Vietnam National University, Ho Chi Minh City
University of Technology
Faculty of Computer Science and Engineering

Xác suất thống kê (MT2013)

Nhóm MT05

Báo cáo bài tập lớn

"Phân tích giá cả CPU dựa trên các thông số kĩ thuật sử dụng các phương pháp thống kê suy diễn"

Giảng viên: TS. Nguyễn Kiều Dung

STT	Họ tên SV	MSSV	Tên lớp	Tên khoa/ngành
1	**Hồ Chí Công***	**2310370**	**L13**	**KH & KT máy tính**
2	Đặng Gia Bảo	2310210	L13	KH & KT máy tính
3	Nguyễn Văn Đức	2310790	L13	KH & KT máy tính
4	Vũ Minh Sang	2312944	L13	KH & KT máy tính
5	Nguyễn Huy Phúc	2312696	L13	KH & KT máy tính
6	Đặng Hải Sơn	2312955	L13	KH & KT máy tính
7	Nguyễn Trần Anh Quân	2312845	L13	KH & KT máy tính
8	Trần Gia Hiển	2311002	L02	KH & KT máy tính

Ho Chi Minh City, April 2025

Danh sách thành viên & Phân công nhiệm vụ

STT	Họ và tên	MSSV	Nhiệm vụ	% Tiến độ
1	Hồ Chí Công	2310370	- Đánh giá tổng quan - Định dạng và viết báo cáo - Tổng kết nội dung - Đặt ra bài toán hồi quy tuyến tính bội	100%
2	Đặng Gia Bảo	2310210	- Phân tích dữ liệu, tiền xử lý số liệu - Lý thuyết thống kê suy diễn - Đặt ra bài toán kiểm định 2 mẫu - Viết báo cáo	100%
3	Nguyễn Văn Đức	2310790	- Phân tích dữ liệu - Lọc dữ liệu ban đầu - Hiện thực bài toán hồi quy tuyến tính bội - Giải thích phần hiện thực	100%
4	Vũ Minh Sang	2312944	- Phân tích dữ liệu đã làm sạch - Hiện thực bài toán kiểm định một mẫu - Giải thích phần hiện thực	100%
5	Nguyễn Huy Phúc	2312696	- Viết báo cáo - Phân tích dữ liệu đã làm sạch - Đặt ra bài toán Anova một chiều	100%
6	Đặng Hải Sơn	2312955	- Phân tích dữ liệu đã làm sạch - Hiện thực bài toán kiểm định hai mẫu - Giải thích phần hiện thực	100%
7	Nguyễn Trần Anh Quân	2312845	- Giới thiệu về dữ liệu - Lý thuyết thống kê mô tả - Đặt ra bài toán kiểm định một mẫu	100%
8	Trần Gia Hiển	2311002	- Phân tích dữ liệu đã làm sạch - Hiện thực bài toán Anova một chiều - Giải thích phần hiện thực	100%

Bảng 1: Bảng phân công nhiệm vụ

Contents

LỜI CẢM ƠN

Chúng em xin được gửi lời cảm ơn chân thành đến cô Nguyễn Kiều Dung - Giảng viên bộ môn Xác Suất và Thống Kê, Trường Đại học Bách khoa - Đại học Quốc gia TP Hồ Chí Minh. Cô đã tận tình hướng dẫn chúng em trong quá trình giải quyết bài toán, giúp chúng em trang bị những kỹ năng cơ bản và kiến thức cần thiết. Qua đó chúng em dễ dàng nắm bắt được nội dung cũng như tìm ra phương pháp để giải quyết vấn đề đặt ra. Tuy nhiên, trong quá trình hiện thực và hoàn thiện đề tài bài tập lớn, chắc chắn vẫn sẽ còn thiếu sót. Rất mong nhận được sự góp ý, đánh giá của cô để giải pháp của chúng em hoàn thiện hơn.

