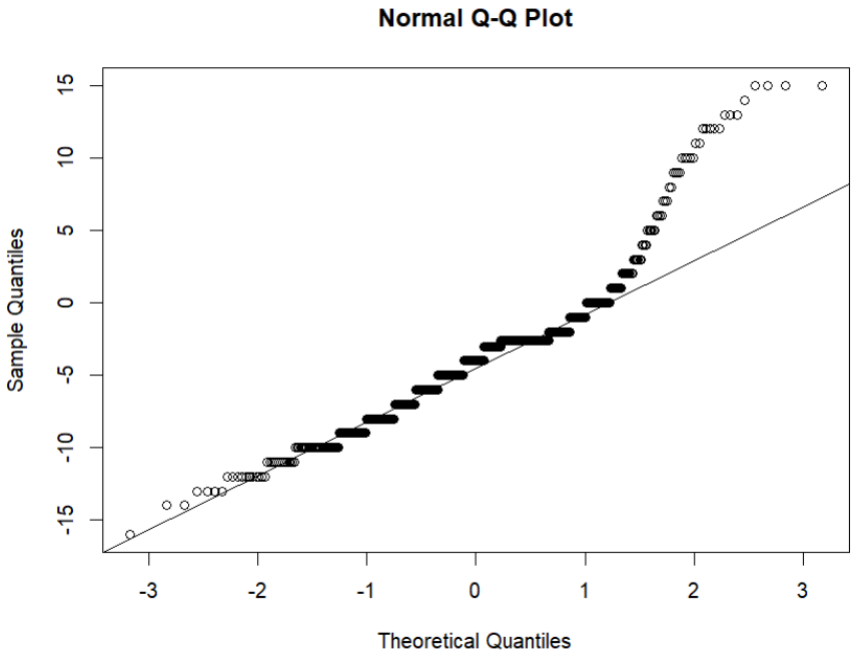
```
        Anderson-Darling normality test

data:  US$dep_delay
A = 101.16, p-value < 2.2e-16
```

Hình 36: Kết quả kiểm định giả định phân phối chuẩn cho biến `dep_delay` ở hãng hàng không `US`

Nhận xét: Xét biểu đồ `QQ-plot`, ta nhận thấy có nhiều giá trị quan sát không nằm trên đường thẳng kì vọng của phân phối chuẩn do đó, biến `dep_delay` ở hãng hàng không `US` không tuân theo phân phối chuẩn.

Ngoài ra, `p_value` ở các kiểm định `ad.test` bé hơn rất nhiều so với mức ý nghĩa $\alpha = 0.05$, nên ta bác bỏ giả thuyết H_0, đồng thời đưa ra kết luận là biến `dep_delay` ở hàng hàng không `US` không tuân theo phân phối chuẩn.



Hình 37: Đồ thị kiểm tra phân phối chuẩn cho `VX`

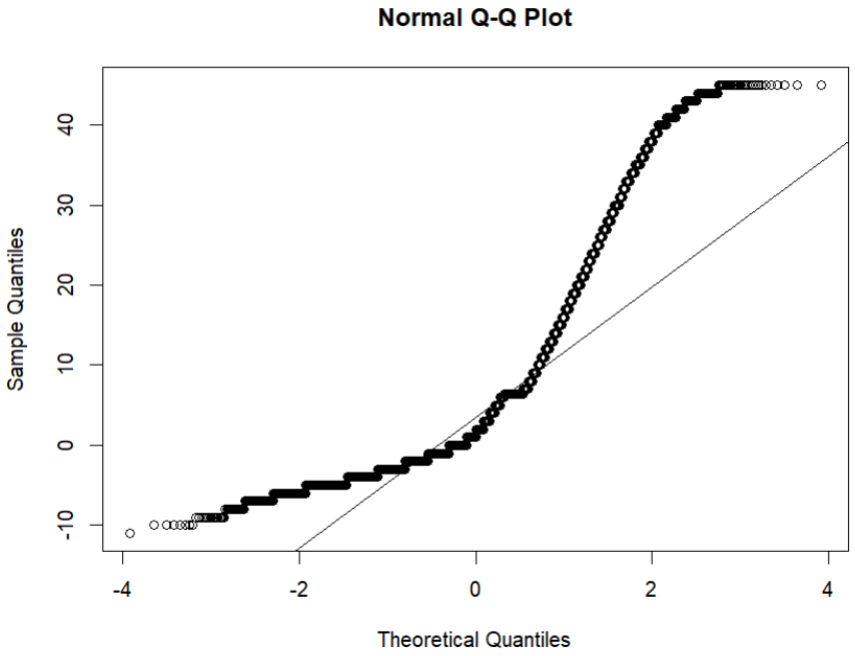
```
        Anderson-Darling normality test

data:  VX$dep_delay
A = 108.5, p-value < 2.2e-16
```

Hình 38: Kết quả kiểm định giả định phân phối chuẩn cho biến `dep_delay` ở hãng hàng không `VX`

Nhận xét: Xét biểu đồ `QQ-plot`, ta nhận thấy có nhiều giá trị quan sát không nằm trên đường thẳng kì vọng của phân phối chuẩn do đó, biến `dep_delay` ở hãng hàng không `VX` không tuân theo phân phối chuẩn.

Ngoài ra, `p_value` ở các kiểm định `ad.test` bé hơn rất nhiều so với mức ý nghĩa $\alpha = 0.05$, nên ta bác bỏ giả thuyết H_0, đồng thời đưa ra kết luận là biến `dep_delay` ở hàng hàng không `VX` không tuân theo phân phối chuẩn.



Hình 39: Đồ thị kiểm tra phân phối chuẩn cho `WN`

```
        Anderson-Darling normality test

data:  WN$dep_delay
A = 1179.8, p-value < 2.2e-16
```

Hình 40: Kết quả kiểm định giả định phân phối chuẩn cho biến `dep_delay` ở hãng hàng không `WN`

Nhận xét: Xét biểu đồ `QQ-plot`, ta nhận thấy có nhiều giá trị quan sát không nằm trên đường thẳng kì vọng của phân phối chuẩn do đó, biến `dep_delay` ở hãng hàng không `WN` không tuân theo phân phối chuẩn.

Ngoài ra, `p_value` ở các kiểm định `ad.test` bé hơn rất nhiều so với mức ý nghĩa $\alpha = 0.05$, nên ta bác bỏ giả thuyết H_0, đồng thời đưa ra kết luận là biến `dep_delay` ở hàng hàng không `WN` không tuân theo phân phối chuẩn.

Kiểm định giả định về tính đồng nhất của các phương sai

- Giả thuyết H_0: Phương sai việc lệch giờ bay ở các hãng hàng không bằng nhau.
- Giả thuyết đối H_1: Có ít nhất 2 hãng hàng không đối có phương sai việc lệch giờ bay khác nhau.

Ta sẽ sử dụng thư viện **car** và **Levene test** để kiểm định giả định về tính đồng nhất của phương sai.

```
library(car)
leveneTest(dep_delay~as.factor(carrier), data =new_dataset)
```

```
> leveneTest(dep_delay~as.factor(carrier), data =new_dataset)
Levene's Test for Homogeneity of Variance (center = median)
          Df F value    Pr(>F)
group     10  2352.7 < 2.2e-16 ***
      160737
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Hình 41: Kết quả khi kiểm định tính đồng nhất của phương sai

Nhận xét: Dựa trên `p_value` ở các kiểm định `leveneTest` bé hơn rất nhiều so với mức ý nghĩa $\alpha = 0.05$, nên ta bác bỏ giả thuyết H_0, đồng thời đưa ra kết luận là có ít nhất 2 hãng hàng không có phương sai lệch giờ khác nhau.

Thực hiện phân tích phương sai một nhân tố

Qua hai phần kiểm định trên, ta thấy dữ liệu của ta không thỏa mãn các giả thiết của phương pháp `ANOVA`. Thay vào đó ta sẽ sử dụng phương pháp phi tham số `Kruskal - Wallis`.

```
kruskal.test(dep_delay~carrier,new_dataset)
```

```
> kruskal.test(dep_delay~carrier,new_dataset)

        Kruskal-Wallis rank sum test

data:  dep_delay by carrier
Kruskal-Wallis chi-squared = 24758, df = 10, p-value < 2.2e-16
```

Hình 42: Kết quả khi thực hiện phương pháp `Kruskal - Wallis`

Nhận xét: Dựa vào kết quả trên ta thấy được giá trị của `p_value` rất bé, xấp xỉ `0`. Vậy nên ta bác bỏ giá thiết H_0, tức là có ít nhất 2 hãng hàng không có việc lệch giờ bay trung bình khác nhau.

So sánh bội sau phân tích phương sai

Ta đã biết rằng ít nhất có một hãng hàng không có việc lệch giờ bay trung bình khác với các hãng khác, tuy nhiên ta cần thêm một so sánh giữa các cặp hãng hàng không để có thêm thông tin. Ta sẽ sử dụng hàm `kruskalmc` trong thư viện `pgirmess` thực hiện so sánh.

