

DẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH



XÁC SUẤT VÀ THỐNG KÊ (MT2013)

BÀI TẬP LỚN CÁC BỘ PHẬN MÁY TÍNH (CPUs)

Giảng viên hướng dẫn: ThS. Nguyễn Kiều Dung

Tên nhóm thực hiện: MT53

STT	Họ tên SV	MSSV	Tên lớp	Tên khoa
1	Nguyễn Hữu Khánh	2211521	DL01	KH & KT Máy tính
2	Phan Châu Phong	2212563	DL01	KH & KT Máy tính
3	Ngô Minh Thái	2213106	DL02	KH & KT Máy tính
4	Dinh Ngọc Khánh	2311501	DL02	KH & KT Máy tính
5	Lê Quang Lợi	2113976	DL03	KH & KT Máy tính
6	Nguyễn Xuân Thành	2213145	DL03	KT Xây dựng
7	Trương Nhật Thành	2313145	DL03	KH & KT Máy tính

TP. HỒ CHÍ MINH, THÁNG 8/ 2024



Danh sách phân công công việc

Nhóm: MT53

STT	Họ và tên	Nhiệm vụ
1	Nguyễn Hữu Khánh	Bài toán phân tích phương sai & Thảo luận và mở rộng
2	Phan Châu Phong	Tiền xử lý số liệu
3	Ngô Minh Thái	Kiến thức nền & Bài toán hai mẫu
4	Dinh Ngọc Khánh	Bài toán một mẫu & Thảo luận và mở rộng
5	Lê Quang Lợi	Tổng quan dữ liệu
6	Nguyễn Xuân Thành	Thống kê mô tả
7	Trương Nhật Thành	Bài toán hồi quy tuyến tính & Thảo luận và mở rộng

Nhận xét từ giảng viên

STT	Họ và tên	Điểm số	Nhận xét
1	Nguyễn Hữu Khánh		
2	Phan Châu Phong		
3	Ngô Minh Thái		
4	Dinh Ngọc Khánh		
5	Lê Quang Lợi		
6	Nguyễn Xuân Thành		
7	Trương Nhật Thành		



Mục lục

1	Tổng quan dữ liệu	1
1.1	Ngữ cảnh dữ liệu	1
1.2	Giới thiệu đề tài	2
2	Kiến thức nền	3
2.1	Một số đặc trưng của mẫu	3
2.2	Kiểm định trung bình một mẫu	4
2.3	Kiểm định trung bình hai mẫu	5
2.4	Phân tích phương sai một yếu tố	7
2.5	Hồi quy tuyến tính đa biến	8
3	Tiền xử lý số liệu	11
3.1	Đọc dữ liệu	11
3.2	Xử lý định dạng dữ liệu	12
3.3	Xử lý dữ liệu khuyết	12
4	Thống kê mô tả	16
4.1	Thống kê các giá trị cơ bản	16
4.2	Dồ thị Histogram	18
4.3	Dồ thị Boxplot	19
4.4	Dồ thị Scatterplot	21
5	Thống kê suy diễn (mô hình chính)	23
5.1	Bài toán một mẫu	23
5.2	Bài toán hai mẫu	26
5.3	Bài toán phân tích phương sai	28
5.4	Bài toán hồi quy tuyến tính	33
6	Thảo luận và mở rộng	42
6.1	Kiểm định trung bình một mẫu	42
6.2	Kiểm định trung bình hai mẫu	42



6.3 Phân tích phương sai một yếu tố	43
6.4 Hồi quy tuyến tính bội	43
7 Nguồn dữ liệu và nguồn code	43



1 Tổng quan dữ liệu

1.1 Ngữ cảnh dữ liệu

`Intel CPUs.csv` là một tập dữ liệu bao gồm thông tin chi tiết của từng mẫu CPU (Central Processing Unit), từ các dòng sản phẩm phổ thông đến các dòng cao cấp như ngày phát hành, giá bán, tình trạng sản xuất... và các thông số kỹ thuật chi tiết của nó về kiến trúc, hiệu suất, đồ họa, độ phân giải...

Tập dữ liệu được tạo ra để sử dụng với nhiều mục đích khác nhau như:

- Hỗ trợ người dùng tham khảo và lựa chọn CPU phù hợp với nhu cầu;
- Hỗ trợ việc phân tích hiệu suất, phát triển các ứng dụng, công cụ góp phần cải tiến CPU;
- Hỗ trợ cung cấp dữ liệu cho việc phân tích xu hướng phát triển thị trường CPU hoặc mối quan hệ giữa các thông số.

Tập dữ liệu này chủ yếu tổng hợp thông tin từ Intel, Game-Debate, và các công ty tham gia sản xuất phần cứng máy tính.

```
> str(data)
'data.frame': 2283 obs. of 45 variables:
 $ Product_Collection : chr "7th Generation Intel® Core™ i7 F"
 $ Vertical_Segment    : chr "Mobile" "Mobile" "Mobile" "Deskt
 $ Processor_Number    : chr "i7-7Y75" "i5-8250U" "i7-8550U" "
 $ Status              : chr "Launched" "Launched" "Launched"
 $ Launch_Date         : chr "Q3'16" "Q3'17" "Q3'17" "Q1'12" .
 $ Lithography          : chr "14 nm" "14 nm" "14 nm" "32 nm" .
 $ Recommended_Customer_Price: chr "$393.00" "$297.00" "$409.00" .
 $ nb_of_Cores          : int 2 4 4 4 2 2 2 2 1 ...
 $ nb_of_Threads         : int 4 8 8 8 4 2 2 2 2 NA ...
 $ Processor_Base_Frequency: chr "1.30 GHz" "1.60 GHz" "1.80 GHz"
 $ Max_Turbo_Frequency   : chr "3.60 GHz" "3.40 GHz" "4.00 GHz"
 $ Cache                : chr "4 MB SmartCache" "6 MB SmartCach
 $ Bus_Speed             : chr "4 GT/s OPI" "4 GT/s OPI" "4 GT/s
 $ TDP                  : chr "4.5 W" "15 W" "15 W" "130 W" ...
 $ Embedded_Options_Available: chr "No" "No" "No" "No" ...
 $ Conflict_Free          : chr "Yes" "Yes" "Yes" NA ...
 $ Max_Memory_Size        : chr "16 GB" "32 GB" "32 GB" "64.23 GE
 $ Memory_Types           : chr "LPDDR3-1866, DDR3L-1600" "DDR4-2
 $ Max_nb_of_Memory_Channels: int 2 2 2 4 2 2 1 2 2 NA ...
```

Hình 1.1: Thống kê biến và giá trị quan trắc của file `Intel CPUs.csv`

Tập dữ liệu bao gồm 2,283 giá trị quan trắc (dòng dữ liệu) từ 45 biến, mỗi dòng đại diện cho một mẫu CPU khác nhau của Intel.

1.2 Giới thiệu đề tài

Trong thời đại công nghệ hiện nay, CPU đóng vai trò quan trọng như là "bộ não" của các hệ thống máy tính. Với sự phát triển không ngừng của ngành công nghiệp máy tính, thị trường CPU trở nên ngày càng đa dạng với nhiều sản phẩm khác nhau về tính năng và hiệu năng. Tuy nhiên, giá cả của CPU không chỉ đơn thuần được xác định bởi thương hiệu mà còn phụ thuộc vào nhiều thông số kỹ thuật và công nghệ mà nó sở hữu.

Bằng cách phân tích dữ liệu từ các mẫu CPU của Intel, nghiên cứu này không chỉ giúp người tiêu dùng hiểu rõ hơn về giá trị của các thông số kỹ thuật mà còn cung cấp thông tin hữu ích cho các nhà sản xuất và phân phối trong việc định giá sản phẩm. Đồng thời, nó cũng giúp các nhà phân tích thị trường nắm bắt được xu hướng giá cả và phát triển chiến lược cạnh tranh hiệu quả. Chính vì vậy nhóm chọn đề tài: **Các thông số của CPU ảnh hưởng tới giá cả của nó như thế nào?**

Nhóm lựa chọn từ tập dữ liệu 10 biến sau để phục vụ cho việc thống kê, phân tích liên quan đến đề tài:

STT	Tên biến	Loại biến	Đặc tả
1	Vertical_Segment	Định tính	Phân loại thiết bị theo thị trường mục tiêu và yêu cầu kỹ thuật cụ thể, từ đó hỗ trợ việc thiết kế, lựa chọn, và sử dụng sản phẩm phù hợp (loại thiết bị)
2	Lithography	Rời rạc	Công nghệ bán dẫn được sử dụng để sản xuất mạch tích hợp (kích thước transistor, đơn vị nm)
3	Recommended_Customer_Price	Liên tục	Giá bán lẻ đề xuất của CPU (đơn vị \$)
4	nb_of_Cores	Rời rạc	Số lượng lõi xử lý độc lập trong một bộ vi xử lý (số lõi)
5	nb_of_Threads	Rời rạc	Số lượng luồng xử lý mà một CPU có thể thực hiện đồng thời (số luồng)

STT	Tên biến	Loại biến	Đặc tả
6	Processor_Base_Frequency	Liên tục	Tốc độ xung nhịp cơ bản của CPU khi hoạt động trong điều kiện tối ưu nhất (tần số xung nhịp, đơn vị GHz)
7	Cache	Liên tục	Là một khu vực bộ nhớ nhanh nằm trên CPU, giúp tăng tốc độ truy cập và cải thiện hiệu suất tổng thể của hệ thống (dung lượng bộ nhớ đệm, đơn vị MB)
8	Max_Memory_Size	Liên tục	Dung lượng tối đa của bộ nhớ RAM mà một hệ thống máy tính hoặc thiết bị có thể hỗ trợ (dung lượng tối đa của bộ nhớ, đơn vị GB)
9	Max_nb_of_Memory_Channels	Rời rạc	Số lượng kênh bộ nhớ mà một CPU hoặc bo mạch chủ có thể hỗ trợ (số kênh)
10	Max_Memory_Bandwidth	Liên tục	Tốc độ tối đa mà một CPU có thể đọc hoặc lưu trữ dữ liệu vào bộ nhớ bán dẫn (băng thông bộ nhớ tối đa, đơn vị GB/s)

2 Kiến thức nền

2.1 Một số đặc trưng của mẫu

$$* \text{Trung bình mẫu: } \bar{x} = \frac{1}{n} \sum_{i=1}^n x_i = \frac{1}{n} \sum_{i=1}^k n_i x_i$$

$$* \text{Phương sai mẫu: } s^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2$$

* **Trung vị:** Giả sử mẫu có kích thước n , các phần tử sắp xếp tăng dần.

Nếu $n = 2k + 1$ thì trung vị mẫu là giá trị x_{k+1} .

Nếu $n = 2k$ thì trung vị mẫu là giá trị $\frac{x_k + x_{k+1}}{2}$.



* **Tứ phân vị:** Tứ phân vị chia mẫu dữ liệu thành bốn tập có số phần tử bằng nhau. Trung vị chia mẫu dữ liệu đã sắp xếp thành hai tập có số phần tử bằng nhau là Q_2 . Trung vị của tập dữ liệu nhỏ hơn là Q_1 (tứ phân vị dưới) và trung vị của tập dữ liệu lớn hơn là Q_3 (tứ phân vị trên).

Dộ trải giữa (khoảng tứ phân vị) $IQR = R_Q = Q_3 - Q_1$.

* **Điểm ngoại lai:**

Là các phần tử của mẫu có giá trị nằm ngoài khoảng $(Q_1 - 1,5IQR; Q_3 + 1,5IQR)$.

2.2 Kiểm định trung bình một mẫu

Bài toán:

Giả sử tổng thể có trung bình là μ .

Mẫu có kích thước là n , trung bình mẫu là \bar{x} , phương sai mẫu là s^2 .

Hãy kiểm định giả thuyết $H_0 : \mu = \mu_0$ với mức ý nghĩa α cho trước.

* **Quy trình kiểm định:**

Bước 1: Phát biểu giả thuyết H_0 và đối thuyết H_1 .

Bước 2: Tính giá trị kiểm định thống kê dựa trên việc giả sử rằng H_0 đúng.

Bước 3: Xác định miền bác bỏ dựa trên phân phối của tiêu chuẩn kiểm định thống kê.

Bước 4: Dựa ra kết luận bác bỏ H_0 nếu giá trị kiểm định thống kê thuộc miền bác bỏ.

* **Các dạng của bài toán kiểm định trung bình một mẫu:**

a) *Tổng thể có phân phối chuẩn, đã biết phương sai σ^2*

H_0	H_1	Tiêu chuẩn kiểm định	Miền bác bỏ
$\mu = \mu_0$	$\mu \neq \mu_0$	$Z_{qs} = \frac{\bar{x} - \mu_0}{\frac{s}{\sqrt{n}}}$	$(-\infty; -Z_{\alpha/2}) \cup (Z_{\alpha/2}; +\infty)$
	$\mu < \mu_0$		$(-\infty; -Z_\alpha)$
	$\mu > \mu_0$		$(Z_\alpha; +\infty)$

b) *Tổng thể có phân phối chuẩn, chưa biết phương sai σ^2 , $n < 30$*

H_0	H_1	Tiêu chuẩn kiểm định	Miền bác bỏ
$\mu = \mu_0$	$\mu \neq \mu_0$	$t_{qs} = \frac{\bar{x} - \mu_0}{\frac{s}{\sqrt{n}}}$	$(-\infty; -t_{\alpha/2;n-1}) \cup (t_{\alpha/2;n-1}; +\infty)$
	$\mu < \mu_0$		$(-\infty; -t_{\alpha;n-1})$
	$\mu > \mu_0$		$(t_{\alpha;n-1}; +\infty)$



c) *Tổng thể có phân phối tùy ý, $n \geq 30$*

H_0	H_1	Tiêu chuẩn kiểm định	Miền bác bỏ
$\mu = \mu_0$	$\mu \neq \mu_0$	$Z_{qs} = \frac{\bar{x} - \mu_0}{\sigma / \sqrt{n}}$ (nếu chưa biết σ thì thay bằng s)	$(-\infty; -Z_{\alpha/2}) \cup (Z_{\alpha/2}; +\infty)$
	$\mu < \mu_0$		$(-\infty; -Z_\alpha)$
	$\mu > \mu_0$		$(Z_\alpha; +\infty)$

2.3 Kiểm định trung bình hai mẫu

Bài toán:

Giả sử tổng thể I và tổng thể II có trung bình lần lượt là μ_1 và μ_2 .

Từ tổng thể I có mẫu kích thước là n_1 , trung bình mẫu là \bar{x}_1 , phương sai mẫu là s_1^2 .

Từ tổng thể II có mẫu kích thước là n_2 , trung bình mẫu là \bar{x}_2 , phương sai mẫu là s_2^2 .

Hãy kiểm định giả thuyết $H_0 : \mu_1 = \mu_2$ với mức ý nghĩa α cho trước.

* **Quy trình kiểm định:** Như quy trình kiểm định trung bình một mẫu.