1 Tổng quan dữ liệu

1.1 Giới thiệu chung

Bộ xử lý trung tâm (CPU) là thành phần cốt lõi của hệ thống máy tính hiện đại, đóng vai trò quan trọng trong việc thực hiện các lệnh và thực hiện các phép tính cần thiết cho tất cả các phần mềm tương tác với phần cứng. Là thành phần xử lý chính, CPU chịu trách nhiệm xử lý nhiều loại tác vụ trong môi trường cá nhân và doanh nghiệp. Ngoài vai trò chính trong tính toán tổng quát, CPU còn rất cần thiết trong lĩnh vực như xử lý dữ liệu, quản lý hệ thống và chạy các hệ thống phức tạp.

Mục tiêu của dự án này là phân tích các đặc điểm cụ thể của CPU như được trình bày trong tập dữ liệu được cung cấp. Bảng dưới đây tóm tắt thông tin về tập dữ liệu CPU mà chúng ta đang sử dụng:

Thuộc tính	**Thông tin**
Nguồn dữ liệu	Intel_CPUs.csv
Đối tượng nghiên cứu	CPU
Số lượng quan sát	2283
Số lượng các loại biến	45

1.2 Tổng quan về các loại biến

1. **Product_Collection:** Là tên của các bộ xử lý trung tâm (CPU) khác nhau.
2. **Vertical_Segment:** Là một phân khúc thị trường hoặc ngành cụ thể mà sản phẩm nhắm tới.
3. **Processor Base Frequency (Xung nhịp cơ bản của CPU, đơn vị GHz hoặc MHz):** đề cập đến tốc độ hoạt động tối thiểu được đảm bảo của CPU trong các điều kiện hoạt động tiêu chuẩn. Xung nhịp cơ bản càng cao thì tốc độ xử lý của CPU càng nhanh.

4. **Lithography (Quang khắc):** là quá trình được sử dụng để tạo ra các mẫu tinh vi của bóng bán dẫn và các thành phần khác trên tấm silicon, tạo nên CPU.
5. **Nb_of_cores (số lượng lõi):** là số đơn vị xử lý riêng lẻ bên trong một bộ vi xử lý.
6. **Nb_of_threads (số lượng luồng):** là số lượng các tác vụ mà CPU có thể xử lý đồng thời.
7. **Cache (bộ nhớ đệm):** là một loại bộ nhớ tốc độ cao nằm ngay hoặc gần với CPU và được sử dụng để lưu trữ tạm thời các dữ liệu mà CPU thường xuyên truy cập.
8. **TDP (Thermal Design Power, đơn vị W):** là chỉ số thể hiện lượng nhiệt tối đa mà bộ vi xử lý (CPU) tạo ra dưới tải trong làm việc trung bình mà hệ thống làm mát cần phải tiêu tán.
9. **Bus_Speed:** là tốc độ truyền dữ liệu giữa CPU và các thành phần khác của máy tính, ngoài ra Bus Speed còn cho biết số lượng chu kì mà bus có thể thực hiện mỗi giây.
10. **Max_Memory_Size:** dung lượng RAM tối đa hỗ trợ.
11. **Max_nb_of_Memory_channels:** là số lượng kênh mà CPU có thể hỗ trợ kết nối với RAM.
12. **Max_Memory_Bandwidth:** hay còn gọi là băng thông bộ nhớ tối đa (GB/s) là chỉ số cho biết tốc độ tối đa mà dữ liệu có thể truyền tải giữa CPU và RAM trong một hệ thống máy tính.
13. **Instruction_set:** Là tập hợp các lệnh mà một bộ vi xử lý (CPU) có thể thực hiện. Những lệnh này được sử dụng để thực hiện các phép toán, điều khiển dòng chương trình, và tương tác với phần cứng của máy tính.

14. **Recommend_Customer_Price:** là mức giá gợi ý cho khách hàng dựa trên những tiêu chí về hiệu suất của CPU được nêu ở trên.