```
library(pgirmess)
kruskalmc(new_dataset$dep_delay,new_dataset$carrier)
```

```
Multiple comparison test after Kruskal-Wallis
alpha: 0.05
Comparisons
         obs.dif critical.dif stat.signif
AA-AS  7410.7190     1884.525        TRUE
AA-B6   529.3527     3155.021       FALSE
AA-DL  5058.4388     2143.521        TRUE
AA-F9  4015.3464     3464.416        TRUE
AA-HA 26960.0152     4986.954        TRUE
AA-OO 25614.1027     2111.980        TRUE
AA-UA 10153.8539     2147.253        TRUE
AA-US 13752.9875     2683.845        TRUE
AA-VX 10056.0741     3228.878        TRUE
AA-WN 37874.3019     2047.188        TRUE
AS-B6  6881.3664     2676.710        TRUE
AS-DL 12469.1578     1343.614        TRUE
AS-F9 11426.0654     3035.259        TRUE
AS-HA 19549.2962     4698.971        TRUE
AS-OO 18203.3837     1292.701        TRUE
AS-UA 17564.5730     1349.561        TRUE
AS-US  6342.2684     2100.866        TRUE
AS-VX  2645.3551     2763.381       FALSE
AS-WN 45285.0209     1183.891        TRUE
B6-DL  5587.7914     2864.965        TRUE
B6-F9  4544.6990     3951.647        TRUE
B6-HA 26430.6626     5336.951        TRUE
B6-OO 25084.7500     2841.443        TRUE
B6-UA 10683.2066     2867.758        TRUE
B6-US 13223.6348     3288.824        TRUE
B6-VX  9526.7215     3746.865        TRUE
B6-WN 38403.6546     2793.621        TRUE
DL-F9  1043.0924     3202.505       FALSE
DL-HA 32018.4540     4808.697        TRUE
DL-OO 30672.5415     1647.520        TRUE
DL-UA  5095.4151     1692.502        TRUE
DL-US 18811.4263     2335.998        TRUE
DL-VX 15114.5129     2946.102        TRUE
DL-WN 32815.8631     1563.600        TRUE
F9-HA 30975.3616     5525.492        TRUE
F9-OO 29629.4491     3181.480        TRUE
F9-UA  6138.5076     3205.005        TRUE
F9-US 17768.3338     3586.695        TRUE
F9-VX 14071.4205     4010.862        TRUE
F9-WN 33858.9555     3138.844        TRUE
HA-OO  1345.9125     4794.721       FALSE
HA-UA 37113.8692     4810.362        TRUE
HA-US 13207.0278     5072.663        TRUE
HA-VX 16903.9411     5380.943        TRUE
HA-WN 64834.3172     4766.536        TRUE
OO-UA 35767.9566     1652.374        TRUE
OO-US 11861.1152     2307.090        TRUE
OO-VX 15558.0286     2923.234        TRUE
OO-WN 63488.4046     1520.073        TRUE
UA-US 23906.8414     2339.423        TRUE
UA-VX 20209.9280     2948.819        TRUE
UA-WN 27720.4480     1568.713        TRUE
US-VX  3696.9134     3359.741        TRUE
US-WN 51627.2894     2247.929        TRUE
VX-WN 47930.3760     2876.772        TRUE
```

Hình 43: Kết quả khi thực hiện so sánh bội

Nhận xét: Có thể thấy đa số các cặp hãng hàng không có độ trễ trung bình lệch đáng kể, trừ các nhóm `AA-B6`, `AS-VX`, `DL-F9` và `HA-OO`.

3.6 Mô hình hồi quy tuyến tính

3.6.1 Tìm mô hình chứa các biến phù hợp ảnh hưởng đến nhân tố giờ đến arr_delay

Để phân tích các yếu tố ảnh hưởng việc lệch giờ đến **arr_delay** của các chuyến bay, ta xem biến **arr_delay** là biến phụ thuộc, các biến độc lập là biến hãng hàng không **carrier**, biến sân bay khởi hành **origin**, biến chênh lệch thời gian khởi hành **dep_delay** và biến khoảng cách giữa hai sân bay **distance**. Đây là những yếu tố dự báo có thể giúp giải thích sự biến đổi lệch giờ đến của các chuyến bay.

Ta xây dựng mô hình hồi quy tuyến tính bao gồm:

- Biến phụ thuộc: **arr_delay**.
- Biến độc lập: **carrier, origin, dep_delay, distance** mô hình được biểu diễn như sau.

$$\begin{aligned}\texttt{arr\_delay} = \beta_0 + \beta_1 \times \texttt{carriersAS} + \beta_2 \times \texttt{carriersB6} + \beta_3 \times \texttt{carriersDL} + \beta_4 \times \texttt{carriersF9} \\ + \cdots + \beta_{10} \times \texttt{carriersWN} + \beta_{11} \times \texttt{originSEA} + \beta_{12} \times \texttt{dep\_delay} + \beta_{13} \times \texttt{distance} + \varepsilon\end{aligned}$$

Ta thực hiện ước lượng các hệ số $\beta_0, \mathrm{i} = 0, \ldots, 13$ dựa trên bộ dữ liệu **dataset**.

```
dataset_lr1 <- lm(arr_delay~carrier+origin+dep_delay+distance,new_dataset)
summary(dataset_lr1)
```

```
Call:
lm(formula = arr_delay ~ carrier + origin + dep_delay + distance,
    data = flightscl)

Residuals:
    Min      1Q  Median      3Q     Max
-58.359  -7.061  -0.712   5.960 168.466

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -2.750e-01  1.687e-01  -1.630    0.103
carrierAS    6.322e-01  1.473e-01   4.291 1.78e-05 ***
carrierB6    7.521e-02  2.413e-01   0.312    0.755
carrierDL   -8.403e-01  1.643e-01  -5.116 3.13e-07 ***
carrierF9    2.068e+00  2.678e-01   7.722 1.15e-14 ***
carrierHA    6.628e+00  3.842e-01  17.253  < 2e-16 ***
carrierOO    1.858e-01  1.726e-01   1.076    0.282
carrierUA   -3.494e+00  1.651e-01 -21.165  < 2e-16 ***
carrierUS    1.160e-01  2.058e-01   0.564    0.573
carrierVX   -2.075e+00  2.526e-01  -8.217  < 2e-16 ***
carrierWN   -2.890e+00  1.628e-01 -17.748  < 2e-16 ***
originSEA    3.192e-01  6.508e-02   4.905 9.36e-07 ***
dep_delay    9.965e-01  1.020e-03 977.083  < 2e-16 ***
distance    -2.684e-03  5.273e-05 -50.910  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 11.77 on 160734 degrees of freedom
Multiple R-squared:  0.8577,    Adjusted R-squared:  0.8577
F-statistic: 7.451e+04 on 13 and 160734 DF,  p-value: < 2.2e-16
```

Hình 44: Code R và kết quả khi xây dựng mô hình hồi quy tuyến tính **dataset_lr1**

Nhận xét: Từ kết quả phân tích, ta thu được.
$\hat{\beta}_0 = -2.750\text{e}^{-01}$; $\hat{\beta}_1 = 6.322\text{e}^{-01}$; $\hat{\beta}_2 = 7.521\text{e}^{-02}$; $\hat{\beta}_3 = -8.403\text{e}^{-01}$; $\hat{\beta}_4 = 2.068\text{e}^{+00}$; $\hat{\beta}_5 = 6.628\text{e}^{+00}$; $\hat{\beta}_6 = 1.858\text{e}^{-01}$; $\hat{\beta}_7 = -3.494\text{e}^{+00}$; $\hat{\beta}_8 = 1.160\text{e}^{-01}$; $\hat{\beta}_9 = -2.075\text{e}^{+00}$; $\hat{\beta}_{10} = -2.890\text{e}^{+00}$; $\hat{\beta}_{11} = 3.192\text{e}^{-01}$; $\hat{\beta}_{12} = 9.965\text{e}^{-01}$; $\hat{\beta}_{13} = -2.684\text{e}^{-03}$.

Như vậy, đường thẳng hồi quy được cho bởi phương trình:

$$\begin{aligned}\widehat{\texttt{arr\_delay}} = &-2.750\text{e}^{-01} + 6.322\text{e}^{-01} \times \texttt{carrierAS} + 7.521\text{e}^{-02} \times \texttt{carrierB6} \\ &+ -8.403\text{e}^{-01} \times \texttt{carrierDL} + 2.068\text{e}^{+00} \times \texttt{carrierF9} \\ &+ 6.628\text{e}^{+00} \times \texttt{carrierHA} + 1.858\text{e}^{-01} \times \texttt{carrierOO} \\ &+ -3.494\text{e}^{+00} \times \texttt{carrierUA} + 1.160\text{e}^{-01} \times \texttt{carrierUS} \\ &+ -2.075\text{e}^{+00} \times \texttt{carrierVX} + -2.890\text{e}^{+00} \times \texttt{carrierWN} \\ &+ 3.192\text{e}^{-01} \times \texttt{originSEA} + 9.965\text{e}^{-01} \times \texttt{dep\_delay} + -2.684\text{e}^{-03} \times \texttt{distance}\end{aligned}$$

Kiểm định các hệ số hồi quy:

- Giả thuyết H_0: Hệ số hồi quy không có ý nghĩa thống kê ($\beta_i = 0$).
- Giả thuyết đối H_1: Hệ số hồi quy có ý nghĩa thống kê ($\beta_i \neq 0$).

$\texttt{Pr}(>|\texttt{t}|)$ của các hệ số ứng với biến `carrierB6`, `carrierOO` lớn hơn mức ý nghĩa $\alpha = 0.05$ nên ta chưa đủ cơ sở để bác bỏ giả thuyết H_0. Do đó hệ số đối với biến này không có ý nghĩa với mô hình hồi quy ta xây dựng. Ta có thể cân nhắc loại bỏ biến `carrier` ra khỏi mô hình.

```
dataset_lr2 <- lm(arr_delay~origin+dep_delay+distance,new_dataset)
summary(dataset_lr2) #Summarize
```

```
Call:
lm(formula = arr_delay ~ origin + dep_delay + distance, data = flightscl)

Residuals:
    Min      1Q  Median      3Q     Max
-57.168  -7.224  -0.739   6.153 165.707

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.218e+00  7.148e-02  -17.04   <2e-16 ***
originSEA    6.915e-01  6.382e-02   10.84   <2e-16 ***
dep_delay    9.903e-01  1.020e-03  971.02   <2e-16 ***
distance    -2.537e-03  4.594e-05  -55.23   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 11.88 on 160744 degrees of freedom
Multiple R-squared:  0.8548,    Adjusted R-squared:  0.8548
F-statistic: 3.155e+05 on 3 and 160744 DF,  p-value: < 2.2e-16
```

Hình 45: Code `R` và kết quả khi xây dựng mô hình hồi quy tuyến tính `dataset_lr1` sau khi loại bỏ biến `carrier`

Ta so sánh mô hình 1 và 2:

- Giả thuyết H_0: $\beta_1 = \beta_2 = \cdots = \beta_{10} = 0$: Hai mô hình hoạt động hiệu quả giống nhau (nghĩa là mô hình 2 hiệu quả hơn mô hình 1 vì ít biến hơn).
- Đối thuyết H_1: $\exists \beta_i \neq 0, i = 1, \ldots, 10$: Hai mô hình hiệu quả khác nhau (nghĩa là mô hình 1 hiệu quả hơn mô hình 2).

```
anova(dataset_lr1,dataset_lr2) #Apply anova
```

```
Model 1: arr_delay ~ carrier + origin + dep_delay + distance
Model 2: arr_delay ~ origin + dep_delay + distance
  Res.Df      RSS  Df Sum of Sq     F    Pr(>F)
1 160734 22256571
2 160744 22700776 -10   -444206 320.8 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Hình 46: Code R và kết quả khi so sánh hai mô hình dataset_lr1 và dataset_lr2

Nhận xét: Dựa trên việc so sánh 2 mô hình, ta nhận thấy p_value $< 2.2e^{-16}$, rất bé so với mức ý nghĩa $\alpha = 0.05$ nên ta bác bỏ giả thuyết H_0, ta có thể kết luận 2 mô hình hiệu quả khác nhau, có nghĩa là mô hình 1 hiệu quả hơn mô hình 2 (vì ít nhất có 1 hệ số β_i có ý nghĩa thống kê nên mô hình đầy đủ là mô hình 1 hiệu quả hơn).

Ngoài ra, ta có thể dựa vào hệ số hiệu chỉnh ở mô hình 1 (AdjustedR_squared = 0.8577) cao hơn so với mô hình 2 (AdjustedR_squared = 0.8548), chứng tỏ sự biến thiên của biến arr_delay được giải thích nhiều hơn bởi các biến độc lập. Như vậy mô hình 1 hiệu quả hơn mô hình 2.