* **Các dạng của bài toán kiểm định trung bình hai mẫu:**

a) Hai mẫu độc lập; tổng thể I và II có phân phối chuẩn, đã biết phương sai σ_1^2 và σ_2^2

H_0	H_1	Tiêu chuẩn kiểm định	Miền bác bỏ
$\mu_1 = \mu_2$	$\mu_1 \neq \mu_2$	$Z_{qs} = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}}$	$(-\infty; -Z_{\alpha/2}) \cup (Z_{\alpha/2}; +\infty)$
	$\mu_1 < \mu_2$		$(-\infty; -Z_\alpha)$
	$\mu_1 > \mu_2$		$(Z_\alpha; +\infty)$

b) Hai mẫu độc lập; tổng thể I và II có phân phối chuẩn, chưa biết phương sai σ_1^2 và σ_2^2 , $\sigma_1^2 = \sigma_2^2$

H_0	H_1	Tiêu chuẩn kiểm định	Miền bắc bỏ
Dấu hiệu quy ước để nhận biết từ mẫu $\frac{s_1}{s_2} \in \left[\frac{1}{2}; 2 \right]$			
$\mu_1 = \mu_2$	$\mu_1 \neq \mu_2$	$s_p^2 = \frac{(n_1 - 1)s_1^2 + (n_2 - 1)s_2^2}{n_1 + n_2 - 2}$	$(-\infty; -t_{\alpha/2;n_1+n_2-2}) \cup (t_{\alpha/2;n_1+n_2-2}; +\infty)$
	$\mu_1 < \mu_2$	$t_{qs} = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{s_p^2 \left(\frac{1}{n_1} + \frac{1}{n_2} \right)}}$	$(-\infty; -t_{\alpha;n_1+n_2-2})$
	$\mu_1 > \mu_2$		$(t_{\alpha;n_1+n_2-2}; +\infty)$

c) Hai mẫu độc lập; tổng thể I và II có phân phối chuẩn, chưa biết phương sai σ_1^2 và σ_2^2 , $\sigma_1^2 \neq \sigma_2^2$

H_0	H_1	Tiêu chuẩn kiểm định	Miền bắc bỏ
Dấu hiệu quy ước để nhận biết từ mẫu $\frac{s_1}{s_2} \notin \left[\frac{1}{2}; 2 \right]$			
$\mu_1 = \mu_2$	$\mu_1 \neq \mu_2$	$v = \frac{(s_1^2/n_1 + s_2^2/n_2)^2}{\frac{(s_1^2/n_1)^2}{n_1 - 1} + \frac{(s_2^2/n_2)^2}{n_2 - 1}}$	$(-\infty; -t_{\alpha/2;v}) \cup (t_{\alpha/2;v}; +\infty)$
	$\mu_1 < \mu_2$	v làm tròn thành số nguyên	$(-\infty; -t_{\alpha;v})$
	$\mu_1 > \mu_2$	$t_{qs} = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}}$	$(t_{\alpha;v}; +\infty)$

d) Hai mẫu độc lập; tổng thể I và II có phân phối tùy ý, n_1 và $n_2 \geq 30$

H_0	H_1	Tiêu chuẩn kiểm định	Miền bắc bỏ
$\mu_1 = \mu_2$	$\mu_1 \neq \mu_2$	$Z_{qs} = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}}$	$(-\infty; -Z_{\alpha/2}) \cup (Z_{\alpha/2}; +\infty)$
	$\mu_1 < \mu_2$	(nếu chưa biết σ_1 và σ_2 thì thay bằng s_1 và s_2)	$(-\infty; -Z_{\alpha})$
	$\mu_1 > \mu_2$		$(Z_{\alpha}; +\infty)$



e) Hai mẫu tương ứng theo cặp; tổng thể I và II có phân phối chuẩn, chưa biết phương sai σ_1^2 và σ_2^2

H_0	H_1	Tiêu chuẩn kiểm định	Miền bác bỏ
$\mu_1 = \mu_2$	$\mu_1 \neq \mu_2$	Đặt $D = X_1 - X_2$ $t_{qs} = \frac{\bar{d}}{s_D} \sqrt{n}$	$(-\infty; -t_{\alpha/2;n-1}) \cup (t_{\alpha/2;n-1}; +\infty)$
	$\mu_1 < \mu_2$		$(-\infty; -t_{\alpha;n-1})$
	$\mu_1 > \mu_2$		$(t_{\alpha;n-1}; +\infty)$

f) Hai mẫu tương ứng theo cặp; tổng thể I và II có phân phối tùy ý, n_1 và $n_2 \geq 30$

H_0	H_1	Tiêu chuẩn kiểm định	Miền bác bỏ
$\mu_1 = \mu_2$	$\mu_1 \neq \mu_2$	Đặt $D = X_1 - X_2$ $t_{qs} = \frac{\bar{d}}{s_D} \sqrt{n}$ (nếu chưa biết σ_D thay bằng s_D)	$(-\infty; -Z_{\alpha/2}) \cup (Z_{\alpha/2}; +\infty)$
	$\mu_1 < \mu_2$		$(-\infty; -Z_\alpha)$
	$\mu_1 > \mu_2$		$(Z_\alpha; +\infty)$

2.4 Phân tích phương sai một yếu tố

Phân tích phương sai (ANOVA) là so sánh trung bình của nhiều nhóm (tổng thể) dựa trên các giá trị trung bình của các mẫu quan sát từ các nhóm này và thông qua kiểm định giả thuyết để kết luận về sự bằng nhau của các trung bình tổng thể này.

Phân tích phương sai một yếu tố là phân tích ảnh hưởng của một yếu tố nguyên nhân ảnh hưởng đến một yếu tố kết quả đang nghiên cứu.

* Giả thiết của bài toán ANOVA một yếu tố:

- Các tổng thể có phân phối chuẩn, số lượng tổng thể thường lớn hơn hoặc bằng 3.
- Phương sai của các tổng thể bằng nhau.
- Các mẫu quan sát từ các tổng thể được lấy độc lập.

* Các bước thực hiện:

Bước 1: Đặt giả thuyết kiểm định.

Giả thuyết $H_0: \mu_1 = \mu_2 = \dots = \mu_k$, Đối thuyết $H_1: \exists \mu_i \neq \mu_j$ (với $i \neq j$)



Bước 2: Tính giá trị kiểm định thống kê.

- Tính các trung bình: $\bar{x}_1, \bar{x}_2, \dots, \bar{x}_k; \bar{x}$

- Tính các tổng bình phương:

$$\begin{aligned} SSB &= \sum_{i=1}^k n_i (\bar{x}_i - \bar{x})^2 \\ SSW &= \sum_{i=1}^k \sum_{j=1}^{n_i} (x_{ij} - \bar{x}_i)^2 = \sum_{i=1}^k (n_i - 1) s_i^2 \\ SST &= \sum_{i=1}^k \sum_{j=1}^{n_i} (x_{ij} - \bar{x})^2 = (N - 1)s^2 = SSB + SSW \end{aligned}$$

* Chú ý: Ý nghĩa của các tổng bình phương:

- SSB: Phần biến thiên do các mức độ của yếu tố đang xem xét tạo ra. (Nghiệm thực)

- SSW: Phần biến thiên do các yếu tố nào đó không được đề cập đến tạo ra. (Sai số)

- SST: Tổng các biến thiên do tất cả các yếu tố tạo ra.

- Tính các phương sai (bình phương trung bình): $MSB = \frac{SSB}{k-1}; MSW = \frac{SSW}{N-k}$

- Giá trị kiểm định: $F = \frac{MSB}{MSW}$

Bước 3: Xác định miền bắc bỏ. RR = (F_{\alpha; k-1; N-k}; +\infty)

Bước 4: Kết luận.

Nếu $F \in RR$, kết luận bác bỏ giả thuyết H_0 . Nghĩa là ở độ tin cậy α thì trung bình của các nhóm khác nhau, hay trung bình của các nhóm phụ thuộc vào điều kiện đang xem xét.

Hệ số xác định $R^2 = \frac{SSB}{SST} \cdot 100\%$ đo mức độ ảnh hưởng của yếu tố đang xem xét đối với sự biến động của các giá trị của biến ngẫu nhiên X quanh giá trị của nó. R^2 càng lớn thì mô hình càng gọi là thích hợp.

2.5 Hồi quy tuyến tính đa biến

Phân tích hồi quy là nghiên cứu mối liên hệ phụ thuộc của một biến (biến phụ thuộc, biến được giải thích) vào một hay nhiều biến độc lập khác (các biến độc lập, các biến giải thích) với ý tưởng ước lượng hoặc dự đoán giá trị trung bình (tổng thể) của biến phụ thuộc trên cơ sở các giá trị biết trước (trong mẫu) của các biến độc lập.

*** Mô hình hồi quy tuyến tính đa biến:**

Mô hình hồi quy tuyến tính đa biến (còn được gọi là hồi quy tuyến tính bội) sử dụng nhiều biến giải thích để dự đoán giá trị của biến phụ thuộc.



a) ***Phương trình hồi quy tuyến tính tổng thể đa biến*** với k biến giải thích và n quan sát có dạng:
$$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \dots + \beta_k X_{ki} + \varepsilon_i \quad (\text{với } i = 1, 2, \dots, n)$$

Trong đó:

β_0 : hệ số tung độ gốc (hệ số chặn);

β_j (với $j = 1, 2, \dots, k$): hệ số hồi quy riêng (hệ số độ dốc) của Y theo biến X_j giữ các biến còn lại không đổi. Nếu X_j tăng 1 đơn vị thì giá trị kỳ vọng của Y sẽ tăng β_j đơn vị và ngược lại;

ε_i : các sai số ngẫu nhiên, được giả sử là độc lập và có cùng phân phối chuẩn $N(0, \sigma^2)$.

b) ***Phương trình hồi quy tuyến tính mẫu đa biến*** với k biến giải thích và n quan sát có dạng:
$$\hat{Y}_i = b_0 + b_1 X_{1i} + b_2 X_{2i} + \dots + b_k X_{ki} \quad (\text{với } i = 1, 2, \dots, n)$$

Trong đó:

b_0 : hệ số tung độ gốc ước lượng của β_0 căn cứ trên dữ liệu mẫu;

b_j (với $j = 1, 2, \dots, k$): hệ số độ dốc ước lượng của β_j căn cứ trên dữ liệu mẫu.

* Các bước thực hiện:

Bước 1: Dùng phần mềm để tính toán các hệ số hồi quy mẫu và các số thống kê cần thiết sử dụng để đánh giá mô hình.

- Tính tổng bình phương toàn phần SST , tổng bình phương hồi quy SSR và tổng bình phương sai số SSE .

- Tính hệ số xác định bội (Multiple R-Squared):
$$R^2 = \frac{SSR}{SST}$$

Hệ số xác định bội giải thích trong 100% sự biến động của Y so với trung bình của nó thì có bao nhiêu % là do các biến X gây ra.

- Tính hệ số xác định hiệu chỉnh (Adjusted R-squared):

$$R_a^2 = 1 - (1 - R^2) \frac{N - 1}{N - k - 1} \quad (\text{với } N \text{ là cỡ mẫu})$$

Đây là hệ số cho chúng ta biết mức độ cải tiến của phương sai phần dư (residual variance) do các yếu tố có mặt trong mô hình tuyến tính. Hệ số này không khác mấy so với hệ số xác định bội.

- Tính độ lệch chuẩn ước lượng:
$$\hat{\sigma} = \sqrt{\frac{SSE}{N - k - 1}}$$

Độ lệch chuẩn ước lượng đo lường sự phân tán của các giá trị quan sát xung quanh các giá trị dự đoán của biến phụ thuộc.

- Tính bình phương trung bình hồi quy MSR và bình phương trung bình sai số MSE .

- Tính giá trị kiểm định
$$F = \frac{MSR}{MSE}$$

Bước 2: Đánh giá sự phù hợp của mô hình.

Một số phương pháp thống kê để tiến hành đánh giá sự phù hợp của mô hình là:

a) Kiểm định F

Kiểm định F được dùng để xác định có tồn tại mối liên hệ có ý nghĩa giữa biến phụ thuộc và toàn bộ các biến độc lập, được xem như kiểm định ý nghĩa tổng thể.

Xây dựng giả thuyết $H_0 : R^2 = 0$ và đối thuyết $H_1 : R^2 \neq 0$

Bản chất của giả thuyết H_0 có ý nghĩa là mô hình hồi quy đang xây dựng các biến độc lập không giải thích được cho sự biến động của biến phụ thuộc. Nói một cách khác là tất cả các hệ số hồi quy đều không có ý nghĩa.

Vậy nên có thể viết giả thuyết $H_0 : \beta_j = 0$ và đối thuyết $H_1 : \beta_j \neq 0$ (với $j = 1, 2, \dots, k$).

Với mức ý nghĩa α nếu $F > F_{\alpha;k;N-k-1}$ thì bác bỏ giả thuyết H_0 , kết luận toàn bộ mô hình có ý nghĩa về mặt thống kê.

b) Kiểm định t

Kiểm định t được dùng để xác định xem từng biến độc lập riêng có ý nghĩa hay không, được xem như kiểm định ý nghĩa riêng lẻ.

Xét từng biến độc lập X_j (với $j = 1, 2, \dots, k$).

Xây dựng giả thuyết $H_0 : \beta_j = 0$ và đối thuyết $H_1 : \beta_j \neq 0$

$$\text{Giá trị kiểm định } t = \frac{b_j}{\hat{\sigma}_{b_j}} \text{ với } \hat{\sigma}_{b_j} = \sqrt{\frac{\hat{\sigma}^2}{\sum(X_j - \bar{X})^2}}$$

Với mức ý nghĩa α nếu $|t| < t_{\alpha/2;N-k-1}$ thì chấp nhận giả thuyết H_0 , kết luận biến độc lập X_j này không có khả năng giải thích.

Bước 3: Sau khi hài lòng với độ phù hợp của mô hình, diễn dịch ý nghĩa của các hệ số hồi quy.

Xem lại phương trình hồi quy mẫu $\hat{Y}_i = b_0 + b_1 X_{1i} + b_2 X_{2i} + \dots + b_k X_{ki}$

Hệ số chặn b_0 chính là giá trị ước lượng \hat{Y}_i khi $X_j = 0$

Hệ số độ dốc b_j cho biết khi X_j tăng thêm 1 đơn vị, trong điều kiện các biến khác không đổi, thì giá trị ước lượng \hat{Y}_i sẽ tăng thêm trung bình b_j đơn vị.

Bước 4: Sử dụng mô hình để dự đoán giá trị trung bình của biến phụ thuộc.

Phương trình hồi quy tổng thể có dạng $Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \dots + \beta_k X_{ki} + \varepsilon_i$ được ước lượng bởi mô hình hồi quy mẫu $\hat{Y}_i = b_0 + b_1 X_{1i} + b_2 X_{2i} + \dots + b_k X_{ki}$. Như vậy muốn dự đoán giá trị trung bình của Y với điều kiện X_j đã có giá trị cụ thể, thế các giá trị đó



vào phương trình đường hồi quy mẫu để ra kết quả dự đoán.

Bước 5: Kiểm tra sự phù hợp khi lựa chọn mô hình hồi quy tuyến tính.

Phần dư (độ lệch ngẫu nhiên) $\varepsilon_i = (Y_i - \hat{Y}_i)$ là khoảng chênh lệch theo phương thẳng đứng giữa giá trị quan sát với giá trị dự đoán của biến phụ thuộc.

Tính đúng đắn của mô hình hồi quy được xem xét bằng cách vẽ đồ thị phần dư lần lượt theo giá trị dự đoán của biến phụ thuộc và từng biến độc lập. Nếu các đồ thị không thể hiện một hình dạng rõ ràng nào của các chấm phân tán thì mô hình khá phù hợp.