2 Kiến thức nền

2.1 Khai phá dữ liệu (thống kê mô tả)

Khai phá dữ liệu (EDA) là một bước quan trọng ban đầu trong các dự án khoa học dữ liệu. Nó bao gồm việc phân tích và trực quan hóa dữ liệu để hiểu các đặc điểm chính, phát hiện các mẫu, và xác định các mối quan hệ giữa các biến – là cơ sở để chọn phương pháp nghiên cứu và khám phá các tập dữ liệu để làm nổi bật các đặc điểm chính, khám phá mẫu, xác định ngoại lệ, và xác định mối quan hệ giữa các biến. EDA thường được thực hiện như một bước sơ bộ trước khi thực hiện các phân tích thống kê chính thức hoặc mô hình hóa.

2.1.1 Biểu đồ tương quan (Correlogram)

Correlogram (còn gọi là biểu đồ tương quan) là một loại biểu đồ thể hiện mối quan hệ giữa từng cặp biến số dạng số trong một tập dữ liệu. Mối quan hệ giữa từng cặp biến được trực quan hóa thông qua biểu đồ phân tán hoặc ký hiệu thể hiện hệ số tương quan r_{xy} (như chấm, bong bóng, số, v.v.).

Correlogram có thể giúp trả lời các câu hỏi sau:

1. Dữ liệu có ngẫu nhiên không?
2. Một quan sát có liên quan đến quan sát liền kề không?
3. Mô hình nào là phù hợp cho chuỗi dữ liệu quan sát?

2.1.2 Các công cụ trực quan hóa thường dùng khác

- **Biểu đồ Q-Q (Quantile-quantile plot)** được dùng để xác định xem một tập dữ liệu có tuân theo một phân phối xác suất cụ thể nào đó hay

không, hoặc hai mẫu dữ liệu có đến từ cùng một tổng thể hay không. Q-Q plot đặc biệt hữu ích để kiểm tra giả định về phân phối chuẩn hoặc một số phân phối đã biết khác.

- **Biểu đồ Histogram** dùng để biểu diễn số lượng các giá trị của một biến nhất định rơi vào từng khoảng cụ thể. Nó giúp ta hình dung được phân bố dữ liệu.
- **Biểu đồ phân tán (Scatter plot)** biểu diễn mối quan hệ giữa một biến độc lập (trên trục x) và một biến phụ thuộc (trên trục y).
- **Biểu đồ hộp (Boxplot)** là biểu đồ mô tả trực quan các đặc điểm quan trọng của một tập dữ liệu như trung tâm, độ phân tán, sự lệch khỏi đối xứng, và các giá trị bất thường hoặc ngoại lệ.

2.2 Thống kê suy diễn

2.2.1 Khái niệm

Thống kê suy diễn (Inferential Statistics) là các phương pháp toán thống kê để đánh giá thông tin mô tả từ dữ liệu mẫu nhằm xác định tính tin cậy của thống kê bằng cách kiểm chứng giả thuyết. Có rất nhiều công cụ để tính toán thống kê suy diễn như: Phân phối Student, phân phối Fisher, phương pháp ANOVA, hồi quy. . . Bài báo cáo này chủ yếu tập trung vào bốn công cụ là kiểm định một mẫu, kiểm định hai mẫu, phân tích phương sai một chiều (ANOVA) và hồi quy tuyến tính bội (multiple Linear Regression).

2.2.2 Kiểm định một mẫu

Kiểm định trung bình 1 mẫu được sử dụng để kiểm tra xem giá trị trung bình của 1 mẫu có khác biệt so với giá trị giả thuyết (giá trị kì vọng) hay không. Cặp giả thuyết của kiểm định này là:

- **Giả thuyết** H_0**:** Trung bình của mẫu bằng với trung bình kì vọng ($\mu = \mu_0$).

- **Giả thuyết đối** H_1**:** Trung bình của mẫu không bằng ($\mu \neq \mu_0$), lớn hơn ($\mu > \mu_0$), hoặc nhỏ hơn giá trị kì vọng ($\mu < \mu_0$).

Trước khi thực hiện kiểm định, ta cần xác định mẫu muốn kiểm định thuộc dạng phân phối gì, đã biết hay chưa biết phương sai (σ^2) của tổng thể nhằm chọn được loại kiểm định phù hợp với mô hình cũng như tính toán đúng được giá trị thống kê. Và đánh giá kết quả dựa vào giá trị thống kê kết hợp với mức ý nghĩa.