3.6.2 Phân tích sự tác động của các nhân tố lên việc lệch giờ đến

Như vậy mô hình hồi quy tuyến tính về việc ảnh hưởng của các nhân tố lên việc lệch giờ đến được cho bởi:

$$\begin{aligned}\widehat{\texttt{arr\_delay}} = &-2.750e^{-01} + 6.322e^{-01} \times \texttt{carrierAS} + 7.521e^{-02} \times \texttt{carrierB6} \\ &+ -8.403e^{-01} \times \texttt{carrierDL} + 2.068e^{+00} \times \texttt{carrierF9} \\ &+ 6.628e^{+00} \times \texttt{carrierHA} + 1.858e^{-01} \times \texttt{carrierOO} \\ &+ -3.494e^{+00} \times \texttt{carrierUA} + 1.160e^{-01} \times \texttt{carrierUS} \\ &+ -2.075e^{+00} \times \texttt{carrierVX} + -2.890e^{+00} \times \texttt{carrierWN} \\ &+ 3.192e^{-01} \times \texttt{originSEA} + 9.965e^{-01} \times \texttt{dep\_delay} + -2.684e^{-03} \times \texttt{distance}\end{aligned}$$

Trước hết, ta thấy rằng p_value tương ứng với thống kê F $< 2.2e^{-16}$, có ý nghĩa rất cao. Điều này chỉ ra rằng, ít nhất một biến mô hình có ý nghĩa giải thích rất cao đến việc lệch giờ bay arr_delay.

Để xét ảnh hưởng cụ thể của từng biến độc lập, ta xét trong hệ số β_i và `p_value` tương ứng. Ta thấy rằng `p_value` tương ứng với các biến `carrierHA`, `carrierVX`, `carrierWN`, `dep_delay`, `distance` $< 2.2e^{-16}$, điều này nói lên rằng ảnh hưởng của các biến này có ý nghĩa rất lơn đến việc lệch giờ bay `arr_delay`.

Mặt khác, hệ số hồi quy β_i của một biến dự báo cũng có thể được xem như ảnh hưởng trung bình lên biến phụ thuộc `arr_delay` khi tăng một đơn vị của biến dự báo đó, giả sử rằng các biến dự báo khác không đổi. Cụ thể, $\hat{\beta}_{12} = 9.965e^{-01}$ có nghĩa rằng khi chênh lệch giờ đi `dep_delay` tăng `1` phút thì ta có thể kỳ vọng chênh lệch giờ đến sẽ tăng lên $9.965e^{-01}$ phút (cho rằng các biến dự báo khác không đổi). Với $\hat{\beta}_{13} = -2.684e^{-03}$ thì khi khoảng cách `distance` giữa hai sân bay tăng `1` dặm, ta có thể kỳ vọng chênh lệch giờ đến giảm $-2.684e^{-03}$ phút (cho rằng các biến dự báo khác không đổi).

Hệ số R^2 hiệu chỉnh bằng `0.8577` nghĩa là `85.77%` sự biến thiên trong việc lệch giờ đến được giải thích bởi các biến độc lập.

3.6.3 Kiểm tra các giả định của mô hình.

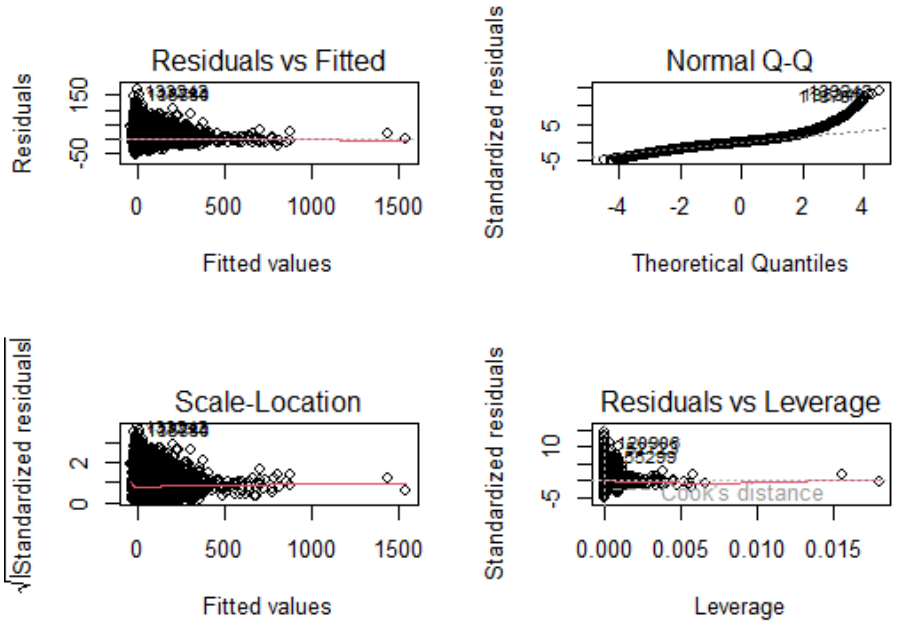
Nhắc lại các giả định của mô hình hồi quy:

$$(U) : Y = \beta_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_4 X_4 + u$$

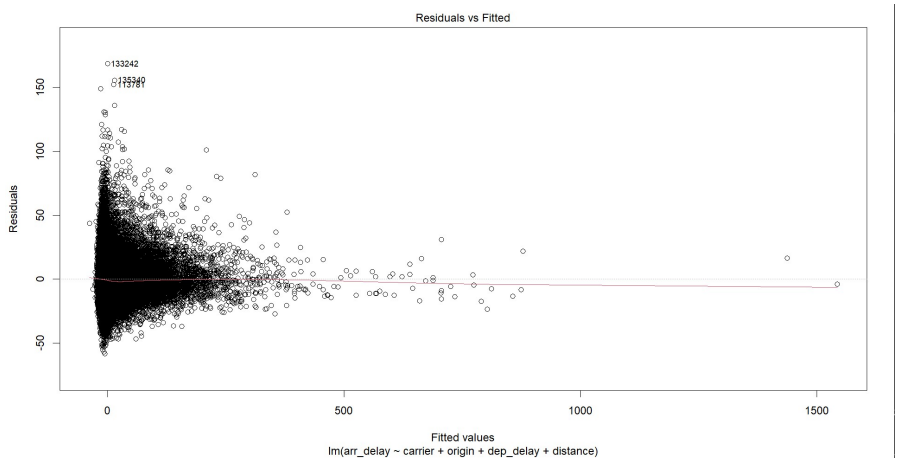
- Tính tuyến tính của dữ liệu: mối quan hệ giữa biến dự báo X và biến phụ thuộc Y được giả sử là tuyến tính.
- Sai số có kỳ vọng bằng `0` và có phân phối chuẩn.
- Phương sai của các sai số là hằng số.
- Các sai số độc lập nhau.

Ta thực hiện phân tích thặng dư để kiểm tra các giả định của mô hình.

```
par(mfrow=c(2,2)) #create matrix 2x2
plot(dataset_lr1) #analyst plot
```



Hình 47: Code R và kết quả đồ thị phân tích thặng dư để kiểm tra các giả định của mô hình



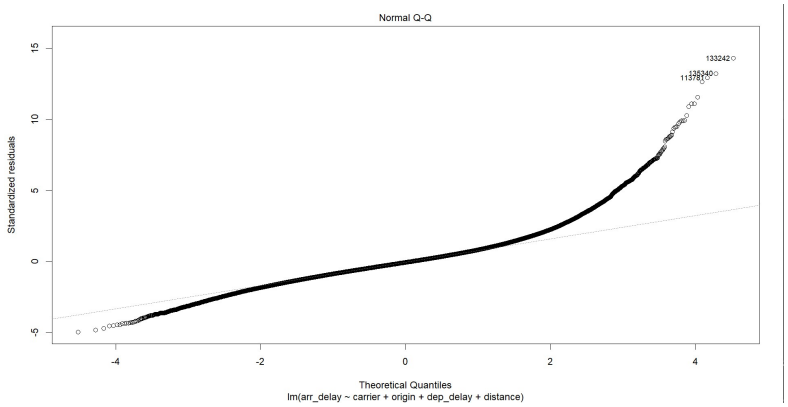
Hình 48: Code R và kết quả khi vẽ đồ thị Residuals and fitted

Nhận xét:

Đồ thị thứ nhất vẽ các sai số tương ứng với các giá trị dự báo, kiểm tra giả định tuyến tính của dữ liệu, giả định sai số có kỳ vọng bằng 0, giả định phương sai của sai số là hằng số.

Dựa trên đồ thị ta thấy, đường màu đỏ là đường thẳng nằm ngang nên giả định tuyến tính của dữ liệu thỏa mãn. Đường màu đỏ nằm sát đường $y = 0$ nên giả định sai số có kỳ vọng bằng

0 thỏa mãn. Các sai số không phân tán ngẫu nhiên dọc theo đường màu đỏ mà phân tán thành cụm ở góc trái đồ thị nên giả định phương sai các biến là hằng số không thỏa mãn.

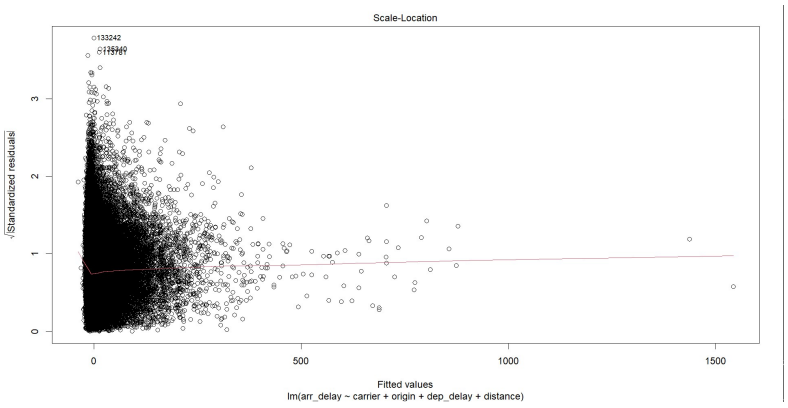


Hình 49: Code R và kết quả vẽ đồ thị QQ-plot

Nhận xét:

Đồ thị thứ hai vẽ các sai số đã được chuẩn hoá, kiểm tra giả định sai số có phân phối chuẩn.

Dựa trên đồ thị ta thấy, có nhiều điểm quan trắc lệch ra khỏi dường thẳng kì vọng phân phối chuẩn nên giả định sai số có phân phối chuẩn chưa thoả mãn.

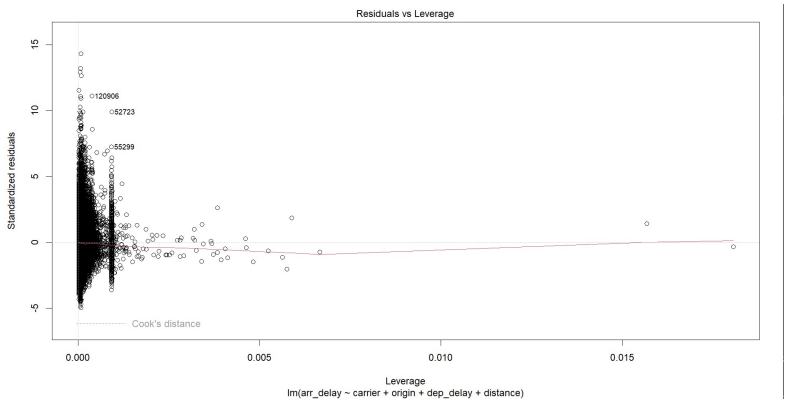


Hình 50: Code R và kết quả khi vẽ đồ thị Scale - Location

Nhận xét:

Đồ thị thứ ba vẽ căn bậc hai của các sai số đã được chuẩn hoá, kiểm tra giả định phương sai các sai số là hằng số.