Bước 6: Chẩn đoán sự vi phạm các giả thuyết trong mô hình, nếu điều này xảy ra tiến hành khắc phục.

* Các giả định cần kiểm tra sau khi chọn mô hình hồi quy tuyến tính đa biến:

- Các sai số ngẫu nhiên (phần dư) ε_i độc lập. Khi kiểm tra, điều kiện này sẽ được thay thế bằng việc kiểm tra giả định các X_i không xảy ra quan hệ đa cộng tuyến.
- Các sai số ngẫu nhiên ε_i là độc lập và có cùng phân phối chuẩn $N(0, \sigma^2)$ với σ không đổi.

3 Tiết xử lý dữ liệu

3.1 Đọc dữ liệu

Để đọc dữ liệu trong tập dữ liệu Intel CPUs.csv, ta làm theo các bước sau:

- Đầu tiên, ta định nghĩa một vector naStrings chứa các chuỗi được xác định là dữ liệu bị thiếu (NA), bao gồm "\n-", (gạch ngang kết thúc dòng), "" (chuỗi rỗng) và "N/A".
- Tiếp theo, ta đọc dữ liệu từ file Intel CPUs.csv. Tham số header = TRUE xác định rằng file csv có hàng đầu tiên chứa tên cột. Tham số na.strings = naStrings chỉ định rằng các giá trị trong naStrings sẽ được xem là dữ liệu khuyết và được đánh dấu là NA trong bộ dữ liệu.

```

1 naStrings = c("\n-", "", "N/A")
2 data <- read.csv("Intel CPUs.csv", header = TRUE, na.strings
= naStrings)

```

Dữ liệu từ file Intel CPUs.csv được lưu vào khung dữ liệu (data frame) data, với các giá trị dữ liệu bị thiếu được đánh dấu là NA. Bằng cách nhập thêm lệnh dưới đây, dữ liệu từ khung dữ liệu data được đưa vào khung dữ liệu mới main_Factors, chỉ chứa các giá trị của 10 biến dữ liệu ta đã chọn ở phần **Tổng quan dữ liệu**.

```
main_Factors <- data [, c("Vertical_Segment", "Lithography",
  "Recommended_Customer_Price", "nb_of_Cores", "nb_of_
  Threads", "Processor_Base_Frequency", "Cache", "Max_Memory
  _Size", "Max_nb_of_Memory_Channels", "Max_Memory_Bandwidth
  ") ]
```

```
> head(main_Factors)
  Vertical_Segment Lithography Recommended_Customer_Price nb_of_Cores nb_of_Threads Processor_Base_Frequency
1       Mobile      14 nm             $393.00          2              4            1.30 GHz
2       Mobile      14 nm             $297.00          4              8            1.60 GHz
3       Mobile      14 nm             $409.00          4              8            1.80 GHz
4      Desktop      32 nm             $305.00          4              8            3.60 GHz
5       Mobile      14 nm             $281.00          2              4            1.20 GHz
6       Mobile      14 nm             $107.00          2              2            1.50 GHz
  Cache Max_Memory_Size Max_nb_of_Memory_Channels Max_Memory_Bandwidth
1   4 MB SmartCache        16 GB                  2           29.8 GB/s
2   6 MB SmartCache        32 GB                  2           34.1 GB/s
3   8 MB SmartCache        32 GB                  2           34.1 GB/s
4  10 MB SmartCache       64.23 GB                 4           51.2 GB/s
5   4 MB SmartCache        16 GB                  2           29.8 GB/s
6     2 MB               16 GB                  2           25.6 GB/s
```

Hình 3.1: Kết quả 6 hàng đầu tiên của khung dữ liệu main_Factors

3.2 Xử lý định dạng dữ liệu

Bảng dữ liệu mới được tạo vẫn còn đơn vị nên sẽ có định dạng chuỗi (string), dữ liệu vẫn chưa hoàn toàn sạch. Vì vậy, ta cần tiến hành xử lý định dạng dữ liệu cho từng biến.

- Vertical_Segment: Giữ nguyên định dạng ký tự.
- nb_of_Cores, nb_of_Threads, Max_nb_of_Memory_Channels: Chuyển đổi từ định dạng chuỗi sang định dạng số.
- Các biến còn lại: Loại bỏ các ký tự không phải số, chuyển đổi từ định dạng chuỗi sang định dạng số. Trường hợp biến Recommended_Customer_Price xuất hiện giá nằm giữa hai giá trị (ví dụ: \$70.00 - \$77.00), ta loại bỏ ký tự không phải số và sau đó tính trung bình của chúng. Ngoài ra ta cần thống nhất toàn bộ giá trị của một số biến về cùng 1 đơn vị, cụ thể là Processor_Base_Frequency theo đơn vị GHz, Cache theo đơn vị MB và Max_Memory_Size theo đơn vị GB.

3.3 Xử lý dữ liệu khuyết

Khi tuỳ chỉnh số lượng hàng cần đọc, ta có thể phát hiện ra một vài dữ liệu khuyết.



	Vertical_Segment	Lithography	Recommended_Customer	Price	nb_of_Cores	nb_of_Threads
1	Mobile	14		393	2	4
2	Mobile	14		297	4	8
3	Mobile	14		409	4	8
4	Desktop	32		305	4	8
5	Mobile	14		281	2	4
6	Mobile	14		107	2	2
7	Mobile	22		NA	2	2
8	Desktop	22		NA	2	2
9	Desktop	22		42	2	2
10	Mobile	90		NA	1	NA
11	Mobile	22		134	2	2
12	Mobile	90		NA	1	NA
	Processor_Base_Frequency	Cache	Max_Memory_Size	Max_nb_of_Memory_Channels	Max_Memory_Bandwidth	
1	1.30	4	16.00	2	29.8	
2	1.60	6	32.00	2	34.1	
3	1.80	8	32.00	2	34.1	
4	3.60	10	64.23	4	51.2	
5	1.20	4	16.00	2	29.8	
6	1.50	2	16.00	2	25.6	
7	1.46	1	4.00	1	NA	
8	2.41	12	8.00	2	NA	
9	2.60	2	32.00	2	21.0	
10	2.80	12	NA	NA	NA	
11	2.40	2	32.00	2	25.6	
12	1.30	22	NA	NA	NA	

Hình 3.2: Kết quả lệnh `head(main_Factors, 12)` sau khi xử lý định dạng dữ liệu

Ta cần tạo bảng thống kê phân tích dữ liệu khuyết trong bộ dữ liệu.

- Đầu tiên, ta sử dụng hàm `summarise_all` để tính tổng số lượng các giá trị bị khuyết cho từng biến.
- Tiếp theo, ta sử dụng hàm `gather` để chuyển đổi dữ liệu từ dạng wide sang long, với các cột là `Column` và `Missing_Count`.
- Cuối cùng, tính tổng số dòng và tỷ lệ phần trăm giá trị bị khuyết. Ta bổ sung thêm các cột `Total_Count` và `Missing_Percentage` để biểu diễn tỷ lệ phần trăm dữ liệu bị khuyết.

```

1 missing_data_stats <- main_Factors %>%
2   summarise_all(~sum(is.na(.))) %>%
3   gather(key = "Column", value = "Missing_Count") %>%
4   mutate>Total_Count = nrow(main_Factors),
5   Missing_Percentage = (Missing_Count / Total_Count) * 100)
6 print(missing_data_stats)

```

```

> # Hiển thị bảng thống kê dữ liệu khuyết lần 1
> print(missing_data_stats)
   Column Missing_Count Total_Count Missing_Percentage
1 Vertical_Segment          0        2283      0.0000000
2 Lithography                 71        2283      3.1099431
3 Recommended_Customer_Price    982        2283      43.0135786
4 nb_of_Cores                  0        2283      0.0000000
5 nb_of_Threads                856        2283      37.4945247
6 Processor_Base_Frequency     18        2283      0.7884363
7 Cache                         12        2283      0.5256242
8 Max_Memory_Size              880        2283      38.5457731
9 Max_nb_of_Memory_Channels   869        2283      38.0639509
10 Max_Memory_Bandwidth        1136       2283      49.7590889

```

Hình 3.3: Bảng thống kê phân tích dữ liệu khuyết trong 10 biến của bộ dữ liệu

Dựa trên phân tích dữ liệu khuyết trong tập dữ liệu, ta có các nhận xét và quyết định xử lý như sau:

- 2 biến `Vertical_Segment` và `nb_of_Cores` không có dữ liệu khuyết.
- Không cần tác động tới 2 biến này.
- 3 biến `Cache` (0.53%), `Processor_Base_Frequency` (0.79%) và `Lithography` (3.11%) có tỷ lệ dữ liệu khuyết thấp.
 - Ta sẽ tiến hành loại bỏ các dòng bị khuyết của những biến này, đảm bảo tính chính xác của phân tích.
 - Các biến còn lại có tỷ lệ dữ liệu khuyết cao, từ 37.49% đến 49.76%.
 - Ta có thể tiến hành thay thế giá trị trung bình hoặc trung vị tương ứng vào các dòng bị khuyết. Trong trường hợp này, nhóm thống nhất 2 biến `Recommended_Customer_Price` và `Max_Memory_Bandwidth` sẽ sử dụng giá trị trung bình để thay thế, còn 3 biến `nb_of_Threads`, `Max_nb_of_Memory_Channels` và `Max_Memory_Size` sẽ sử dụng giá trị trung vị để tránh trường hợp thay giá trị trung bình vào ra số lẻ, không phù hợp với thực tế.



```
> head(main_Factors,12)
   Vertical_Segment Lithography Recommended_Customer_Price nb_of_Cores nb_of_Threads
1           Mobile        14            393.0000          2             4
2           Mobile        14            297.0000          4             8
3           Mobile        14            409.0000          4             8
4         Desktop        32            305.0000          4             8
5           Mobile        14            281.0000          2             4
6           Mobile        14            107.0000          2             2
7           Mobile        22            856.7264          2             2
8         Desktop        22            856.7264          2             2
9         Desktop        22            42.0000          2             2
10          Mobile        90            856.7264          1             4
11          Mobile        22            134.0000          2             2
12          Mobile        90            856.7264          1             4
   Processor_Base_Frequency Cache Max_Memory_Size Max_nb_of_Memory_Channels Max_Memory_Bandwidth
1                 1.30       4          16.00                  2          29.80000
2                 1.60       6          32.00                  2          34.10000
3                 1.80       8          32.00                  2          34.10000
4                 3.60      10          64.23                  4          51.20000
5                 1.20       4          16.00                  2          29.80000
6                 1.50       2          16.00                  2          25.60000
7                 1.46       1          4.00                   1          35.56762
8                 2.41      12          8.00                   2          35.56762
9                 2.60       2          32.00                  2          21.00000
10                2.80      12          32.00                  2          35.56762
11                2.40       2          32.00                  2          25.60000
12                1.30      22          32.00                  2          35.56762
```

Hình 3.4: Kết quả lệnh `head(main_Factors, 12)` sau khi xử lý dữ liệu khuyết

Sau đó ta thử thống kê dữ liệu khuyết một lần nữa, và kết quả hoàn toàn khả quan.

```
> # Hiển thị bảng thống kê dữ liệu khuyết lần 2
> print(missing_data_stats)
   Column Missing_Count Total_Count Missing_Percentage
1 Vertical_Segment          0        2194              0
2 Lithography                 0        2194              0
3 Recommended_Customer_Price 0        2194              0
4 nb_of_Cores                 0        2194              0
5 nb_of_Threads                0        2194              0
6 Processor_Base_Frequency    0        2194              0
7 Cache                         0        2194              0
8 Max_Memory_Size               0        2194              0
9 Max_nb_of_Memory_Channels    0        2194              0
10 Max_Memory_Bandwidth          0        2194              0
```

Hình 3.5: Bảng thống kê phân tích dữ liệu khuyết sau khi xử lý dữ liệu khuyết

Ta có thể kết luận rằng dữ liệu trong khung dữ liệu `main_Factors` đã được làm sạch hoàn toàn, với thông kê biến và giá trị quan trắc như Hình 3.6.



```
> str(main_Factors)
'data.frame': 2194 obs. of 10 variables:
 $ Vertical_Segment      : chr "Mobile" "Mobile" "Mobile" ...
 $ Lithography             : num 14 14 14 32 14 14 22 22 ...
 $ Recommended_Customer_Price: num 393 297 409 305 281 ...
 $ nb_of_Cores             : num 2 4 4 4 2 2 2 2 2 1 ...
 $ nb_of_Threads            : num 4 8 8 8 4 2 2 2 2 4 ...
 $ Processor_Base_Frequency: num 1.3 1.6 1.8 3.6 1.2 1.5 ...
 $ Cache                   : num 4 6 8 10 4 2 1 12 2 12 ...
 $ Max_Memory_Size          : num 16 32 32 64.2 16 ...
 $ Max_nb_of_Memory_Channels: num 2 2 2 4 2 2 1 2 2 2 ...
 $ Max_Memory_Bandwidth     : num 29.8 34.1 34.1 51.2 29.8
```

Hình 3.6: Thống kê biến và giá trị quan trắc của `main_Factors` sau bước tiền xử lý

4 Thống kê mô tả

4.1 Thống kê các giá trị cơ bản

Ta tiến hành thống kê các giá trị trung bình, độ lệch chuẩn, giá trị nhỏ nhất (GTNN), giá trị lớn nhất (GTLN), tứ phân vị dưới (Q1), trung vị, tứ phân vị trên (Q3) và xuất kết quả dưới dạng bảng cho các biến liên tục, bao gồm 5 biến sau: `Recommended_Customer_Price`, `Processor_Base_Frequency`, `Cache`, `Max_Memory_Size` và `Max_Memory_Bandwidth`.

row.names	TrungBinh	DoLechChuan	GTNN	GTLN	Q1	TrungVi	Q3
1 Recommended_Customer_Price	856.726380	1111.6501994	9.620000	13011.0	268.25	856.72638	856.72638
2 Processor_Base_Frequency	2.273449	0.7761004	0.300000	4.3	1.70	2.30000	2.83000
3 Cache	21.619033	40.9624391	0.015625	362.0	3.00	8.00000	22.00000
4 Max_Memory_Size	185.802912	465.6467564	1.000000	4198.4	32.00	32.00000	64.00000
5 Max_Memory_Bandwidth	35.567616	22.3676812	1.600000	352.0	25.60	35.56762	35.56762

Hình 4.1: Giá trị thống kê cho các biến liên tục

Dựa vào kết quả trên, ta có một vài nhận xét như sau:

- Trung bình một bộ CPU mua trên thị trường sẽ có giá khoảng 856.73\$, với tần số xung nhịp 2.27GHz, dung lượng bộ nhớ đệm 21.62MB, dung lượng bộ nhớ tối đa 185.8GB và băng thông bộ nhớ tối đa 35.57GB/s.



– Giá bán trung bình của một bộ CPU là 856.73\$ và có độ lệch chuẩn là 1111.65\$. Giá trị nhỏ nhất và lớn nhất lần lượt là 9.62\$ và 13011.00\$. Số liệu này cho thấy mức độ biến động lớn trong giá bán của CPU, kèm theo đó độ chênh lệch giữa GTLN và GTNN cũng rất lớn.