Dựa vào đồ thị ta thấy, đường màu đỏ nằm ngang nhưng các quan trắc không phân tán ngẫu nhiên dọc theo đường màu đỏ mà phân tán thành cụm ở góc trái đồ thị nên giả định phương sai của các biến là các hằng số không thỏa mãn.



Hình 51: Code R và kết quả khi vẽ đồ thị Residual and Leverage

Nhận xét:

Đồ thị thứ tư chỉ ra các quan trắc thứ 52723, 55299 và 120906 có thể là các điểm có ảnh hưởng cao trong bộ dữ liệu. Tuy nhiên ta không thấy đường Cook ở góc đồ thị bên phải và các điểm này cũng không vượt ra khỏi đường Cook nên các điểm này không thực sự là điểm có ảnh hưởng cao, do đó ta không cần loại bỏ các điểm này khi phân tích.

4 Hoạt động 2

4.1 Yêu cầu

Sinh viên tự tìm một bộ dữ liệu thuộc về chuyên ngành của mình. Khuyến khích sinh viên sử dụng dữ liệu thực tế sẵn có từ các thí nghiệm, khảo sát, dự án,... trong chuyên ngành của mình. Ngoài ra sinh viên có thể tự tìm kiếm dữ liệu từ những nguồn khác hoặc tham khảo trong kho dữ liệu cung cấp trong tập tin `kho_du_lieu_BTL_xstk.xlsx`.

Sinh viên được tự do chọn phương pháp lý thuyết phù hợp để áp dụng phân tích dữ liệu của mình, nhưng phải đảm bảo 2 phần: Làm rõ dữ liệu (`data visualization`) và mô hình dữ liệu (`model fitting`).

Trên cơ sở đó, nhóm đã lựa chọn bộ dữ liệu `e-commerce.csv` - liên quan đến chuyên ngành Khoa học Máy tính của nhóm ở lĩnh vực thương mại điện tử để hiện thực hoạt động 2, làm rõ dữ liệu được cho trong bộ dữ liệu và mô hình dữ liệu dựa trên các phương pháp đã nêu rõ ở phần cơ sở lý thuyết.

4.2 Sơ lược về bộ dữ liệu

Bộ dữ liệu được cho trong tập tin `e-commerce.csv` là thông tin về những lượt mua hàng từ một công ty thương mại điện tử bán đồ dùng điện gia dụng. Từ bộ dữ liệu trên, hành vi của người mua hàng được ghi nhận để dự đoán xu hướng thị trường, cũng như điều chỉnh chiến lược kinh doanh của công ty.

Bộ dữ liệu chứa `10999` hàng, `12` biến, bao gồm:

- `ID`: ID của khách hàng.
- `Warehouse_block`: Tên kho xuất hàng.
- `Mode_of_Shipment`: Phương thức vận chuyển hàng.
- `Customer_care_calls`: Số cuộc gọi từ công ty đến khách hàng.
- `Customer_rating`: Đánh giá của khách hàng.
- `Cost_of_the_Product`: Giá thành của sản phẩm (tính theo $).
- `Prior_purchases`: Số lần mà khách hàng đã mua sản phẩm đó trước đây.
- `Product_importance`: Mức độ quan trọng của mặt hàng.
- `Gender`: Giới tính của khách hàng.
- `Discount_offered`: Phần trăm giảm giá của sản phẩm.
- `Weight_in_gms`: Khối lượng của sản phẩm (tính theo gram).
- `Reached.on.Time_Y.N`: `0` nếu giao hàng đúng hạn, `1` nếu giao hàng trễ hạn.

4.3 Đọc dữ liệu (Import data)

```
library(readr)
library(dplyr)
library(ggplot2)
library(hrbrthemes)

rm(list = ls()) #Clear environments

dataset <- read.csv("e-commerce.csv")
head(dataset, 3) #Get first three lines of the dataset
```

Ta sử dụng hàm `read.csv()` từ thư viện `readr` để đọc vào dữ liệu từ bộ dữ liệu `e-commerce.csv`, xuất mẫu 3 dòng đầu tiên của bộ dữ liệu.

ID	Warehouse_block	Mode_of_Shipment	Customer_care_calls	Customer_rating	Cost_of_the_Product
1	D	Flight	4	2	177
2	F	Flight	4	5	216
3	A	Flight	2	2	183

Prior_purchases	Product_importance	Gender	Discount_offered	Weight_in_gms	Reached.on.Time_Y.N
3	low	F	44	1233	1
2	low	M	59	3088	1
4	low	M	48	3374	1

Hình 52: Bộ dữ liệu được đọc từ `e-commerce.csv`

4.4 Làm sạch dữ liệu (Data cleaning)

Kiểm tra dữ liệu khuyết trong bộ dữ liệu.

```
apply(is.na(dataset),2,sum)
```

ID	Warehouse_block	Mode_of_Shipment	Customer_care_calls
0	0	0	0

Customer_rating	Cost_of_the_Product	Prior_purchases	Product_importance
0	0	0	0

Gender	Discount_offered	Weight_in_gms	Reached.on.Time_Y.N
0	0	0	0

Hình 53: Thống kê số lượng giá trị khuyết đối với từng biến

Nhận xét: Từ số liệu như trên, ta có thể thấy bộ dữ liệu hoàn toàn không chứa giá trị `NA`, do đó ta có thể bỏ qua bước làm sạch dữ liệu.

4.5 Làm rõ dữ liệu (Data visualization)

4.5.1 Một số thuộc tính cơ bản của bộ dữ liệu

4.5.1.a Biến Warehouse_block

```
ware <- as.data.frame(table(dataset$Warehouse_block))
colnames(ware) <- c("Warehouse block", "Customers count")
```

	Warehouse block	Customers count
1	A	1833
2	B	1833
3	C	1833
4	D	1834
5	F	3666

Hình 54: Thống kê số lượng khách hàng ứng với từng kho hàng

Ta sử dụng thư viện ggplot2 để vẽ biểu đồ tròn như sau.

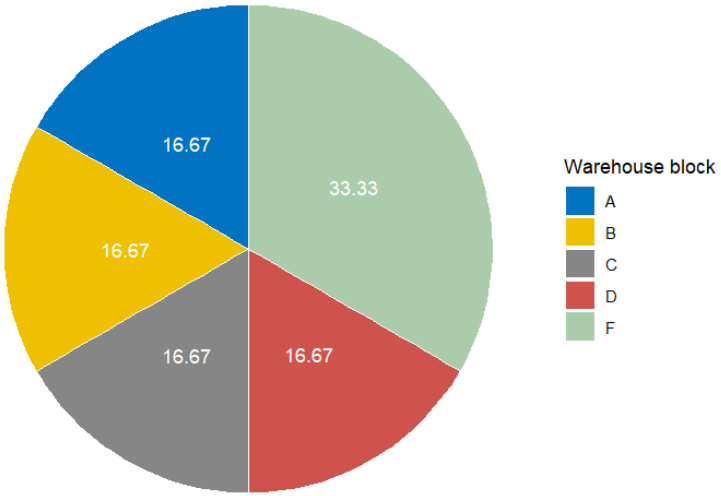
```
ware$Percentage <- round(ware$"Customers count" / sum(ware$"Customers count") * 100,
digits = 2) #Add another column for percentages

ware <- ware %>%
    arrange(desc(`Warehouse block`)) %>%
    mutate(ypos = cumsum(Percentage) - 0.5*Percentage) #Set label position

mycols <- c("#0073C2FF", "#EFC000FF", "#868686FF", "#CD534CFF", "#ABCCAAFF")
#Define set of colors

pie_ware <- ggplot(ware, aes(x = "", y = Percentage, fill = `Warehouse block`)) +
    geom_bar(width = 1, stat = "identity", color = "white") +
    coord_polar("y", start = 0) +
    geom_text(aes(y = ypos, label = Percentage), color = "white")+
    scale_fill_manual(values = mycols) +
    theme_void() #Draw pie chart

pie_ware
```



Hình 55: Tỷ lệ số lượng khách hàng ứng với từng kho hàng

4.5.1.b Biến Mode_Of_Shipment

```
shipment <- as.data.frame(table(dataset$Mode_of_Shipment))
colnames(shipment) <- c("Mode of Shipment", "Customers count")
```

	Mode of Shipment	Customers count
1	Flight	1777
2	Road	1760
3	Ship	7462

Hình 56: Thống kê số lượng khách hàng ứng với từng phương thức giao hàng

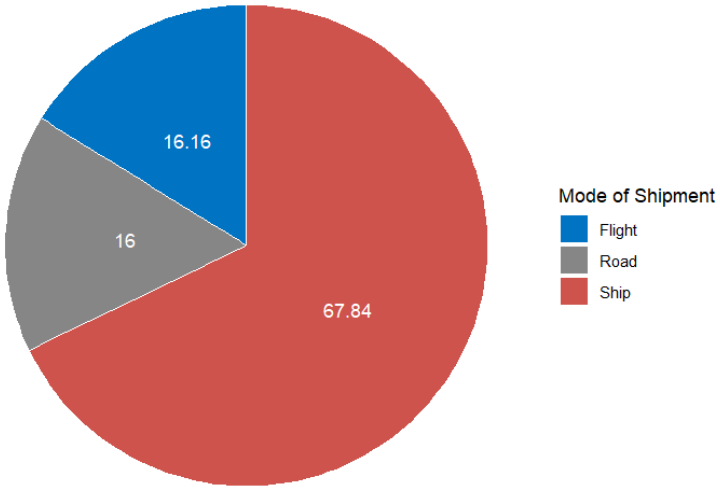
Ta tiếp tục sử dụng thư viên ggplot2 để vẽ biểu đồ tròn như sau.

```
shipment$Percentage <- round(shipment$"Customers count" / sum(shipment$
"Customers count") * 100, digits = 2) #Add another column for percentages
shipment <- shipment %>%
    arrange(desc(`Mode of Shipment`)) %>%
    mutate(ypos = cumsum(Percentage) - 0.5*Percentage) #Set label position

mycols2 <- c("#0073C2FF", "#868686FF", "#CD534CFF") #Define set of colors
```

```
pie_shipment <- ggplot(shipment, aes(x = "", y = Percentage, fill = `Mode of Shipment`)) +
    geom_bar(width = 1, stat = "identity", color = "white") +
    coord_polar("y", start = 0)+
    geom_text(aes(y = ypos, label = Percentage), color = "white")+
    scale_fill_manual(values = mycols2) +
    theme_void() #Draw pie chart

pie_shipment
```



Hình 57: Tỷ lệ số lượng khách hàng ứng với từng phương thức giao hàng

4.5.1.c Biến Reached.on.Time_Y.N

```
time <- as.data.frame(table(dataset$Reached.on.Time_Y.N))
colnames(time) <- c("Reached on time? (0) On time, (1) Not on time",
"Customers count")
```

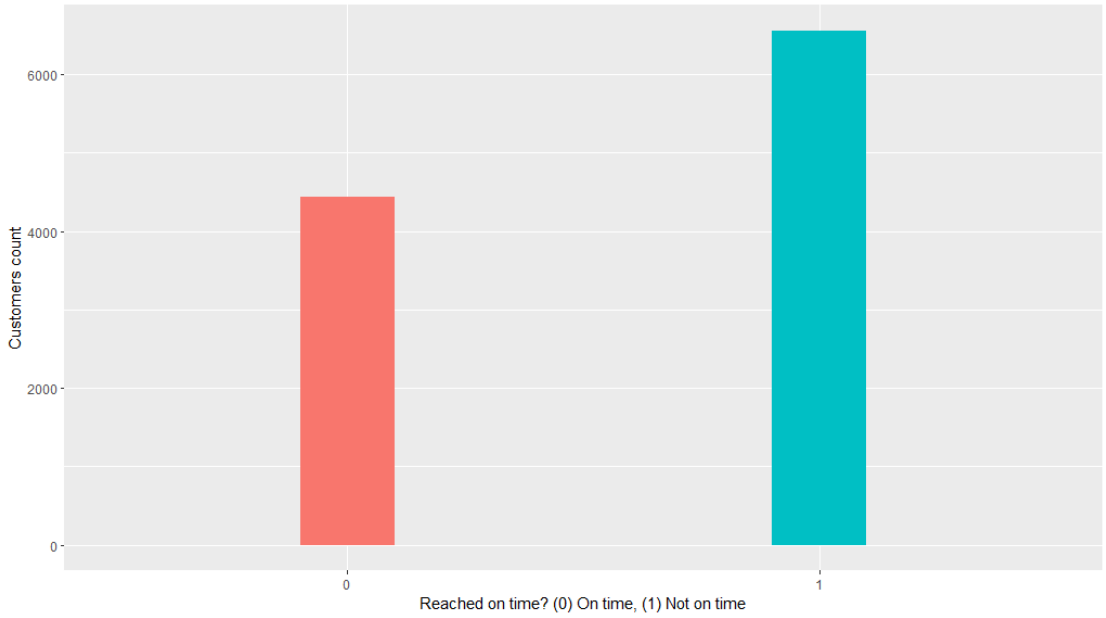
	Reached on time? (0) On time, (1) Not on time	Customers count
1	0	4436
2	1	6563

Hình 58: Thống kê số lượng khách hàng được giao hàng đúng hạn và số lượng khách hàng được giao hàng trễ hạn

Ta dựng biểu đồ cột cho biến Reached.on.Time_Y.N sử dụng ggplot2 như sau.

```
bar_time <- ggplot(time, aes(x=`Reached on time? (0) On time, (1) Not on time`,
y=`Customers count`, fill = `Reached on time? (0) On time, (1) Not on time`)) +
    geom_bar(stat = "identity", width = 0.2) +
    theme(legend.position="none")

bar_time
```



Hình 59: Biểu đồ thống kê số lượng khách hàng được giao hàng đúng hạn và số lượng khách hàng được giao hàng trễ hạn

4.5.2 Các thông số thống kê đặc trưng của biến `Customer_care_calls`

```
total_calls <- sum(dataset$Customer_care_calls)
total_calls
```

Qua đoạn code trên, ta thu được tổng số cuộc gọi chăm sóc khách hàng là `44595` cuộc gọi.

Tiếp theo, ta sẽ thống kê các thuộc tính đặc trưng của biến `Customer_care_calls` cho từng kho hàng (biến `Warehouse_block`).

```
calls <- dataset %>%
    group_by(Warehouse_block) %>%
        summarise(
            "Customer Care Calls count" = sum(Customer_care_calls),
            "Average" = mean(Customer_care_calls),
            "Minimum" = min(Customer_care_calls),
```

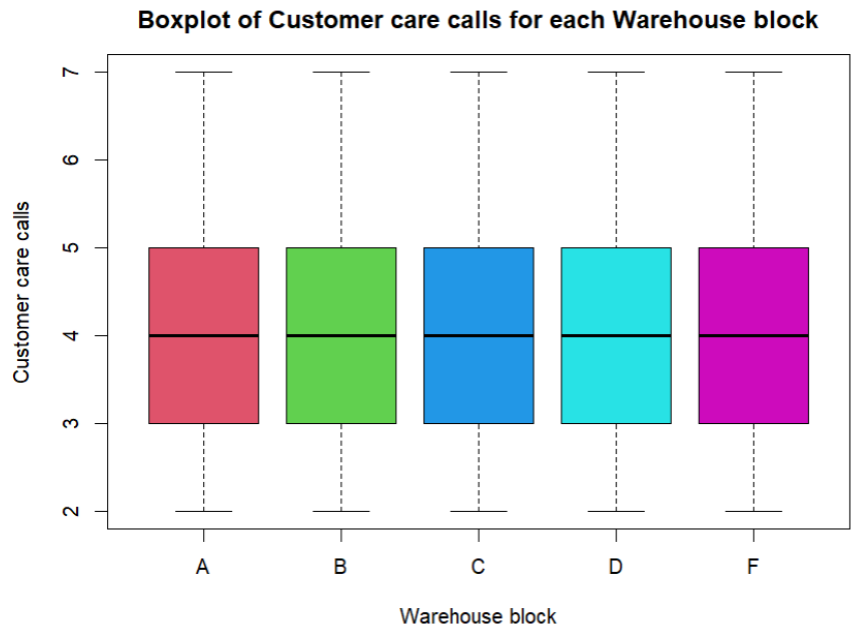
```
        "Maximum" = max(Customer_care_calls)
    ) # Calls info for each Warehouse
```

	Warehouse_block	Customer Care Calls count	Average	Minimum	Maximum
1	A	7402	4.038189	2	7
2	B	7369	4.020185	2	7
3	C	7451	4.064921	2	7
4	D	7434	4.053435	2	7
5	F	14939	4.075014	2	7

Hình 60: Thống kê số lượng cuộc gọi chăm sóc khách hàng cho từng kho hàng

Ta sử dụng biểu đồ hộp (Boxplot) để mô hình hóa dữ liệu về số lượng cuộc gọi chăm sóc khách hàng do từng kho hàng đảm nhận.

```
boxplot(Customer_care_calls ~ Warehouse_block, xlab = "Warehouse block", ylab = "Customer
care calls", main = "Boxplot of Customer care calls for each Warehouse block",
data = dataset , col = 2:7)
```



Hình 61: Biểu đồ hộp thống kê số lượng cuộc gọi chăm sóc khách hàng cho từng kho hàng

Nhận xét: Các kho hàng `A, B, C, D, F` đều có số cuộc gọi chăm sóc khách hàng cho một đơn hàng tối thiểu là `2` cuộc gọi, tối đa là `7` cuộc gọi, và trung bình khoảng `4` cuộc gọi. Tuy nhiên, về tổng số cuộc gọi mà từng kho hàng đảm nhận, các kho hàng `A, B, C, D` có tổng số cuộc gọi chăm sóc khách hàng rơi vào khoảng `7400` cuộc gọi, trong khi đó `14939` cuộc gọi là con số mà kho hàng `F` phải đảm nhận - gấp `2` lần các kho còn lại.

4.5.3 Các thông số thống kê đặc trưng của biến `Customer_rating`

```
rating <- as.matrix(table(dataset$Customer_rating,dataset$Warehouse_block))
#Ratings for each warehouse block
rating
```

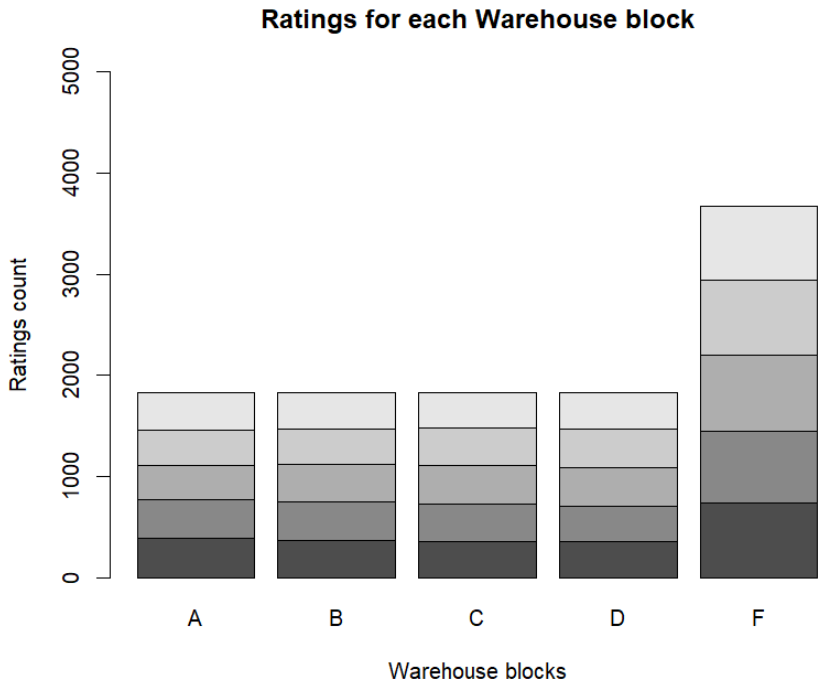
Sử dụng đoạn code trên, ta thu được ma trận `5 x 5` thể hiện đánh giá của khách hàng trên sản phẩm của mỗi kho hàng.

	A	B	C	D	F
1	394	371	364	364	742
2	376	376	362	340	711
3	345	371	383	390	750
4	350	348	369	379	743
5	368	367	355	361	720

Hình 62: Thống kê đánh giá của khách hàng trên sản phẩm của mỗi kho hàng

Ta sử dụng biểu đồ cột chồng để biểu diễn số lượng đánh giá của khách hàng trên sản phẩm của mỗi kho hàng - với màu sắc trên từng cột từ đậm đến nhạt tương ứng với đánh giá của khách hàng từ `1 - 5`.

```
barplot(rating, main = "Ratings for each Warehouse block",
    xlab ="Warehouse blocks", ylab = "Ratings count", ylim = c(0, 5000))
    #Stacked bar plot for ratings
```



Hình 63: Biểu đồ cột chồng thống kê đánh giá của khách hàng trên sản phẩm của mỗi kho hàng

Nhận xét: Các kho hàng `A, B, C, D` có số lượng đánh giá của khách hàng cho từng giá trị từ `1 - 5` là gần bằng nhau, riêng kho hàng `F` có số lượng đánh giá vượt trội, và điều này có thể giải thích được bởi lượng khách mua hàng từ kho hàng `F` nhiều hơn `4` kho hàng còn lại. Đồng thời, từng kho hàng có số lượng đánh giá ứng với từng giá trị từ `1 - 5` gần như bằng nhau.