→ *Ta có thể kết luận rằng thị trường CPU rất đa dạng về phân khúc khách hàng, có thể đáp ứng được nhiều khả năng chi trả của người mua.*

– Về các thông số khác như:

+ Tần số xung nhịp: dao động trong khoảng 0.3 – 4.3 GHz. Số liệu của trung vị, tứ phân vị trên và dưới cho ta biết được trong khung dữ liệu mẫu có 75% giá trị lớn hơn 1.7 GHz, có 50% giá trị lớn hơn 2.3 GHz và có 25% giá trị lớn hơn 2.83 GHz. Độ lệch chuẩn là 0.78 GHz. Số liệu này cho thấy phân phối của dữ liệu khá đều.

+ Dung lượng bộ nhớ đệm: dao động trong khoảng 0.016 – 362.000 MB. Trong khung dữ liệu mẫu có 75% giá trị lớn hơn 3MB, có 50% giá trị lớn hơn 8MB và có 25% giá trị lớn hơn 22MB. Độ lệch chuẩn là 9.39MB. Số liệu này cho thấy phân phối của dữ liệu không đều, có thể lệch về phía trái.

+ Dung lượng bộ nhớ tối đa: dao động trong khoảng 1.0 – 4198.4 GB. Trong khung dữ liệu mẫu có 75% giá trị lớn hơn 32GB và có 50% giá trị lớn hơn 64GB. Độ lệch chuẩn là 465.65GB. Vì tứ phân vị dưới Q1 bằng trung vị và khoảng cách với tứ phân vị trên Q3 không quá lớn nên phân phối của dữ liệu có thể lệch về phía trái, và phân tán rất mạnh ở nửa trên 64GB.

+ Băng thông bộ nhớ tối đa: dao động trong khoảng 1.6 – 352 GB/s. Trong khung dữ liệu mẫu có 75% giá trị lớn hơn 25.6GB/s và có 50% giá trị lớn hơn 35.57GB/s. Độ lệch chuẩn là 22.37GB/s. Vì tứ phân vị trên Q3 bằng trung vị nên phân phối của dữ liệu có thể lệch về phía phải, và phân tán rất mạnh ở nửa trên 35.57GB/s.

→ *Có thể thấy, thị trường CPU có nhiều loại sản phẩm phục vụ cho các phân khúc khách hàng khác nhau. Từ các CPU với các thông số thấp, từ GTNN đến Q1, phù hợp cho các máy phục vụ nhu cầu cá nhân đến các CPU mạnh mẽ, với thông số trong khoảng từ Q3 đến GTLN, được dùng trong các server, các siêu máy tính.*

Tiếp theo ta phân tích các biến rác, bao gồm các biến `Lithography`, `nb_of_Cores`, `nb_of_Threads` và `Max_nb_of_Memory_Channels`.

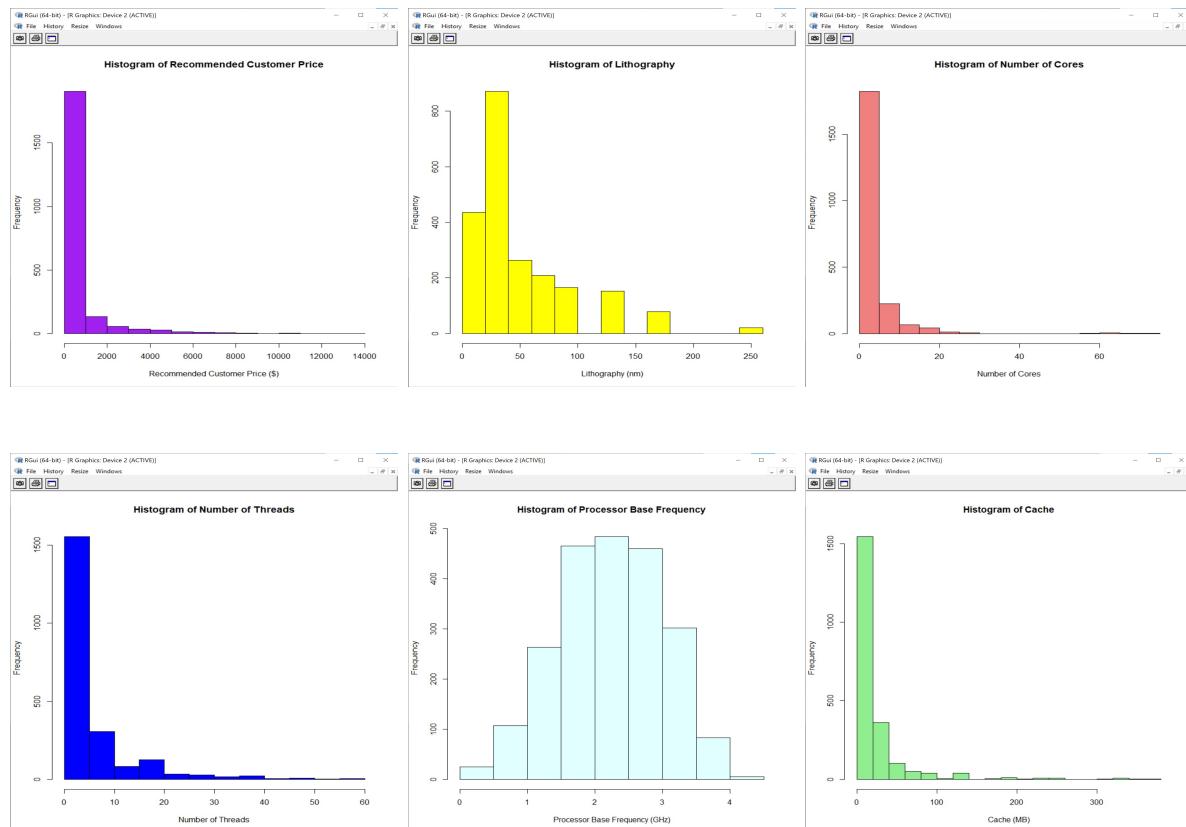
row.names	TrungBinh	DoLechChuan	GTNN	GTLN	TrungVi	GTriXuatHienNhiieuNhat
1 Lithography	49.168186	45.422177	14	250	32	22
2 Cores	4.147220	6.418788	1	72	2	2
3 Threads	7.042844	7.640699	1	56	4	4
4 Maximum Memory Channels	2.396992	1.212720	1	16	2	2

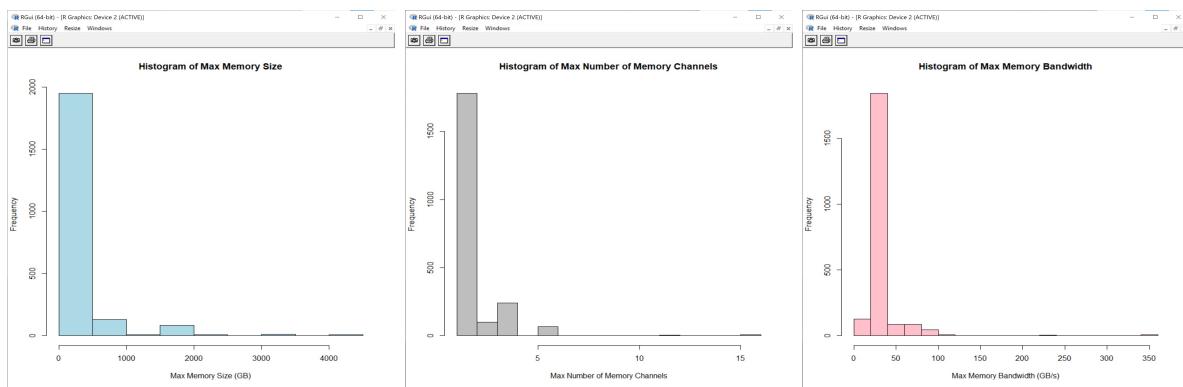
Hình 4.2: Giá trị thống kê cho các biến rời rạc

Nhận xét: Một CPU trung bình được bán trên thị trường sẽ có lithography (kích thước transistor) 49.17nm, 4 lõi, 7 luồng và tối đa 2 kenh bộ nhớ. Các CPU được bán nhiều nhất trên thị trường hiện nay có thông số lithography 22nm, 2 lõi, 4 luồng và tối đa 2 kenh bộ nhớ.

Độ lệch chuẩn của 3 biến Lithography, nb_of_Cores, nb_of_Threads là rất lớn so với giá trị trung bình, kèm theo đó cả 4 biến đều có giá trị trung vị gần với GTNN và cách rất xa với GTLN. Từ đây, ta có thể phỏng đoán rằng các giá trị của 4 biến này có mức độ biến động (sự phân tán) rất lớn và dữ liệu sẽ có sự hiện diện của rất nhiều điểm ngoại lai.

4.2 Đồ thị Histogram



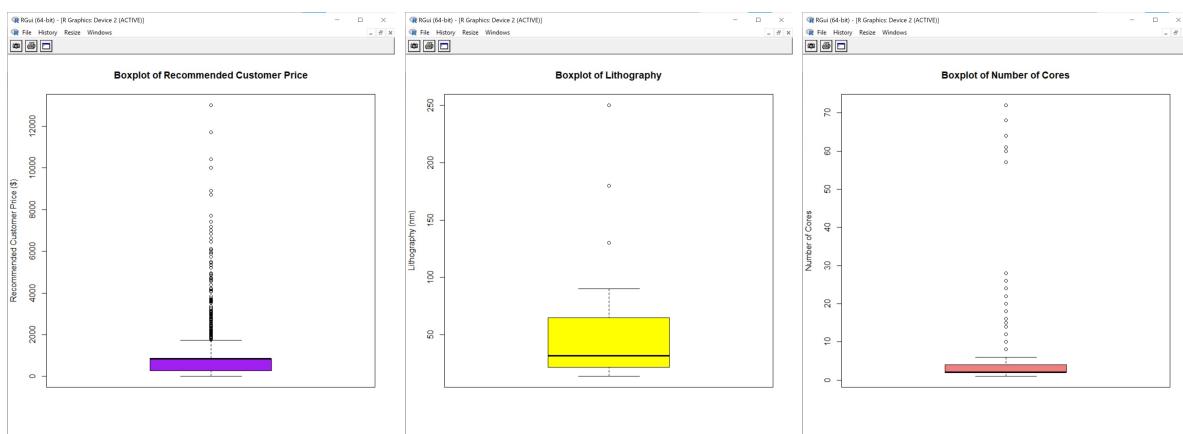


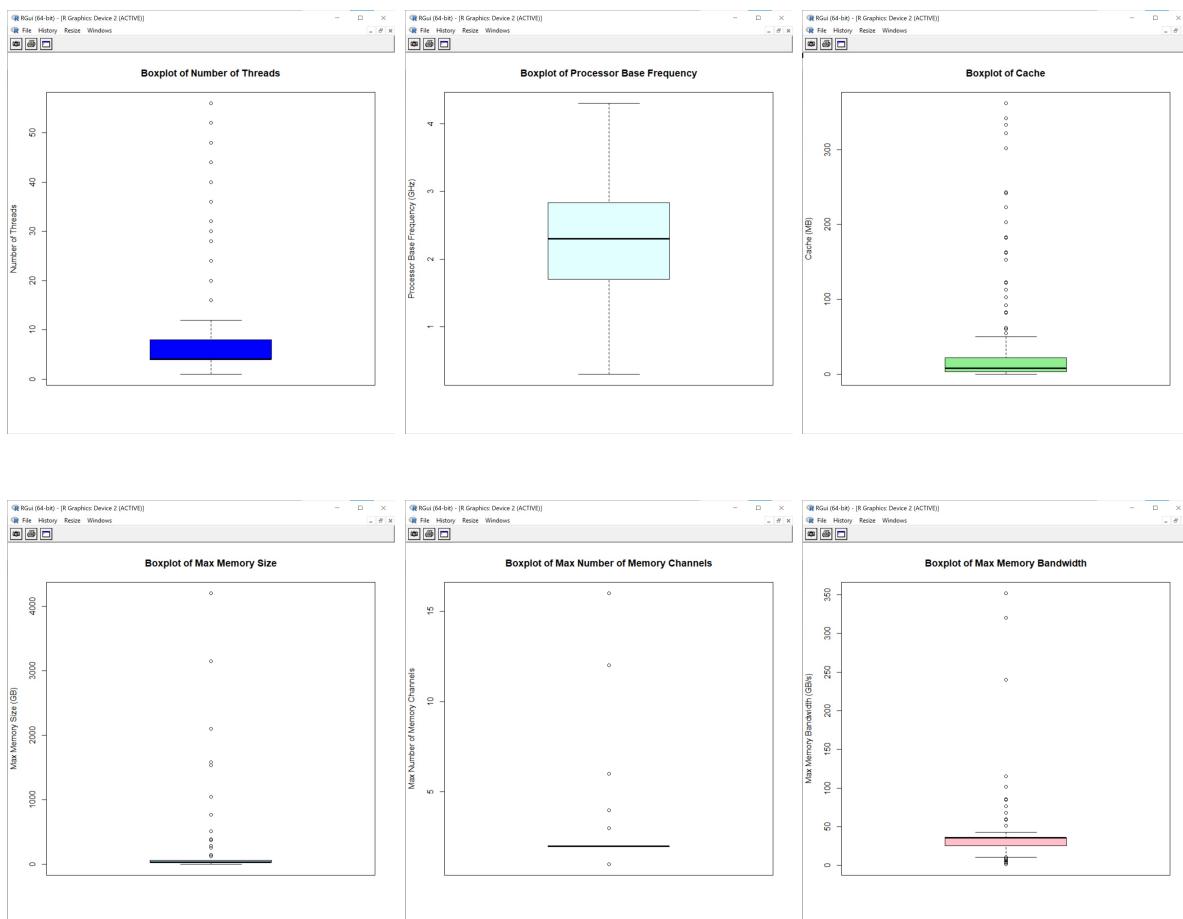
Hình 4.3: Đồ thị Histogram của 9 biến được chọn

Nhận xét: Ta có thể thấy các biến có sự phân bố không đồng đều và chủ yếu lệch về phía trái khi phần lớn số lượng CPU được sản xuất đều có các thông số xấp xỉ giá trị trung bình, ngoại trừ Processor_Base_Frequency.

Về hình dáng, đồ thị biến Processor_Base_Frequency có dạng hình chuông, nên có thể biến này tuân theo phân phối chuẩn. 2 biến Lithography và Max_Memory_Bandwidth có hình dáng khá giống đường cong Fisher. Các biến còn lại đều có hình dáng khá giống đường cong phân phối mũ, nhất là các biến Recommended_Customer_Price nb_of_Cores và Cache.

4.3 Đồ thị Boxplot





Hình 4.4: Đồ thị Boxplot của 9 biến được chọn

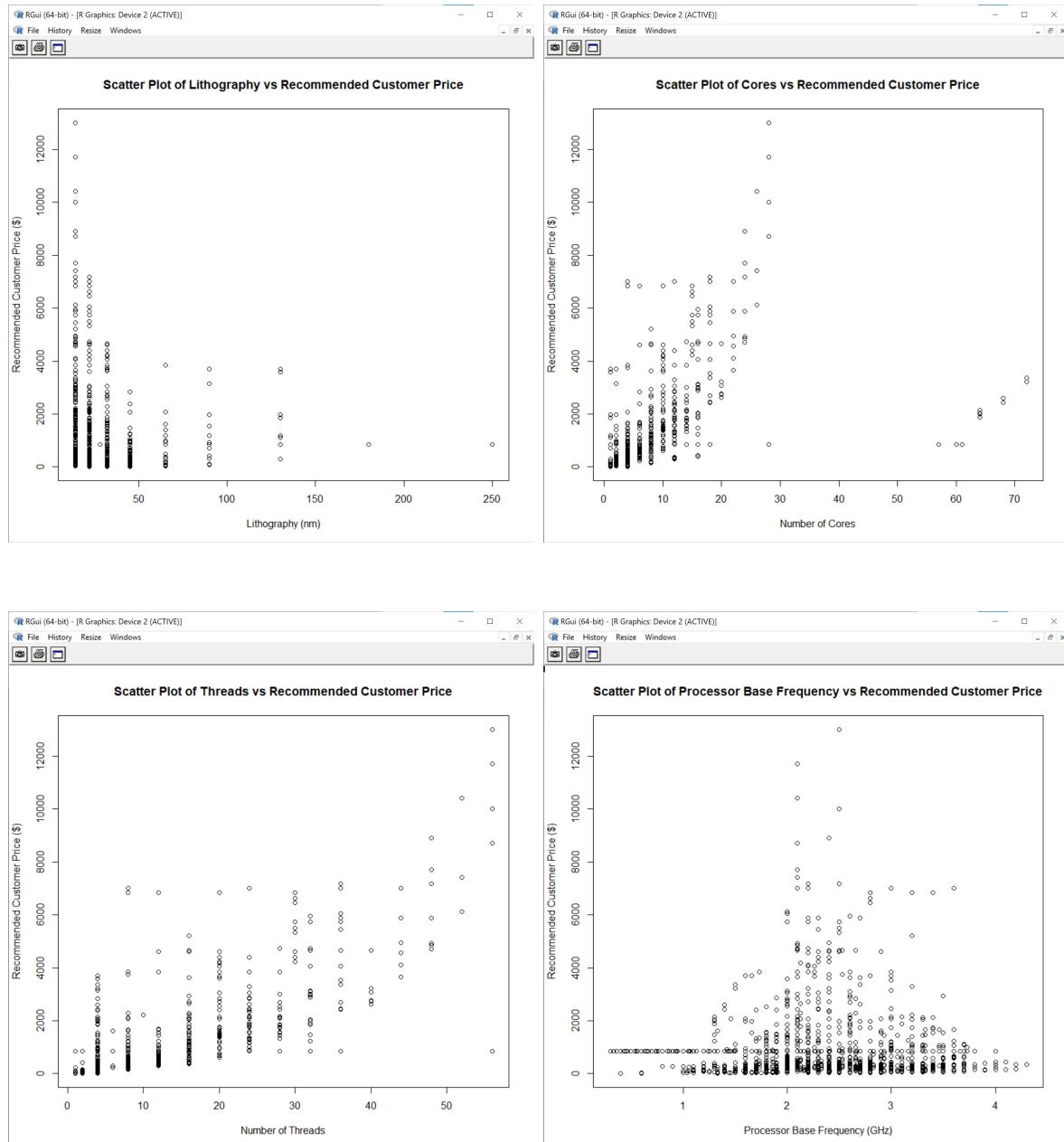
Nhận xét: Ngoài đồ thị của biến Processor_Base_Frequency không có điểm ngoại lai và 2 biến Lithography, Max_nb_of_Memory_Channels rất ít điểm ngoại lai ra, ta thấy các đồ thị còn lại chứa rất nhiều điểm ngoại lai, điều đó cho thấy 3 biến ta kể trên có tính ổn định và chất lượng cao.

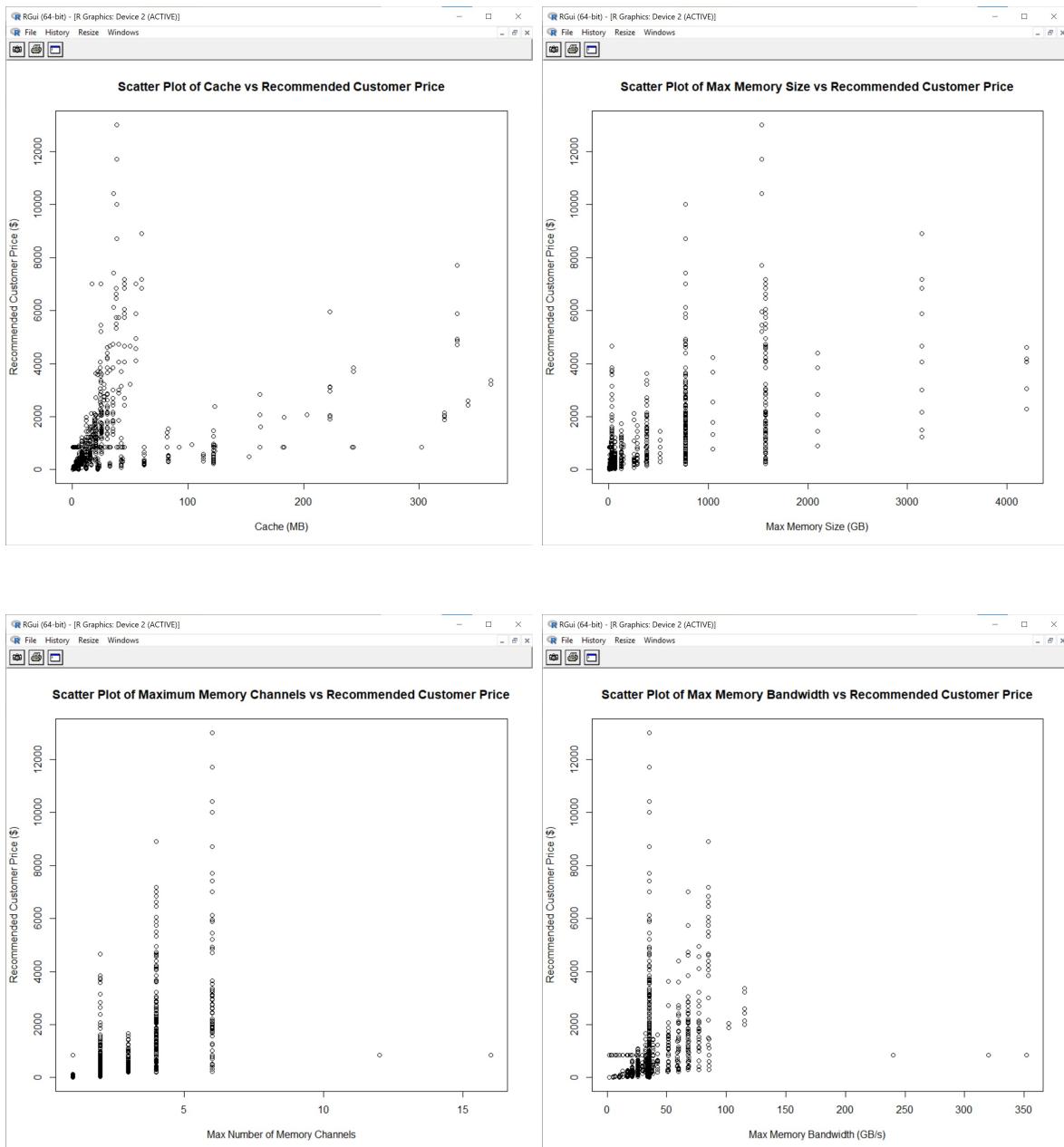
Ngoài đồ thị của biến Processor_Base_Frequency có phần hộp lớn và ở vị trí giữa, 8 biến còn lại phần hộp đều ở mức vừa đến rất nhỏ và lệch hẳn về phía dưới đồ thị, đặc biệt 2 biến Max_Memory_Size và Max_nb_of_Memory_Channels phần hộp nhỏ đến mức nhìn như 1 đường thẳng. Điều này cho thấy đa phần các giá trị gần giá trị trung bình chiếm số lượng áp đảo.

Đồ thị của biến Processor_Base_Frequency có phần râu dài đồng đều hai bên, chứng tỏ dữ liệu của biến này phân phối đều. Đồ thị của biến nb_of_Threads có phần râu không dài bằng nhưng tương đối đồng đều hai bên, từ những nhận xét trên chứng tỏ dữ liệu của biến này phân phối đều nhưng lệch hẳn về phía trái. Các biến khác như Recommended_Customer_Price, nb_of_Cores, Cache và Max_Memory_Bandwidth đều

có phần râu hẹp và có hai bên không đồng đều, càng chứng tỏ giá trị của chúng chỉ được phân bổ chủ yếu về một phía.

4.4 Đồ thị Scatterplot





Hình 4.5: Đồ thị Scatterplot của 8 biến (trục tung) so với biến Recommended_Customer_Price (trục hoành)

Nhận xét: Nhìn chung ta có thể thấy mối tương quan dương khá yếu giữa biến Recommended_Customer_Price và các biến như nb_of_Cores, nb_of_Threads, Cache, Max_Memory_Bandwidth khi giá trị của trục hoành tăng dần đến sự gia tăng giá trị của trục tung, mặc dù vẫn có nhiều điểm ngoại lệ. Các biến còn lại dường như không cho thấy sự liên hệ tuyến tính với biến Recommended_Customer_Price khi một giá trị ở trục hoành có thể cho ra nhiều giá trị ở trục tung.

5 Thống kê suy diễn (mô hình chính)

5.1 Bài toán một mẫu

Đề: Sau khi nghiên cứu tập dữ liệu Intel_CPU.csv, bạn Ngọc Khánh cho rằng những CPU sở hữu tốc độ băng thông cao (từ 85 GB/s trở lên) có giá bán trung bình là \$3000. Với mức ý nghĩa 5%, kiểm tra xem phát biểu của bạn đúng hay sai?

Giải:

Gọi μ là giá bán trung bình của một CPU có tốc độ băng thông lớn hơn hoặc bằng 85 GB/s.

Bước 1: Xây dựng giả thuyết.

Giả thuyết $H_0 : \mu = 3000$, Đối thuyết $H_1 : \mu \neq 3000$

Bước 2: Thu thập dữ liệu từ file Intel CPUs.csv.

Ta thực hiện lệnh lấy mẫu với điều kiện lọc Max_Memory_Bandwidth ≥ 85 như sau:

```
1 HighBandwidthVertical <- main_Factors %>%
2 filter(Max_Memory_Bandwidth >= 85) %>%
3 select(Max_Memory_Bandwidth, Recommended_Customer_Price)
```

Qua đó, ta có dữ liệu của các thiết bị có tốc độ băng thông từ 85 GB/s.

RGui (64-bit) - [R Console]		
File	Edit	View
Misc	Packages	Windows
> print(HighBandwidthVertical)		
	Max_Memory_Bandwidth	Recommended_Customer_Price
1	115.2	3368.0000
2	240.0	856.7264
3	240.0	856.7264
4	85.3	856.7264
5	85.3	1440.0000
6	85.3	1113.0000
7	85.3	835.0000
8	85.3	617.0000
9	85.3	444.0000
10	85.3	294.0000

Hình 5.1: Dữ liệu 10 dòng đầu trong HighBandwidthVertical

Bước 3: Xác định phương pháp kiểm định.

Trước tiên, ta áp dụng kiểm định Shapiro-Wilk để kiểm tra dữ liệu bài toán có tuân theo phân phối chuẩn hay không bằng lệnh sau:

```

> # Kiểm định phân phối chuẩn Shapiro-Wilk
> shapiro.test(HighBandwidthVertical$Recommended_Customer_Price)

Shapiro-Wilk normality test

data: HighBandwidthVertical$Recommended_Customer_Price
W = 0.91614, p-value = 0.000481
  
```

Hình 5.2: Kết quả kiểm định Shapiro-Wilk

Với giá trị p-value nhỏ hơn 0.05, ta có thể nhận xét rằng phân phối giá bán lẻ của HighBandwidthVertical không tuân theo phân phối chuẩn.

Tiếp theo, ta xác định kích thước mẫu bằng lệnh:

```
length(HighBandwidthVertical$Recommended_Customer_Price)
```

Kết quả kích thước mẫu chạy được là 61.

Từ đó, ta có thể kết luận hướng giải bài toán thuộc dạng **tổng thể có phân phối tùy ý, kích thước mẫu lớn hơn 30**. Ta sử dụng phương pháp kiểm định Z_{test} .

Bước 4: Tính giá trị kiểm định và xác định miền bắc bỏ.

Đầu tiên, để tìm giá trị kiểm định, ta có thể giải bằng 1 trong 2 cách sau:

- *Cách 1: Tính bằng công thức và tra bảng.* Thực hiện các lệnh để tìm giá trị trung bình mẫu \bar{X} và độ lệch chuẩn mẫu s của HighBandwidthVertical.

```

> mean(HighBandwidthVertical$Recommended_Customer_Price)
[1] 3606.484
> sd(HighBandwidthVertical$Recommended_Customer_Price)
[1] 2371.682
  
```

Hình 5.3: Kết quả thực hiện lệnh



Với các kết quả trên, ta có thể tính giá trị kiểm định như sau:

$$Z_{qs} = \frac{\bar{X} - \mu_0}{\frac{s}{\sqrt{n}}} = \frac{3606.484 - 3000}{\frac{2371.682}{\sqrt{61}}} \approx 1.9972$$

Tra bảng với $\alpha = 0.05$ ta tìm được miền bắc bỏ

$$RR = (-\infty, -Z_{\alpha/2}) \cup (Z_{\alpha/2}, +\infty) = (-\infty, -1.96) \cup (1.96, +\infty)$$

- *Cách 2: Sử dụng lệnh `t.test` trong RStudio.* Ta tiến hành kiểm định Z_{test} trên phần mềm RStudio và thu được kết quả như hình sau.

```

> # Tính giá trị z cho kiểm định một mẫu
> z <- qnorm(1-0.05/2)
> cat("Miền bắc bỏ: RR = (-\infty; -z) \cup (z; +\infty)\n")
Miền bắc bỏ: RR = (-\infty; -1.959964 ) \cup ( 1.959964 ; +\infty)
>
> # Thực hiện Welch's t-test
> t.test(HighBandwidthVertical$Recommended_Customer_Price, mu = 3000)

One Sample t-test

data: HighBandwidthVertical$Recommended_Customer_Price
t = 1.9972, df = 60, p-value = 0.05034
alternative hypothesis: true mean is not equal to 3000
95 percent confidence interval:
 2999.068 4213.900
sample estimates:
mean of x
 3606.484

```

Hình 5.4: Kết quả kiểm định trung bình một mẫu bằng Z_{test}

- * **Chú ý:** t_{test} được dùng trên RStudio đã được viết lại dưới dạng Z_{test} , giá trị của t chính là Z_{qs} .

Từ kết quả chạy trên RStudio, ta có miền bắc bỏ $RR = (-\infty, -1.959964) \cup (1.959964, +\infty)$ và $Z_{qs} = 1.9972$, hoàn toàn trùng khớp với kết quả ở *Cách 1*.

Bước 5: Kết luận.

Ta thấy $Z_{qs} \in RR$, dẫn đến bác bỏ giả thuyết H_0 và công nhận H_1 đúng. Tức là với mức ý nghĩa 5% thì giá bán trung bình của một CPU có tốc độ băng thông cao (từ 85GB/s trở lên) không phải là \$3000. Phát biểu của bạn Ngọc Khánh là sai.



5.2 Bài toán hai mẫu

Đề: Bạn Minh Thái dựa vào tập dữ liệu Intel CPUs.csv cho rằng giá bán lẻ đề xuất của CPU cho loại thiết bị Desktop sẽ đắt hơn các CPU cho loại thiết bị Mobile. Với mức ý nghĩa 5%, kiểm tra xem phát biểu của bạn đúng hay sai?