4.5.4 Các thông số thống kê đặc trưng của biến `Cost_of_the_Product`

```
costs <- dataset %>%
  summarise(
    "Average" = mean(Cost_of_the_Product),
    "Minimum" = min(Cost_of_the_Product),
    "Maximum" = max(Cost_of_the_Product),
    "Q1" = quantile(Cost_of_the_Product, 0.25),
    "Q2" = median(Cost_of_the_Product),
    "Q3" = quantile(Cost_of_the_Product, 0.75)
  ) # Costs summary

costs
```

Average	Minimum	Maximum	Q1	Q2	Q3
210.1968	96	310	169	214	251

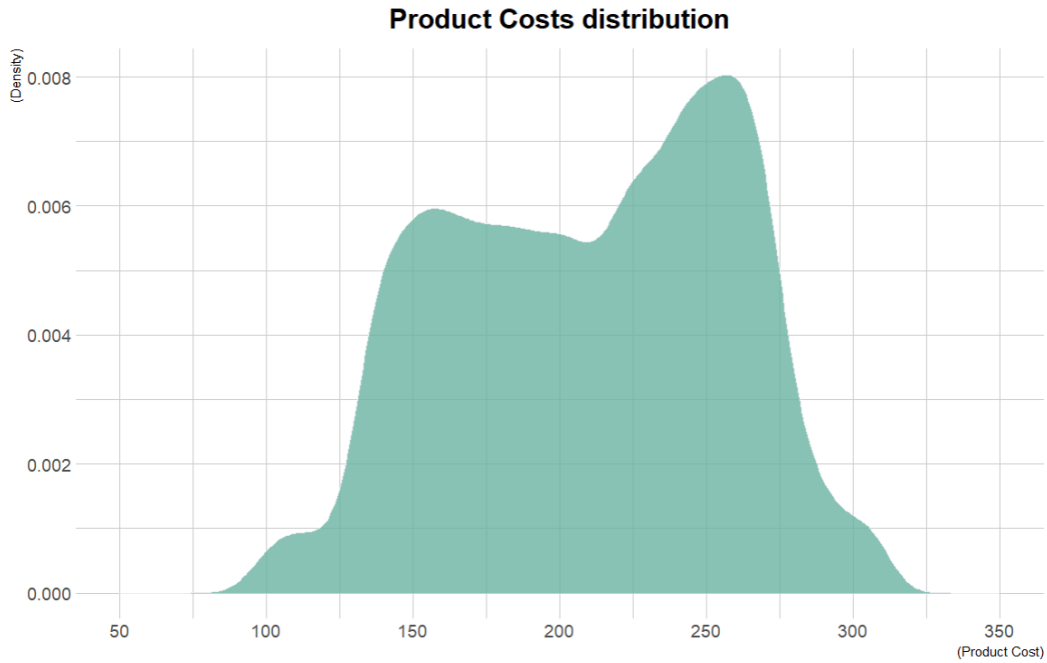
Hình 64: Thống kê sơ bộ về giá tiền của các mặt hàng

Trên đây là số liệu về giá tiền của các mặt hàng được bán ra, với:

- Giá thấp nhất là 96$.
- Giá cao nhất là 310$.
- Trung bình mỗi mặt hàng có giá 210.1968$.
- 25% mặt hàng có giá thấp hơn hoặc bằng 169$.
- 50% mặt hàng có giá thấp hơn hoặc bằng 214$.
- 75% mặt hàng có giá thấp hơn hoặc bằng 251$.

Ta thống kê giá tiền của các mặt hàng sử dụng biểu đồ mật độ như sau.

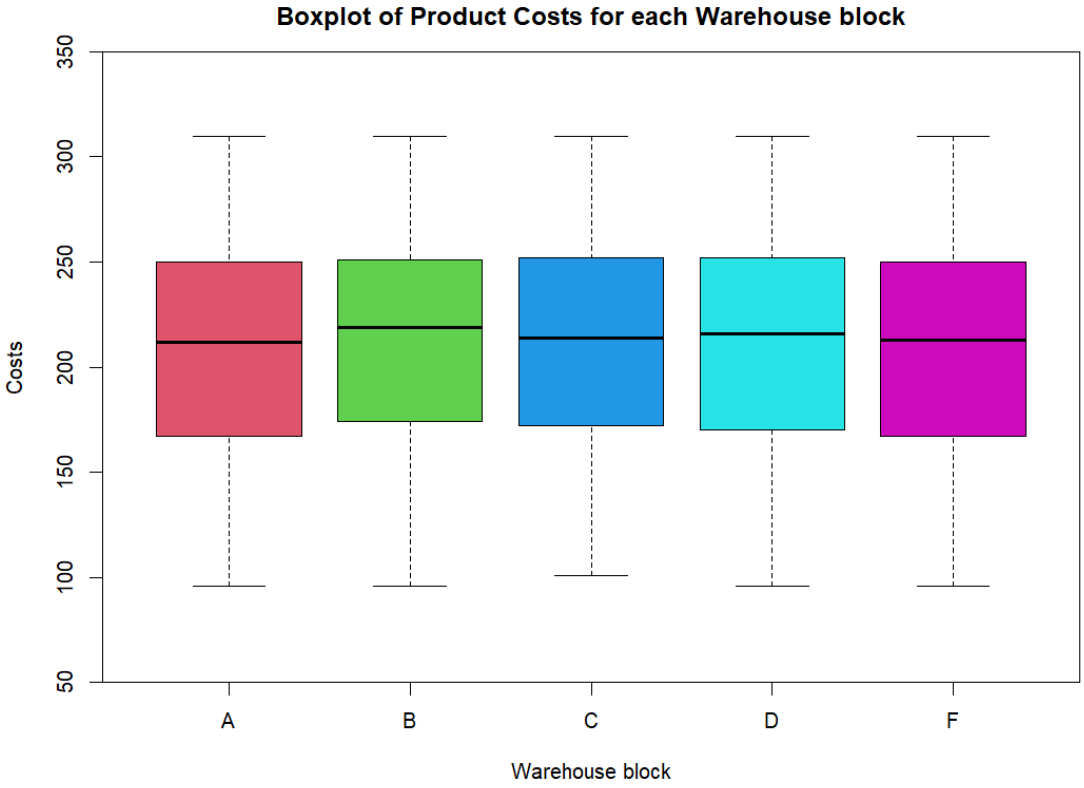
```
dataset %>%
    ggplot( aes(x=Cost_of_the_Product)) +
    geom_density(fill="#69b3a2", color="#e9ecef", alpha=0.8) +
    ggtitle("Product Costs distribution") +
    theme_ipsum() +
    theme(plot.title = element_text(hjust = 0.5)) +
    xlab("(Product Cost)") +
    ylab("(Density)") +
    scale_x_continuous(limits = c(50, 350), breaks = seq(50, 350, by = 50))
    #Density chart for costs distribution
```



Hình 65: Biểu đồ mật độ thống kê giá tiền của các mặt hàng

Nhận xét: Giá tiền của các mặt hàng dao động trong khoảng từ 96$ đến 310$. Khoảng giá tiền có nhiều mặt hàng nhất là 250$ đến 260$, trong đó đỉnh điểm là 257$.

```
boxplot(Cost_of_the_Product ~ Warehouse_block, xlab = "Warehouse block", ylab = "Costs",
main = "Boxplot of Product Costs for each Warehouse block", data = dataset, col = 2:7,
ylim = c(50, 350)) #Box plot for costs
```



Hình 66: Biểu đồ hộp thống kê giá tiền của các mặt hàng theo từng kho hàng

Nhận xét: Các kho hàng có phân phối giá tiền các mặt hàng gần giống nhau, tuy nhiên kho hàng `B` có giá tiền trung vị nhỉnh hơn so với các kho hàng còn lại.

4.5.5 Các thông số thống kê đặc trưng của biến `Weight_in_gms`

```
weight <- dataset %>%
summarise(
    "Average" = mean(Weight_in_gms),
    "Minimum" = min(Weight_in_gms),
    "Maximum" = max(Weight_in_gms),
    "Q1" = quantile(Weight_in_gms, 0.25),
    "Q2" = median(Weight_in_gms),
    "Q3" = quantile(Weight_in_gms, 0.75)
) # Weight summary

weight
```

Average	Minimum	Maximum	Q1	Q2	Q3
3634.017	1001	7846	1839.5	4149	5050

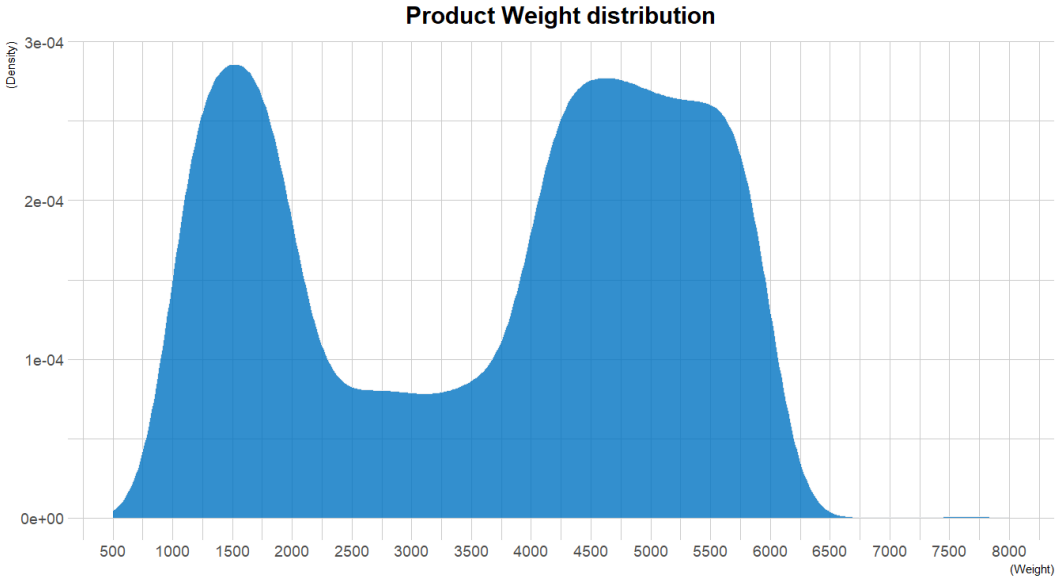
Hình 67: Thống kê sơ bộ về khối lượng của các mặt hàng

Trên đây là số liệu về khối lượng của các mặt hàng được bán ra, cụ thể như sau:

- Nhẹ nhất là 1001g.
- Nặng nhất là 7846g.
- Trung bình mỗi mặt hàng có khối lượng 3634.017g.
- 25% mặt hàng có khối lượng thấp hơn hoặc bằng 1839.5g.
- 50% mặt hàng có khối lượng thấp hơn hoặc bằng 4149g.
- 75% mặt hàng có khối lượng thấp hơn hoặc bằng 5050g.

Ta biểu diễn sự phân phối của khối lượng của các mặt hàng sử dụng biểu đồ mật độ như sau.