Giải:

Gọi μ_1 là giá bán lẻ trung bình của một CPU cho loại thiết bị Desktop.

Gọi μ_2 là giá bán lẻ trung bình của một CPU cho loại thiết bị Mobile.

Bước 1: Xây dựng giả thuyết.

Giả thuyết $H_0 : \mu_1 = \mu_2$ hoặc $\mu_1 \leq \mu_2$, Dối thuyết $H_1 : \mu_1 > \mu_2$

Bước 2: Thu thập dữ liệu từ file Intel CPUs.csv.

Ta thực hiện lấy hai mẫu bằng cách tạo ra hai khung dữ liệu mới Desktop và Mobile, lọc dữ liệu theo điều kiện Vertical_Segment là "Desktop" hoặc "Mobile".

```

1 Desktop <- main_Factors$Recommended_Customer_Price
2 [grep("Desktop", main_Factors$Vertical_Segment,
3 ignore.case = TRUE)]
4 Mobile <- main_Factors$Recommended_Customer_Price
5 [grep("Mobile", main_Factors$Vertical_Segment,
6 ignore.case = TRUE)]
```

Dữ liệu của mỗi mẫu sẽ được lưu dưới dạng 1 dãy số liệu thỏa mãn các điều kiện lọc trên.

<pre>> print(Desktop) [1] 305.0000 856.7264 [4] 94.0000 94.0000 [7] 161.0000 856.7264 [10] 856.7264 856.7264 [13] 856.7264 856.7264 [16] 856.7264 856.7264 [19] 856.7264 856.7264 [22] 856.7264 856.7264 [25] 856.7264 856.7264 [28] 856.7264 856.7264 [31] 64.0000 856.7264 [34] 856.7264 856.7264 [37] 73.5000 856.7264</pre>	<pre>> print(Mobile) [1] 393.0000 297.0000 [4] 281.0000 107.0000 [7] 856.7264 134.0000 [10] 161.0000 161.0000 [13] 161.0000 856.7264 [16] 161.0000 161.0000 [19] 161.0000 161.0000 [22] 161.0000 161.0000 [25] 134.0000 856.7264 [28] 856.7264 134.0000 [31] 134.0000 134.0000 [34] 856.7264 856.7264 [37] 856.7264 856.7264</pre>
--	---

Hình 5.5: 1 phần dữ liệu trong Desktop và Mobile

Bước 3: Xác định phương pháp kiểm định.

Đầu tiên, ta cần xác định giá bán lẻ của hai mẫu có tuân theo phân phối chuẩn hay không. Ta sẽ sử dụng kiểm định Shapiro-Wilk tương tự như ở **Bài toán một mẫu**.

```

RGui (64-bit) - [R Console]
File Edit View Misc Packages Windows
Help
> shapiro.test(Desktop)
Shapiro-Wilk normality test

data: Desktop
W = 0.77357, p-value < 2.2e-16

> shapiro.test(Mobile)
Shapiro-Wilk normality test

data: Mobile
W = 0.74642, p-value < 2.2e-16

```

Hình 5.6: Kết quả kiểm định Shapiro-Wilk

Ta nhận thấy giá trị $p\text{-value}$ của giá bán lẻ ở cả hai mẫu đều nhỏ hơn rất nhiều so với 0.05, nên ta có thể kết luận rằng hai tổng thể có phân phối tuỳ ý.

Tiếp theo, ta xác định kích thước của hai mẫu.

```

1 length/Desktop) # Result: 596
2 length/Mobile) # Result: 720

```

Do đó, ta kết luận hướng giải bài toán thuộc dạng **hai mẫu độc lập, tổng thể có phân phối tùy ý và hai mẫu có kích thước lớn**. Ta sử dụng phương pháp kiểm định Z_{test} .

Bước 4: Tính giá trị kiểm định và xác định miền bắc bỏ.

- *Cách 1: Tính bằng công thức và tra bảng.* Thực hiện các lệnh để tìm giá trị trung bình mẫu \bar{x} và phương sai mẫu s^2 của hai mẫu Desktop và Mobile.

```

1 mean/Desktop) # Result: 507.167
2 mean/Mobile) # Result: 610.5114
3 var/Desktop) # Result: 131372.8
4 var/Mobile) # Result: 94885.97

```

Với các kết quả trên, ta có thể tính giá trị kiểm định như sau:

$$Z_{qs} = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}} = \frac{507.167 - 610.5114}{\sqrt{\frac{131372.8}{596} + \frac{94885.97}{720}}} \approx -5.5066$$

Tra bảng với $\alpha = 0.05$ ta tìm được miền bắc bỏ $RR = (Z_\alpha, +\infty) = (1.64; +\infty)$.



- *Cách 2: Sử dụng lệnh `t.test` trong RStudio.* Ta tiến hành kiểm định Z_{test} trên phần mềm RStudio và thu được kết quả như hình sau.

```

> # Tính giá trị z cho kiểm định hai mẫu
> z <- qnorm(1-0.05)
> cat("Miền bác bỏ: RR = (",z,"; +∞) \n")
Miền bác bỏ: RR = ( 1.644854 ; +∞)
>
> # Thực hiện Welch's t-test với giả thuyết đối một phía ( $\mu_1 > \mu_2$ )
> t.test/Desktop, Mobile, alternative = "greater")

    Welch Two Sample t-test

data: Desktop and Mobile
t = -5.5066, df = 1172.4, p-value = 1
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
-134.2382      Inf
sample estimates:
mean of x mean of y
507.1670   610.5114

```

Hình 5.7: Kết quả kiểm định trung bình hai mẫu bằng Z_{test}

* **Chú ý:** t_{test} được dùng trên RStudio đã được viết lại dưới dạng Z_{test} , giá trị của t chính là Z_{qs} .

Từ kết quả chạy trên RStudio, ta có miền bác bỏ $RR = (1.644854; +\infty)$ và $Z_{qs} = -5.5066$, hoàn toàn trùng khớp với kết quả ở *Cách 1*.

Bước 5: Kết luận.

Ta thấy $Z_{qs} \notin RR$, dẫn đến chưa thể bác bỏ giả thuyết H_0 . Tức là với mức ý nghĩa 5% thì chưa thể kết luận được giả bán lẻ đề xuất của CPU cho loại thiết bị Desktop sẽ đắt hơn các CPU cho loại thiết bị Mobile. Phát biểu của bạn Minh Thái là sai.

5.3 Bài toán phân tích phương sai

Đề: Có ý kiến cho rằng giá thành của một CPU tức `Recommended_Customer_Price` không phụ thuộc vào loại thị trường được nhắm đến nên có thể cho rằng giá bán trung bình của các loại thiết bị Mobile, Desktop, Embedded và Server là bằng nhau.

Với mức ý nghĩa 5% và dữ liệu có trong file `Intel CPUs.csv` hãy thực hiện kiểm định xem có thể cho rằng ý kiến trên đúng hay không?



- Giả thiết 2: Các tổng thể có phân phối chuẩn.

Dể kiểm tra giả thiết này, ta sẽ thực hiện kiểm tra Shapiro-Wilk bằng lệnh sau:

```
1 by(anova_Factors$Recommended_Customer_Price, anova_Factors$  
    Vertical_Segment, shapiro.test)
```

<pre>anova_Factors\$Vertical_Segment: Desktop Shapiro-Wilk normality test data: dd[,] W = 0.77357, p-value < 2.2e-16</pre> <hr/> <pre>anova_Factors\$Vertical_Segment: Embedded Shapiro-Wilk normality test data: dd[,] W = 0.68042, p-value < 2.2e-16</pre>	<pre>anova_Factors\$Vertical_Segment: Mobile Shapiro-Wilk normality test data: dd[,] W = 0.74642, p-value < 2.2e-16</pre> <hr/> <pre>anova_Factors\$Vertical_Segment: Server Shapiro-Wilk normality test data: dd[,] W = 0.6858, p-value < 2.2e-16</pre>
--	--

Hình 5.9: Kết quả kiểm tra Shapiro-Wilk của 4 mẫu

Ta có thể thấy rằng giá trị p-value của cả 4 mẫu đều nhỏ hơn rất nhiều so với 0.05. Do đó, ta kết luận rằng cả 4 mẫu đều không tuân theo phân phối chuẩn.

- Giả thiết 3: Phương sai các tổng thể bằng nhau.

Dể kiểm tra giả thiết này, ta sẽ thực hiện chia tỷ số phương sai cho từng cặp mẫu, với 4 mẫu tương ứng với $C_4^2 = 6$ cặp khác nhau. Tỷ lệ phương sai mẫu (ratio) của 2 cặp bất kì nếu nằm trong khoảng $[\frac{1}{2}, 2]$ thì xem như phương sai tổng thể tương đương bằng nhau. Nếu gặp bất kì cặp nào không thỏa mãn điều kiện trên thì ta có thể kết luận dữ liệu của 4 mẫu không thỏa mãn giả thiết 3.

Ta xét cặp Embedded và Server. $\frac{s_3^2}{s_4^2} = \frac{352003.44}{2929165.50} \approx 0.0102 \notin [\frac{1}{2}, 2]$

Từ đây ta có thể kết luận rằng phương sai của các tổng thể không bằng nhau.

Vì không thỏa mãn 2/3 giả thiết của ANOVA nên kết quả phân tích phương sai ANOVA với bộ dữ liệu hiện tại là **không đáng tin cậy**.



Bước 4: Kiểm định ANOVA và xác định miền bắc bỏ.

Ta có thể thực hiện tra bảng Fisher xác định miền bắc bỏ RR với $\alpha = 0.05$, bậc tử là 3 (số nhóm mẫu - 1) và bậc mẫu là ∞ (vì tổng số quan sát là rất lớn và chưa thể xác định). Trong RStudio, ta sử dụng lệnh sau:

```
1 print(qf(1 - 0.05, 4 - 1, length(Mobile) + length/Desktop) +
      length(Embedded) + length(Server) - 4))
```

Kết quả thu được là $RR = (2.608966; +\infty)$.

Tiếp theo, ta tiến hành kiểm định ANOVA với lệnh sau:

```
1 resultANOVA <- aov(Recommended_Customer_Price ~ Vertical_Segment, data = anova_Factors)
2 print(resultANOVA)
3 print(summary(resultANOVA))
```

```
Call:
aov(formula = Recommended_Customer_Price ~ Vertical_Segment,
     data = anova_Factors)

Terms:
          Vertical_Segment Residuals
Sum of Squares       428082489 2281952713
Deg. of Freedom           3            2190

Residual standard error: 1020.778
Estimated effects may be unbalanced
> print(summary(resultANOVA))
          Df   Sum Sq  Mean Sq F value Pr(>F)
Vertical_Segment     3 4.281e+08 142694163    136.9 <2e-16 ***
Residuals           2190 2.282e+09   1041988
---
Signif. codes:  0 '****' 0.001 '***' 0.01 '**' 0.05 '*' 0.1 '.' 1
```

Hình 5.10: Kết quả kiểm định ANOVA



Để dễ nhìn hơn, ta vẽ ra được bảng sau:

Nguồn biến thiên	Tổng bình phương chênh lệch (SS)	Bậc tự do (Df)	Phương sai (Trung bình BPCL)	Tiêu chuẩn kiểm định F	P-value
Giữa các nhóm (Between)	428,082,489	3	142,694,163	136.9	<2e-16
Trong nội bộ nhóm (Within)	2,281,952,713	2,190	1,041,988		

Giải thích kết quả thu được:

$$SSB = n_1 (\bar{X}_1 - \bar{X})^2 + n_2 (\bar{X}_2 - \bar{X})^2 + n_3 (\bar{X}_3 - \bar{X})^2 + n_4 (\bar{X}_4 - \bar{X})^2 = 428,082,489$$

$$SSW = (n_1 - 1)s_1^2 + (n_2 - 1)s_2^2 + (n_3 - 1)s_3^2 + (n_4 - 1)s_4^2 = 2,281,952,713$$

$$MSB = \frac{SSB}{k - 1} = \frac{428,082,489}{3} = 142,694,163$$

$$MSW = \frac{SSW}{N - k} = \frac{2,281,952,713}{2,190} \approx 1,041,988$$

$$F = \frac{MSB}{MSW} \approx \frac{142,694,163}{1,041,988} \approx 136.9$$

Bước 5: Kết luận.

Do $F \in RR$ và $\alpha = 5\% > P\text{-value}$ nên ta bác bỏ H_0 và chấp nhận H_1 . Tức là không thể cho rằng Recommended_Customer_Price của Mobile, Desktop, Embedded và Server bằng nhau. Ý kiến ở đề bài là sai.

* MỞ RỘNG: Ta có thể thực hiện so sánh bội qua các khoảng tin cậy 95% LSD.

```

Tukey multiple comparisons of means
95% family-wise confidence level

Fit: aov(formula = Recommended_Customer_Price ~ Vertical_Segment, data = anova_Factors)

$Vertical_Segment
      diff      lwr      upr   p adj
Embedded-Desktop -38.71882 -267.96552 190.5279 0.9725777
Mobile-Desktop   103.34436 -41.99149 248.6802 0.2602822
Server-Desktop    984.54946  638.75082 1130.3481 0.0000000
Mobile-Embedded   142.06318 -82.80075 366.9271 0.3650554
Server-Embedded   1023.26828  798.10496 1248.4316 0.0000000
Server-Mobile     881.20510  742.39956 1020.0106 0.0000000

```

Hình 5.11: Kết quả so sánh bội ANOVA với $\alpha = 0.05$



diff là chênh lệch giữa hai trung bình mẫu ($\bar{x}_i - \bar{x}_j$). Khoảng tin cậy cho độ chênh lệch μ_i và μ_j là ($\text{lwr}; \text{upr}$) = $(\bar{x}_i - \bar{x}_j) \pm LSD$, trong đó $LSD = t_{\alpha/2; N-k} \sqrt{MSW(\frac{1}{n_i} + \frac{1}{n_j})}$.

p_{adj} là một giá trị p-value đã được điều chỉnh để giảm thiểu rủi ro sai sót loại I khi thực hiện nhiều phép kiểm định đồng thời. Với mức ý nghĩa 5%, khi p_{adj} nhỏ hơn 0.05 thì cặp đó có sự khác biệt.

Nếu khoảng tin cậy ($\text{lwr}; \text{upr}$) không chứa giá trị 0 thì ta nói có sự khác biệt giữa μ_i và μ_j . Cụ thể hơn, nếu khoảng tin cậy chỉ gồm các số dương, xem như $\mu_i > \mu_j$ và ngược lại.

Ta chưa thể kết luận được gì đối với các cặp Embedded-Desktop, Mobile-Desktop và Mobile-Embedded vì cả 3 khoảng tin cậy của 3 cặp này đều chứa 0 và có $p_{\text{adj}} > 0.05$. Tuy nhiên, ta có thể kết luận rằng có sự khác biệt giữa hai giá bán trung bình của 3 cặp Server-Desktop, Server-Embedded và Server-Mobile. Cụ thể:

- Server-Desktop → **Loại thiết bị Server có giá đắt hơn Desktop.**
- Server-Embedded → **Loại thiết bị Server có giá đắt hơn Embedded.**
- Server-Mobile → **Loại thiết bị Server có giá đắt hơn Mobile.**

Có thể kết luận loại thiết bị Server có giá bán trung bình cao nhất trong cả 4 loại.

5.4 Bài toán hồi quy tuyến tính

Đề: Mỗi chiếc CPU là sự kết hợp của nhiều yếu tố khác nhau nên giá cả của mỗi chiếc CPU cũng sẽ khác nhau. Vậy thì các thông số của CPU ảnh hưởng tới giá cả của nó như thế nào?

Giải:

Ta sẽ xây dựng một mô hình cho các yếu tố ảnh hưởng đến biến **Recommended_Customer_Price**. Với các biến khác ta xem như là biến độc lập, biến giá cả là biến phụ thuộc.

Bước 1: Tính hệ số tương quan giữa biến Recommended_Customer_Price và 8 biến còn lại.

Hệ số tương quan (r) là một chỉ số thống kê đo lường mối liên hệ tương quan giữa hai biến số, có giá trị từ -1 đến 1. Hệ số tương quan bằng 0 (hay gần 0) có nghĩa là hai biến số không có liên hệ gì với nhau; ngược lại nếu hệ số bằng -1 hay 1 có nghĩa là hai biến số có một mối liên hệ tuyệt đối.

Hệ số tương quan âm $r < 0$ có nghĩa là khi x tăng cao thì y giảm và ngược lại; hệ số tương quan dương $r > 0$ có nghĩa là khi x tăng cao thì y tăng và ngược lại.



Mục đích của việc tính hệ số tương quan trước khi đi vào xây dựng mô hình là để xem liệu có biến nào không có mối liên hệ gì với biến giá cả hay không. Nếu có, ta tiến hành loại bỏ biến đó khỏi mô hình ngay từ đầu. Vì hầu hết các biến sử dụng trong BTL này có đồ thị không tuân theo phân phối chuẩn, nên ta sẽ sử dụng hệ số tương quan Spearman.

Sau khi chạy phần code dưới đây, máy sẽ in ra kết quả là 1 ma trận 9x9. Ta chỉ cần quan tâm đến cột đầu của ma trận.

```

1 selected_data <- main_Factors[, c("Recommended_Customer_Price"
2   , "Lithography", "nb_of_Cores", "nb_of_Threads", "
3   Processor_Base_Frequency", "Cache", "Max_Memory_Size", "
4   Max_nb_of_Memory_Channels", "Max_Memory_Bandwidth")]
5
6 # Result:
7
8 Recommended_Customer_Price      Recommended_Customer_Price
9 Lithography                      0.28724393
10 nb_of_Cores                     0.11819063
11 nb_of_Threads                   0.42896380
12 Processor_Base_Frequency       -0.09314525
13 Cache                           0.43652066
14 Max_Memory_Size                0.33518095
15 Max_nb_of_Memory_Channels     0.46586224
16 Max_Memory_Bandwidth          0.51146860

```

Ta thấy tất cả các hệ số tương quan đều khác 0, chứng tỏ 8 biến này đều có mối liên hệ với biến Recommended_Customer_Price. Ta sẽ giữ lại cả 8 biến và bắt đầu xây dựng mô hình.

Bước 2: Xây dựng mô hình. Chú ý chia kết quả ra làm 3 phần:

- Phần 1 mô tả phần dư (Residuals) của mô hình hồi quy, bao gồm các chỉ số (Min, 1Q, Median, 3Q, Max) thể hiện phân bố của phần dư.

- Phần 2 trình bày hệ số hồi quy ước tính Estimate, sai số chuẩn Std. Error kèm theo giá trị của kiểm định t, bao gồm t value (= Estimate/Std. Error) và Pr(>|t|). Nếu Pr(>|t|) có giá trị nhỏ hơn 0.05, chúng ta kết luận mối liên hệ giữa 2 biến có ý nghĩa thống kê. Ngoài ra ta còn có dòng Signif. codes để quy ước các kí hiệu kế bên cạnh cột Pr(>|t|). Các biến càng có nhiều '*' càng có ý nghĩa thống kê mạnh. Các biến có dấu '.' có ý nghĩa biến. Các biến không có gì chứng tỏ là không có ý nghĩa thống kê.



– Phần 3 cho ta thông tin về độ lệch chuẩn của phần dư Residual standard error, chỉ số này cho ta biết độ lệch trung bình của mỗi giá cả thực tế khỏi đường hồi quy của dữ liệu mà ta đang xét là bao nhiêu. Giá trị này càng nhỏ thì mô hình càng tốt. Trong kết quả này còn có kiểm định F F-statistic có ý nghĩa tương tự như kiểm định t, giá trị F càng cao thì mô hình càng tốt. Ngoài ra, ta cũng cần chú ý độ lớn của 2 hệ số Multiple R-squared và Adjusted R-squared.

- **Xây dựng mô hình ban đầu:** đầy đủ cả 8 biến.

```

Call:
lm(formula = Recommended_Customer_Price ~ Lithography + nb_of_Cores +
    nb_of_Threads + Processor_Base_Frequency + Cache + Max_Memory_Size +
    Max_nb_of_Memory_Channels + Max_Memory_Bandwidth, data = main_Factors)

Residuals:
    Min      1Q  Median      3Q     Max 
-4415.4 -270.7   -44.8   196.6  7252.4 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -184.3086   67.6815 -2.723 0.006517 ** 
Lithography    5.5076   0.3876 14.210 < 2e-16 ***  
nb_of_Cores    18.7453   4.8598  3.857 0.000118 ***  
nb_of_Threads   81.6126   3.1704 25.742 < 2e-16 ***  
Processor_Base_Frequency 43.1953   20.7873  2.078 0.037829 *  
Cache          3.2761   0.4273  7.668 2.62e-14 ***  
Max_Memory_Size       0.6110   0.0454 13.458 < 2e-16 ***  
Max_nb_of_Memory_Channels -65.8307   25.1809 -2.614 0.009002 ** 
Max_Memory_Bandwidth   -0.1981   1.2322 -0.161 0.872319 
---
Signif. codes:  0 '****' 0.001 '***' 0.01 '**' 0.05 '*' 0.1 '.' 1 

Residual standard error: 686.3 on 2185 degrees of freedom
Multiple R-squared:  0.6202,    Adjusted R-squared:  0.6188 
F-statistic:  446 on 8 and 2185 DF,  p-value: < 2.2e-16

```

Hình 5.12: Mô hình hồi quy tuyến tính ban đầu

→ **Nhận xét:** 8 biến được xét phụ thuộc này giải thích được 62.02% ý nghĩa của biến giá cả.

Kết quả phân tích ở phần 2 cho thấy biến Max_Memory_Bandwidth có giá trị $Pr(>|t|)$ lớn hơn 0.05 rất nhiều, nên ta có thể kết luận biến Max_Memory_Bandwidth không có ý nghĩa trong mô hình hồi quy bội này.

Ta sẽ bắt đầu cải thiện mô hình bằng phương pháp loại bỏ từng biến có ảnh hưởng không đáng kể đối với mô hình ($Pr(>|t|) > 0.05$) hoặc ảnh hưởng không mạnh so với các biến khác trong mô hình ($Pr(>|t|)$ lớn nhất).



Mô hình cải thiện lần 1: gồm 7 biến, đã loại bỏ biến Max_Memory_Bandwidth.

```

Call:
lm(formula = Recommended_Customer_Price ~ Lithography + nb_of_Cores +
   nb_of_Threads + Processor_Base_Frequency + Cache + Max_Memory_Size +
   Max_nb_of_Memory_Channels, data = main_Factors)

Residuals:
    Min      1Q  Median      3Q     Max 
-4409.5 -268.9  -44.4  195.9  7259.6 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) -184.77651   67.60382 -2.733  0.00632 ** 
Lithography      5.49015   0.37193 14.761 < 2e-16 *** 
nb_of_Cores      18.42788   4.43948  4.151 3.44e-05 *** 
nb_of_Threads     81.80622   2.93198 27.901 < 2e-16 *** 
Processor_Base_Frequency 42.97323   20.73669  2.072  0.03835 *  
Cache            3.28373   0.42451  7.735 1.56e-14 *** 
Max_Memory_Size      0.60944   0.04437 13.736 < 2e-16 *** 
Max_nb_of_Memory_Channels -67.97440   21.35420 -3.183  0.00148 ** 
---
Signif. codes:  0 '****' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 

Residual standard error: 686.2 on 2186 degrees of freedom
Multiple R-squared:  0.6202,    Adjusted R-squared:  0.619 
F-statistic:  510 on 7 and 2186 DF,  p-value: < 2.2e-16

```

Hình 5.13: Mô hình hồi quy tuyến tính cải thiện lần 1

→ **Nhận xét:** 7 biến được xét phụ thuộc này giải thích được 62.02% ý nghĩa của biến giá cả.

Hệ số xác định hiệu chỉnh đã được tăng lên chút ít, cho thấy mức độ cải thiện của phương sai phần dư. Thật vậy, ta để ý giá trị **Median** của phần dư ở phần 1 đã tăng nhẹ, tiến gần hơn tới 0. Ngoài ra, ở phần 3, độ lệch chuẩn của phần dư cũng giảm rất nhẹ và giá trị kiểm định F tăng khá cao.

Ta cũng thấy kết quả phân tích ở phần 2 không có biến nào có giá trị $Pr(>|t|)$ lớn hơn 0.05, chứng tỏ tất cả các biến đều có ý nghĩa thống kê trong mô hình.

Tức là mô hình cải thiện lần 1 này đã tốt hơn so với mô hình ban đầu. Tuy nhiên, ta vẫn sẽ tiếp tục phương pháp loại bỏ từng biến đã nêu trên, để xem mô hình còn có thể cải thiện thêm hay không.



Mô hình cải thiện lần 2: gồm 6 biến, đã loại bỏ biến Processor_Base_Frequency so với mô hình lần 1.

```

Call:
lm(formula = Recommended_Customer_Price ~ Lithography + nb_of_Cores +
    nb_of_Threads + Cache + Max_Memory_Size + Max_nb_of_Memory_Channels,
    data = main_Factors)

Residuals:
    Min      1Q  Median      3Q     Max 
-4418.4 -271.2   -45.1   203.7  7270.6 

Coefficients:
                                         Estimate Std. Error t value Pr(>|t|)    
(Intercept)                   -80.76112   45.32192  -1.782 0.074897 .  
Lithography                      5.18793   0.34241  15.151 < 2e-16 *** 
nb_of_Cores                      16.57201   4.35149   3.808 0.000144 *** 
nb_of_Threads                     82.13704   2.92984  28.035 < 2e-16 *** 
Cache                            3.34042   0.42395   7.879 5.15e-15 *** 
Max_Memory_Size                  0.60587   0.04437  13.656 < 2e-16 *** 
Max_nb_of_Memory_Channels       -62.40718   21.20049  -2.944 0.003278 ** 
---
Signif. codes:  0 '****' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 

Residual standard error: 686.7 on 2187 degrees of freedom
Multiple R-squared:  0.6195,    Adjusted R-squared:  0.6184 
F-statistic: 593.4 on 6 and 2187 DF,  p-value: < 2.2e-16

```

Hình 5.14: Mô hình hồi quy tuyến tính cải thiện lần 2

→ **Nhận xét:** 6 biến được xét phụ thuộc này giải thích được 61.95% ý nghĩa của biến giá cả.

Hệ số xác định bội và hệ số xác định hiệu chỉnh đều bị giảm nhẹ. Giá trị Median của phần dư giảm, tiến xa khỏi 0. Độ lệch chuẩn của phần dư cũng đã tăng lên một chút.

Dù vậy, mô hình này cũng có những điểm tích cực. Kết quả phân tích không có biến nào giá trị $Pr(>|t|)$ lớn hơn 0.05, chứng tỏ tất cả các biến đều có ý nghĩa thống kê trong mô hình. Giá trị kiểm định F tăng cao.

Tức là mô hình cải thiện lần 2 này có 1 vài điểm tốt hơn nhưng cũng có 1 vài điểm chưa bằng so với mô hình cải thiện lần 1. Ta tiếp tục phương pháp loại bỏ từng biến đã nêu trên 1 lần nữa, để xem mô hình còn có thể cải thiện thêm hay không.



Mô hình cải thiện lần 3: gồm 5 biến, đã loại bỏ biến Max_nb_of_Memory_Channels so với mô hình lần 2.

```

Call:
lm(formula = Recommended_Customer_Price ~ Lithography + nb_of_Cores
    nb_of_Threads + Cache + Max_Memory_Size, data = main_Factors)

Residuals:
    Min      1Q  Median      3Q     Max 
-4381.3 -254.1   -44.4   200.8  7326.5 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -182.84728  29.22945 -6.256 4.75e-10 ***
Lithography    5.10489   0.34184  14.934 < 2e-16 ***
nb_of_Cores    7.84041   3.18942   2.458   0.014 *  
nb_of_Threads  81.21164   2.91802  27.831 < 2e-16 ***
Cache          3.51986   0.42028   8.375 < 2e-16 ***
Max_Memory_Size  0.58128   0.04365  13.317 < 2e-16 ***
---
Signif. codes:  0 '****' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 687.9 on 2188 degrees of freedom
Multiple R-squared:  0.618,    Adjusted R-squared:  0.6171 
F-statistic: 707.8 on 5 and 2188 DF,  p-value: < 2.2e-16

```

Hình 5.15: Mô hình hồi quy tuyến tính cải thiện lần 3

→ **Nhận xét:** 5 biến được xét phụ thuộc này giải thích được 61.8% ý nghĩa của biến giá cả.

Nhận xét khá giống với mô hình lần 2, thêm một điểm tích cực là giá trị Median của phần dư ở phần 1 đã tăng nhẹ, tiến gần hơn tới 0.

Tức là mô hình cải thiện lần 3 này có vẻ nhỉnh hơn so với mô hình cải thiện lần 2, nhưng vẫn chưa bằng mô hình cải thiện lần 1.



Thông qua việc so sánh giữa các mô hình cải thiện với nhau, nhóm chọn được mô hình cải thiện lần 1 là mô hình đạt hiệu quả cao nhất. Tức là sự biến đổi của biến Recommended_Customer_Price phụ thuộc rất nhiều vào các biến độc lập, ngoại trừ biến Max_Memory_Bandwidth. Mô hình hồi quy tuyến tính nhóm lựa chọn sẽ theo phương trình sau:

$$\begin{aligned} \text{Recommended_Customer_Price} \\ = -184.7765 + 5.4902 \times \text{Lithography} + 18.4279 \times \text{nb_of_Cores} \\ + 81.8062 \times \text{nb_of_Threads} + 42.9732 \times \text{Processor_Base_Frequency} + 3.2837 \times \text{Cache} \\ + 0.6094 \times \text{Max_Memory_Size} - 67.9744 \times \text{Max_nb_of_Memory_Channels} \end{aligned}$$

Bước 3: Kiểm định lại mô hình.