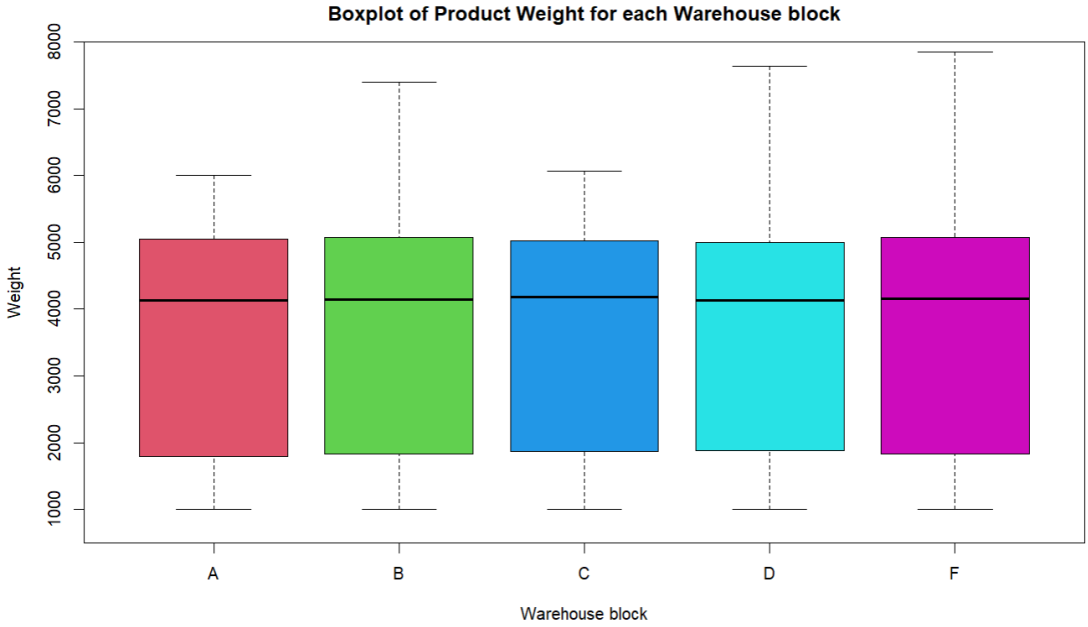
```
dataset %>%
    ggplot( aes(x=Weight_in_gms)) +
    geom_density(fill="#0073C2FF", color="#e9ecef", alpha=0.8) +
    ggtitle("Product Weight distribution") +
    theme_ipsum() +
    theme(plot.title = element_text(hjust = 0.5)) +
    xlab("(Weight)") +
    ylab("(Density)") +
    scale_x_continuous(limits = c(500, 8000), breaks = seq(500, 8000,
    by = 500))
    #Density chart for weight distribution
```



Hình 68: Biểu đồ mật độ thống kê khối lượng của các mặt hàng

Nhận xét: Khối lượng của các mặt hàng dao động trong khoảng từ 1001g đến 7846g. Ta có 2 khoảng khối lượng có nhiều sản phẩm nhất, đó là khoảng xung quanh 1500g và 4500g.

```
boxplot(Weight_in_gms ~ Warehouse_block, xlab = "Warehouse block", ylab = "Weight",
main = "Boxplot of Product Weight for each Warehouse block",
data = dataset, col = 2:7, ylim = c(500, 8000)) #Box plot for costs
```



Hình 69: Biểu đồ hộp thống kê khối lượng của các mặt hàng theo từng kho hàng

Nhận xét: Nhìn chung, các kho hàng có phân phối về khối lượng của các mặt hàng khá giống nhau. Điểm khác biệt rõ nhất là giá trị cực đại cho từng kho hàng, với kho hàng `F` cao nhất, sau đó đến các kho hàng `B, D` và cuối cùng là kho hàng `A, C`.

4.6 Mô hình hồi quy tuyến tính

4.6.1 Phân tích các yếu tố ảnh hưởng đến giá tiền

Ta xây dựng mô hình hồi quy tuyến tính như sau:

- Biến phụ thuộc: `Cost_of_the_Product`.
- Biến độc lập: `Customer_care_calls`, `Customer_rating`, `Prior_purchases`, `Discount_offered`, `Weight_in_gms`, `Reached.on.Time_Y.N`.

Mô hình được biểu diễn như sau:

$$\begin{aligned}\texttt{Cost\_of\_the\_Product} = \beta_0 &+ \beta_1 \times \texttt{Customer\_care\_calls} + \beta_2 \times \texttt{Customer\_rating} \\ &+ \beta_3 \times \texttt{Prior\_purchases} + \beta_4 \times \texttt{Discount\_offered} \\ &+ \beta_5 \times \texttt{Weight\_in\_gms} + \beta_6 \times \texttt{Reached.on.Time\_Y.N}\end{aligned}$$

```
lm_1 <- lm(Cost_of_the_Product ~ Customer_care_calls + Customer_rating + Prior_purchases +
Discount_offered + Weight_in_gms + Reached.on.Time_Y.N, data)
summary(lm_1)
```

```
> summary(lm_1)

Call:
lm(formula = Cost_of_the_Product ~ Customer_care_calls + Customer_rating +
    Prior_purchases + Discount_offered + Weight_in_gms + Reached.on.Time_Y.N,
    data = dataset)

Residuals:
     Min       1Q   Median       3Q      Max
-142.628  -35.263    4.729   35.999  113.766

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)         177.572283   2.973160  59.725  < 2e-16 ***
Customer_care_calls  11.213230   0.408195  27.470  < 2e-16 ***
Customer_rating       0.177991   0.303270   0.587  0.55728
Prior_purchases       1.419818   0.291199   4.876  1.1e-06 ***
Discount_offered     -0.379061   0.031258 -12.127  < 2e-16 ***
Weight_in_gms        -0.003171   0.000311 -10.196  < 2e-16 ***
Reached.on.Time_Y.N  -3.089174   0.965250  -3.200  0.00138 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 44.95 on 10992 degrees of freedom
Multiple R-squared:  0.126,     Adjusted R-squared:  0.1255
F-statistic: 264.1 on 6 and 10992 DF,  p-value: < 2.2e-16
```

Hình 70: Kết quả thu được từ đoạn code R

Nhận xét:

Từ kết quả ta thu được:

$\beta_0 = 177.572283$; $\beta_1 = 11.213230$; $\beta_2 = 0.177991$; $\beta_3 = 1.419818$; $\beta_4 = -0.379061$; $\beta_5 = -0.003171$; $\beta_6 = -3.089174$.

Như vậy phương trình đường thẳng hồi quy là:

$\texttt{Cost\_of\_the\_Product} = 177.572283 + 11.213230 \times \texttt{Customer\_care\_calls} + 0.177991 \times \texttt{Customer\_rating} + 1.419818 \times \texttt{Prior\_purchases} + -0.379061 \times \texttt{Discount\_offered} + -0.003171 \times \texttt{Weight\_in\_gms} + -3.089174 \times \texttt{Reached.on.Time\_Y.N}$

Kiểm định các hệ số hồi quy:

- Giả thuyết H_0: Các hệ số hồi quy không có ý nghĩa thống kê.
- Giả thuyết H_1: Các hệ số hồi quy có ý nghĩa thống kê.

Vì p-value của biến Customer_rating lớn hơn mức ý nghĩa 5% nên ta chấp nhận H_0. Do đó ta loại biến này khỏi mô hình.

Ta tiếp tục xây dựng mô hình 2 từ mô hình 1 sau khi loại biến Customer_rating như sau.

```
lm_2 <- lm(Cost_of_the_Product ~ Customer_care_calls + Prior_purchases + Discount_offered +
Weight_in_gms + Reached.on.Time_Y.N, data)
summary(lm_2)
```

```
> summary(lm_2)

Call:
lm(formula = Cost_of_the_Product ~ Customer_care_calls + Prior_purchases +
    Discount_offered + Weight_in_gms + Reached.on.Time_Y.N, data = dataset)

Residuals:
     Min       1Q   Median       3Q      Max
-142.279  -35.365    4.691   36.000  113.594

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)       178.077868   2.845543  62.581  < 2e-16 ***
Customer_care_calls  11.215939   0.408156  27.480  < 2e-16 ***
Prior_purchases       1.421909   0.291168   4.883 1.06e-06 ***
Discount_offered     -0.379144   0.031256 -12.130  < 2e-16 ***
Weight_in_gms        -0.003170   0.000311 -10.194  < 2e-16 ***
Reached.on.Time_Y.N  -3.079662   0.965085  -3.191  0.00142 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 44.94 on 10993 degrees of freedom
Multiple R-squared:  0.126,     Adjusted R-squared:  0.1256
F-statistic: 316.8 on 5 and 10993 DF,  p-value: < 2.2e-16
```

Hình 71: Kết quả thu được từ đoạn code R

Ta thực hiện so sánh 2 mô hình như sau:

```
anova(lm_1, lm_2)
```

```
> anova(lm_1,lm_2)
Analysis of Variance Table

Model 1: Cost_of_the_Product ~ Customer_care_calls + Customer_rating +
    Prior_purchases + Discount_offered + Weight_in_gms + Reached.on.Time_Y.N
Model 2: Cost_of_the_Product ~ Customer_care_calls + Prior_purchases +
    Discount_offered + Weight_in_gms + Reached.on.Time_Y.N
  Res.Df      RSS Df Sum of Sq      F Pr(>F)
1  10992 22205435
2  10993 22206131 -1   -695.85 0.3445 0.5573
```

Hình 72: Kết quả thu được từ đoạn code R

- Giả thuyết H_0: Hai mô hình có hiệu quả giống nhau.
- Giả thuyết H_1: Hai mô hình hiệu quả khác nhau.

Vì `p-value = 0.5573`, lớn hơn mức ý nghĩa `5%` nên ta chấp nhận H_0, tức hai mô hình có hiệu quả giống nhau. Trong đó biến bỏ đi từ mô hình `1` là không có ý nghĩa nên ta chọn mô hình `2` vì hiệu quả hơn.

Như vậy phương trình đường thẳng hồi quy mới là:

`Cost_of_the_Product` $=$ `178.077868` $+$ `11.215939` $\times$ `Customer_care_calls` $+$ `1.421909` $\times$ `Prior_purchases` $+$ `-0.379144` $\times$ `Discount_offered` $+$ `-0.003170` $\times$ `Weight_in_gms` $+$ `-3.079662` $\times$ `Reached.on.Time_Y.N`

4.6.2 Phân tích tác động của các nhân tố lên sự biến thiên của giá tiền

Trước hết, ta thấy rằng `p_value` tương ứng với thống kê `F` $< 2.2e^{-16}$, có ý nghĩa rất cao. Điều này chỉ ra rằng, ít nhất một biến mô hình có ý nghĩa giải thích rất cao đến việc biến thiên của giá tiền.

Để xét ảnh hưởng của chúng ta xét từng biến độc lập của mô hình đã chọn. Xét hệ số β_i với `i` từ `0` đến `5` và `p-value` tương ứng, ta nhận thấy các biến `Customer_care_calls`, `Discount_offered` và `Weight_in_gms` $< 2.2e^{-16}$ cho thấy ảnh hưởng các biến này là rất lớn. Hệ số β_i cũng rất đáng được quan tâm khi ở biến `Customer_care_calls`, với mỗi cuộc gọi, giá cả lại tăng thêm 11.215939 (cho rằng các biến khác không đổi).