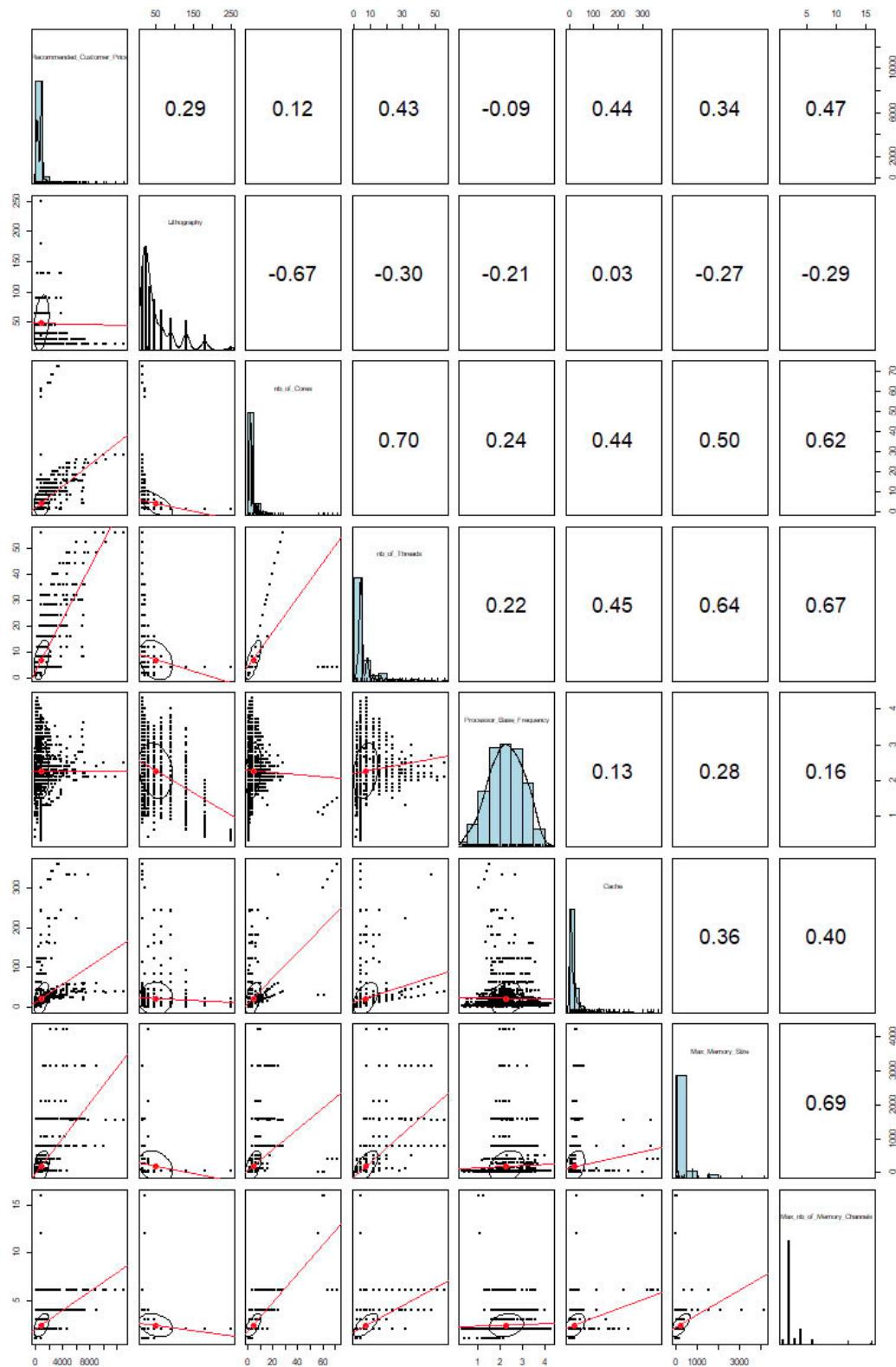
Việc kiểm định lại mô hình hồi quy giúp đảm bảo tính chính xác và độ tin cậy của các dự đoán. Ta cần kiểm tra xem nó có đáp ứng đầy đủ các giả định hay không, từ đó tìm ra được những ưu điểm và hạn chế riêng của mô hình ta đã chọn. Mô hình hồi quy tuyến tính có 2 giả định, bao gồm:

- Giả định 1: Các sai số ngẫu nhiên (phản dư) ε_i độc lập. Khi kiểm tra, điều kiện này sẽ được thay thế bằng việc kiểm tra giả thiết các X_i không xảy ra quan hệ đa cộng tuyến.

Đa cộng tuyến là hiện tượng khi hai hoặc nhiều biến độc lập trong mô hình hồi quy có mối quan hệ tuyến tính mạnh với nhau. Điều này có thể gây ra một số vấn đề nghiêm trọng trong việc ước lượng các tham số của mô hình. Mô hình có dấu hiệu của hiện tượng đa cộng tuyến khi hệ số tương quan r của bất kỳ cặp biến nào trong mô hình lớn hơn 0.8. Ta thực hiện vẽ đồ thị hệ số tương quan bằng đoạn code dưới đây:

```

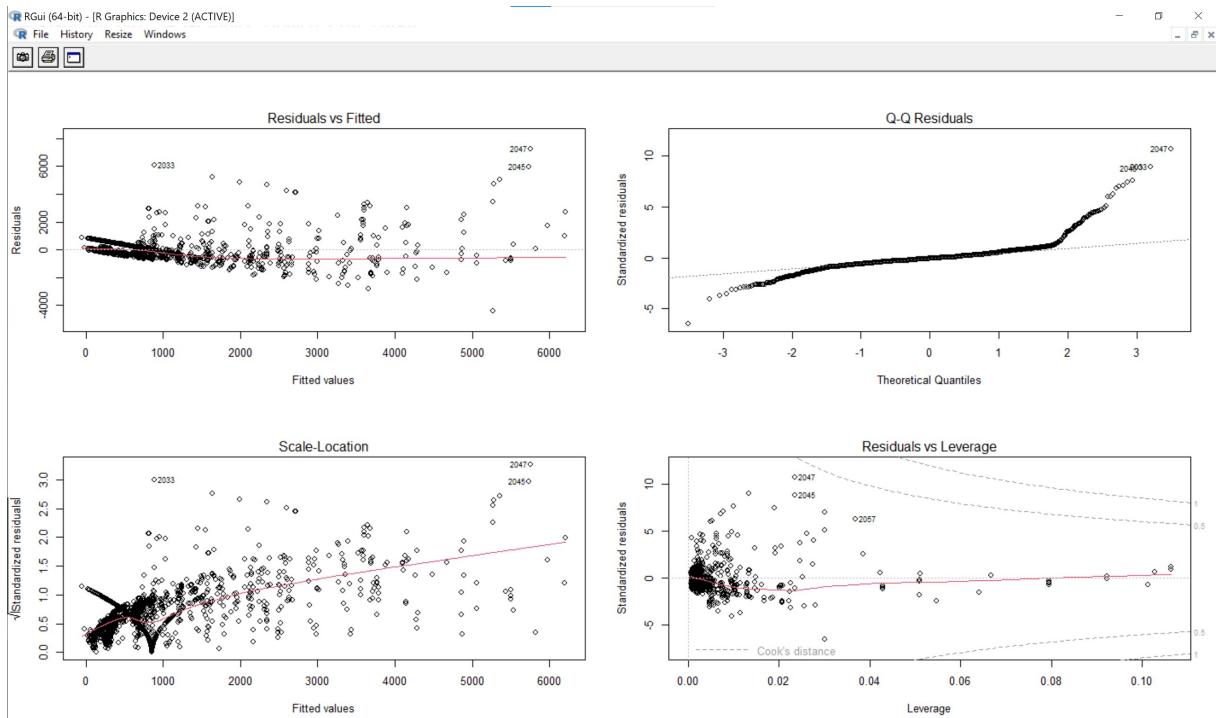
1 library(car)
2
3 pairs.panels(
4   selected_data,
5   method = "spearman",
6   hist.col = "lightblue",
7   density = TRUE,
8   ellipses = TRUE,
9   lm = TRUE)
```



Hình 5.16: Đồ thị hệ số tương quan trong mô hình hồi quy đã chọn

Ta thấy không có cặp biến nào trong mô hình có r lớn hơn 0.8. Vậy mô hình hồi quy ta đã chọn thỏa mãn hoàn toàn giả định này.

- Giả định 2: Các sai số ngẫu nhiên (phần dư) ε_i có cùng phân phối chuẩn $N(0, \sigma^2)$ với σ không đổi.



Hình 5.17: Phân tích phần dư để kiểm tra các giả định

– Đồ thị **Residuals vs Fitted** vẽ ε_i và giá trị tiên đoán cho biến giá cả \hat{y}_i . Đồ thị này được dùng để kiểm tra giả thiết tuyến tính của dữ liệu và giả thiết phần dư có trung bình bằng 0. Nếu đường màu đỏ trên đồ thị phân tán là đường thẳng nằm ngang, thì giả thiết tính tuyến tính của dữ liệu được thỏa mãn. Giả thiết phần dư có trung bình bằng 0 thỏa mãn nếu đường màu đỏ gần với đường nằm ngang (ứng với phần dư = 0).

Nhìn vào đồ thị cho ta thấy giả thiết về tính tuyến tính của dữ liệu hơi bị vi phạm. Tuy nhiên giả thiết trung bình của phần dư có thể coi là thỏa mãn.

– Đồ thị **Q-Q Residuals** vẽ giá trị phần dư ε_i và giá trị kì vọng dựa vào phân phối chuẩn. Ta thấy đa phần các số phần dư tập trung rất gần các giá trị trên đường chuẩn, cho nên ta tạm chấp nhận ε_i tuân theo phân phối chuẩn.

– Đồ thị **Scale-Location** vẽ căn số phần dư chuẩn và giá trị tiên đoán cho biến giá cả **Recommended_Customer_Price** \hat{y}_i . Nếu đường màu đỏ trên đồ thị là đường thẳng nằm ngang và các điểm thặng dư phân tán đều xung quanh đường thẳng này thì giả thiết σ



không đổi được thỏa mãn. Nếu như đường màu đỏ có độ dốc (hoặc cong), hoặc các điểm thăng dư phân tán không đều xung quanh đường thẳng này thì giả thiết này bị vi phạm.

Ta thấy đường màu đỏ của đồ thị có hình dáng đường thẳng hướng thẳng lên trên chứ không phải nằm ngang, kèm theo đó các giá trị phần dư cũng không phân tán đều trên đường thẳng này. Vì vậy, giả thiết σ không đổi bị vi phạm. Đây chính là điểm yếu của mô hình hồi quy nhóm đã chọn, khiến phương trình mà ta ước tính có vấn đề hợp lý.

Đồ thị Residuals vs Leverage cần chú ý vị trí của các điểm. Điểm nào nằm ngoài các đường nét gọi là điểm ảnh hưởng. Ý nghĩa khi có điểm này nhầm báo hiệu rằng mô hình hồi quy ta đang xét vẫn chưa đủ tốt. Ta thấy rằng không có bất kỳ điểm ảnh hưởng nào trong mô hình hồi quy, chứng tỏ đây là 1 mô hình đủ tốt.

Vì không thỏa mãn đầy đủ giả thiết của hồi quy tuyến tính nên kết quả bài toán với bộ dữ liệu hiện tại là **chưa đáng tin cậy**. Tuy nhiên, dựa vào những nhận xét về đặc điểm của mô hình trước đây, nhóm tin đây là mô hình hồi quy tốt nhất mà nhóm có thể tìm ra.

6 Thảo luận và mở rộng

6.1 Kiểm định trung bình một mẫu

* **Ưu điểm:** Đây là phương pháp thống kê đơn giản và dễ thực hiện, thích hợp cho các nghiên cứu nhỏ với dữ liệu có sẵn. Có thể áp dụng cho mọi loại dữ liệu, từ liên tục đến rời rạc.

* **Nhược điểm:** Phương pháp này có hạn chế về mặt so sánh, chỉ cho phép so sánh một nhóm mẫu với một giá trị tham chiếu.

6.2 Kiểm định trung bình hai mẫu

* **Ưu điểm:** Đây là một trong những phương pháp thống kê cơ bản và phổ biến nhất, cho phép so sánh sự khác biệt giữa hai nhóm mẫu. Phù hợp với cả mẫu lớn và mẫu nhỏ.

* **Nhược điểm:** Phương pháp này có thể gây ra hạn chế trong việc điều chỉnh cho các biến có ảnh hưởng đến kết quả nghiên cứu, dẫn đến việc các kết quả có thể bị méo mó vì ảnh hưởng của các biến ngoại lai.



6.3 Phân tích phương sai một yếu tố

* **Ưu điểm:** Phương pháp này dễ hiểu và trực quan, có thể kiểm tra sự khác biệt giữa ba hoặc nhiều hơn hai nhóm. Có hiệu quả về mặt thời gian hơn việc kiểm định t-test hoặc z-test riêng lẻ.

* **Nhược điểm:** Phương pháp này yêu cầu các tổng thể có phân phối chuẩn, phương sai các tổng thể bằng nhau, các mẫu quan sát được lấy độc lập; không chỉ ra sự khác biệt cụ thể của các nhóm (chỉ cho biết các nhóm không bằng nhau).

6.4 Hồi quy tuyến tính bội

* **Ưu điểm:** Phương pháp này cho phép phân tích đồng thời tác động của nhiều biến độc lập lên một biến phụ thuộc, nhằm đưa ra dự đoán giá trị của biến phụ thuộc khi biết các biến độc lập.

* **Nhược điểm:** Hồi quy tuyến tính bội cũng khá nhạy cảm với điểm ngoại lai, yêu cầu dữ liệu lớn để đảm bảo độ tin cậy cho mô hình hồi quy.

7 Nguồn dữ liệu và nguồn code

– Đường link nguồn dữ liệu:

<https://www.kaggle.com/datasets/iliassekkaf/computerparts/data>

– Đường link tải R code:

<https://drive.google.com/file/d/1KvusVo8kJ4Dn7sqEWSIQvMHSgEmHyWHD/>

Tài liệu

- [1] Hoàng Trọng & Chu Nguyễn Mộng Ngọc. *Thống kê ứng dụng trong kinh tế - xã hội*. NXB Thống kê, 2008.
- [2] Nguyễn Đình Huy & Đậu Thế Cấp. *Giáo trình Xác suất và Thống kê*. NXB Đại học Quốc gia TP. Hồ Chí Minh, 2021.
- [3] Nguyễn Thanh Nga. *Mô Hình Hồi Quy Bội*. <https://rpubs.com/TKUD/810985>, 2021.
Ngày truy cập cuối cùng: 14/08/2024.



- [4] Nguyễn Văn Tài. *Bài Tập 1 - Câu 9 - Phân tích hồi quy và kiểm tra đa cộng tuyến.* <https://rpubs.com/vantai/bt1c9>, 2020. Ngày truy cập cuối cùng: 14/08/2024.
- [5] Nguyễn Văn Tuấn. *Phân tích số liệu và biểu đồ bằng R.* https://cran.r-project.org/doc/contrib/Intro_to_R_Vietnamese.pdf. Ngày truy cập cuối cùng: 12/08/2024.
- [6] Overleaf. *Learn LaTeX in 30 minutes.* https://www.overleaf.com/learn/latex/Learn_LaTeX_in_30_minutes. Ngày truy cập cuối cùng: 24/06/2024.
- [7] ZACH BOBBITT. *How to Interpret a Scale-Location Plot (With Examples).* <https://www.statology.org/scale-location-plot/>, 2020. Ngày truy cập cuối cùng: 13/08/2024.
- [8] ZACH BOBBITT. *What is a Residuals vs. Leverage Plot? (Definition Example).* <https://www.statology.org/residuals-vs-leverage-plot/>, 2021. Ngày truy cập cuối cùng: 13/08/2024.