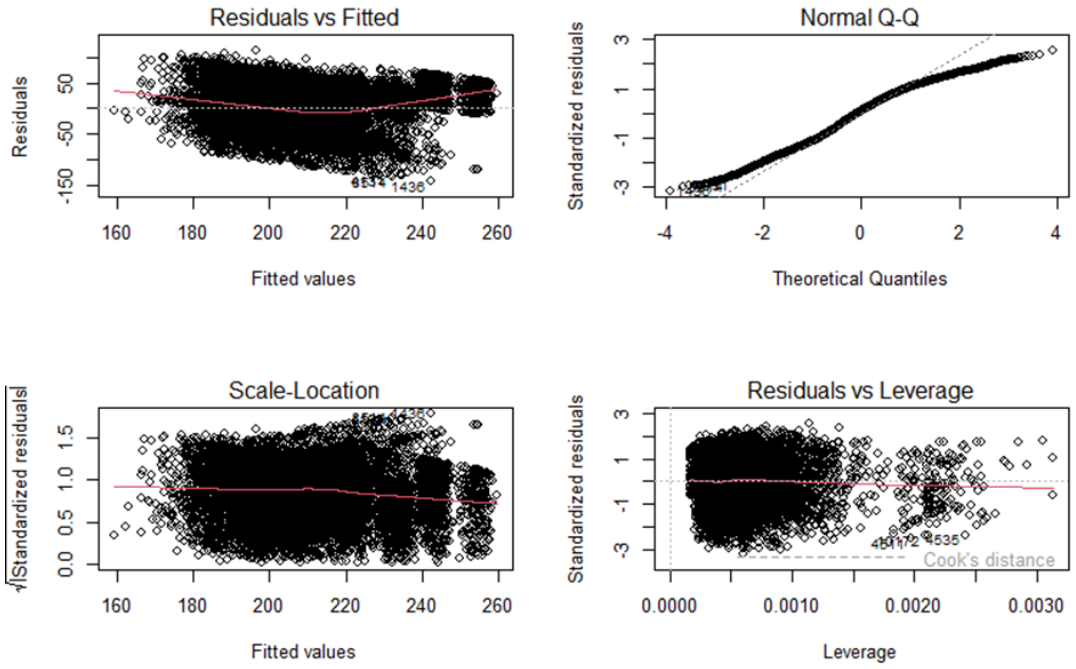
Hệ số R^2 hiệu chỉnh bằng `0.1256` nghĩa là chỉ có `12.56%` được giải thích bởi các biến độc lập.

Để kiểm tra tính giả định của mô hình hồi quy tuyến tính, chúng ta cần kiểm tra các điều kiện sau:

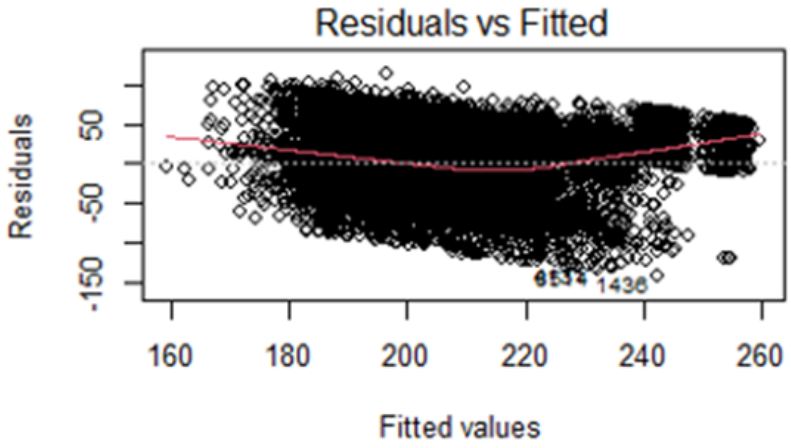
- Điều kiện độc lập tuyến tính giữa biến độc lập và biến phụ thuộc.
- Sai số có kỳ vọng bằng `0` và có phân phối chuẩn.
- Phương sai của sai số là hằng số.
- Các sai số độc lập với nhau.

Ta thực hiện phân tích giá trị thặng dư của mô hình như sau.

```
par(mfrow = c(2,2))
plot(lm_2)
```



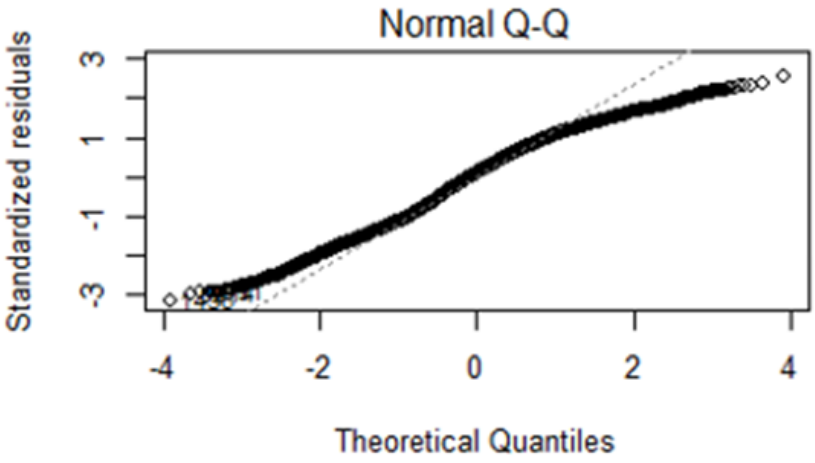
Hình 73: Kết quả thu được từ đoạn code R



Hình 74: Biểu đồ Residuals and Fitted

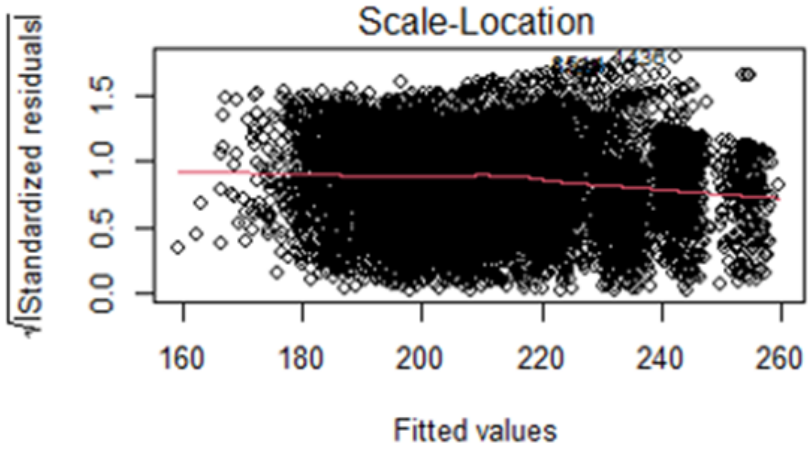
Đồ thị thứ nhất vẽ các sai số tương ứng với các giá trị dự báo, kiểm tra giả định tuyến tính của dữ liệu, giả định sai số có kỳ vọng bằng 0, giả định phương sai của sai số là hằng số. Dựa

trên đồ thị ta thấy, đường màu đỏ có sự biến thiên nên giả định tuyến tính của dữ liệu không thỏa mãn. Đường màu đỏ không nằm sát đường `y = 0` nên giả định sai số có kỳ vọng bằng `0` không thỏa mãn. Các sai số phân tán ngẫu nhiên dọc theo đường màu đỏ nên giả định phương sai các biến là hằng số thỏa mãn.



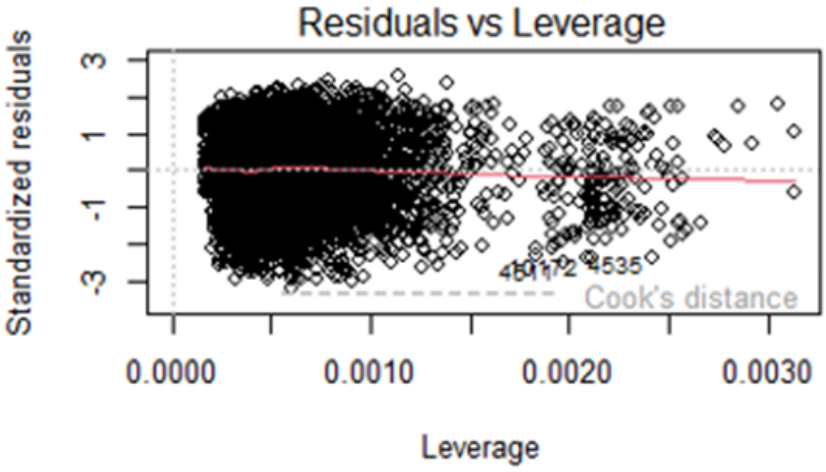
Hình 75: Biểu đồ `Q-Q plot`

Đồ thị thứ hai vẽ các sai số đã được chuẩn hoá, kiểm tra giả định sai số có phân phối chuẩn. Dựa trên đồ thị ta thấy, có nhiều điểm quan trắc lệch ra khỏi dường thẳng kì vọng phân phối chuẩn nên giả định sai số có phân phối chuẩn chưa thoả mãn.



Hình 76: Biểu đồ Scale - Location

Đồ thị thứ ba vẽ căn bậc hai của các sai số đã được chuẩn hoá, kiểm tra giả định phương sai các sai số là hằng số. Dựa vào đồ thị ta thấy, đường màu đỏ nằm ngang và các quan trắc phân tán ngẫu nhiên dọc theo đường màu đỏ nên giả định phương sai của các biến là các hằng số thỏa mãn.



Hình 77: Biểu đồ Residuals and Leverage

Đồ thị thứ tư chỉ ra các quan trắc có thể là các điểm có ảnh hưởng cao trong bộ dữ liệu. Trong đó các quan trắc 4511, 10172 và 4535 có thể là những điểm có ảnh hưởng cao. Tuy nhiên các điểm này không vượt khỏi đường cook nên chúng không thực sự có ảnh hưởng cao nên không cần loại bỏ chúng khi phân tích.

Lời kết

Qua dự án bài tập lớn của học kỳ này, nhóm chúng em không chỉ tích lũy được thêm các kinh nghiệm học thuật khi thực hiện các nội dung mà quý thầy đưa ra, mà còn luyện tập được khả năng giao tiếp, làm việc nhóm một cách hiệu quả, năng suất.

Tuy vậy, trong quá trình thực hiện dự án bài tập lớn, khó tránh khỏi các sai sót, ở điểm này nhóm rất mong quý thầy xem xét bỏ qua. Đồng thời, do trình độ lý luận cũng như kinh nghiệm thực tiễn còn hạn chế nên bài báo cáo không thể tránh khỏi những thiếu sót, nhóm chúng em rất mong nhận được ý kiến đóng góp từ quý thầy để nhóm chúng em có thể tích lũy được thêm nhiều kinh nghiệm và sẽ hoàn thành tốt hơn trong các dự án bài tập lớn sắp tới.

Để kết thúc dự án bài tập lớn này, nhóm xin một lần cuối cùng gửi lời cảm ơn chân thành nhất đến quý thầy, các anh chị, các bạn sinh viên trong cộng đồng sinh viên Đại học Quốc Gia – TP.HCM nói chung và sinh viên Đại học Bách Khoa nói riêng đã giúp nhóm hoàn thành tốt đẹp toàn bộ dự án bài tập lớn của bộ môn Xác suất Thống kê của học kỳ này.

Tài liệu tham khảo

[1] Nguyễn Đình Huy (Chủ biên), Đậu Thế Cấp, Lê Xuân Đại - Giáo trình Xác suất Thống kê, Nhà xuất bản Đại học Quốc gia TP. Hồ Chí Minh (2022)

[2] Nguyễn Văn Tuấn - Phân Tích Dữ Liệu Với R, Nhà xuất bản Tổng hợp Thành phô Hồ Chí Minh

[3] Sidney Siegel - Nonparametric statistics for the behavioral sciences (1956, McGraw-Hill)

[4] Dirk Metzler - Statistics for EES and others, Comparing more than two groups: Multiple testing, ANOVA and Kruskal - Wallis (2021)

[5] https://www.kaggle.com/datasets/prachi13/customer-analytics

[6] Paras Varshney - Q – Q Plots Explained (2020): https://towardsdatascience.com/q-q-plots-explained-5aa8495426c0

[7] PennState Eberly College of Science - Residuals vs. Fits Plot (2018): https://online.stat.psu.edu/stat462/node/117/