2.2.3 Kiểm định hai mẫu

Kiểm định trung bình hai mẫu là phương pháp kiểm định giả thiết thống kê, được sử dụng để so sánh giá trị trung bình giữa hai nhóm độc lập hoặc phụ thuộc. Do đó, cặp giả thuyết của loại kiểm định này cũng được chia thành 2 loại như sau:

Với hai mẫu độc lập:

- **Giả thuyết** H_0**:** Trung bình của 2 tổng thể là bằng nhau, không có sự khác biệt giữa chúng ($\mu_1 = \mu_2$).
- **Giả thuyết đối** H_1**:** Trung bình của 2 tổng thể có sự khác nhau ($\mu_1 \neq \mu_2$) hoặc ($\mu_1 > \mu_2$) hoặc ($\mu_1 < \mu_2$).

Với 2 mẫu phụ thuộc tương ứng theo cặp (với biến $X_D = X_1 - X_2$**):**

- **Giả thuyết** H_0**:** Không có sự khác biệt giữa trung bình của các cặp giá trị tương ứng ($\mu_D = 0$).
- **Giả thuyết đối** H_1**:** Có sự khác biệt giữa trung bình của các cặp giá trị tương ứng ($\mu_D \neq 0$) hoặc ($\mu_D > 0$) hoặc ($\mu_D < 0$).

Tùy vào 2 mẫu là độc lập hay phụ thuộc mà ta sẽ chọn cặp giả thuyết phù hợp, cũng như cần kiểm tra 2 mẫu được thu thập có dạng phân phối

gì, đã biết phương sai hay chưa để áp dụng phương pháp thích hợp. Sau đó tính toán giá trị thống kê và đưa ra kết luận bài toán.

2.2.4 Phân tích phương sai (ANOVA)

Phương pháp phương sai (ANOVA) là công cụ dùng để kiểm định sự khác biệt giữa các trung bình tổng thể. Giả thuyết thống kê của công cụ này là H_0 tương ứng với các trung bình tổng thể bằng nhau.

Đối với mô hình ANOVA một nhân tố:

- Điều kiện của giả định để thực hiện phân tích phương sai một nhân tố là
  - Các quan sát có giá trị độc lập.
  - Có phân phối chuẩn (phân phối Normal).
  - Có phương sai thuần nhất.

Các trung bình tổng thể được kí hiệu: $\mu_1, \mu_2, \ldots, \mu_n$ thì khi đem đi kiểm định ta khi đem đi kiểm định ta sẽ có giả thuyết thống kê tương đương: $H_0 = \mu_1 = \mu_2 = \cdots = \mu_n$. Về mặt ngôn ngữ tự nhiên, nghĩa là những yếu tố đang xét không tác động đến vấn đề đang nghiên cứu. Giả thuyết đối sẽ là H_1: tồn tại ít njaats một cặp không bằng nhau. Về mặt ngôn ngữ tự nhiên, nghĩa là có những yếu tố tác động đến vấn đề đang nghiên cứu. Mục tiêu tính toán của phương pháp này là tỉ số:

$$F_{test} = \frac{MSB}{MSW}$$

Xét n số lượng tổng thể với a là số nhóm tổng thể mà các quan sát chia thành.

	SUM of SQUARE	BẬC TỰ DO	MEAN of SQUARE	KIỂM ĐỊNH
WITHIN	$SSW = \sum_{i=1}^{a}\sum_{j=1}^{n}(x_{ij} - \bar{x}_i)^2$	$N - a$	$\text{MSW} = \frac{\text{SSW}}{\text{N} - \text{a}}$	$F_{\text{test}} = \frac{\text{MSB}}{\text{MSW}}$
BETWEEN	$SSB = \sum_{i=1}^{a}\sum_{j=1}^{n}(\bar{x}_i - \bar{x})^2$	$a - 1$	$\text{MSB} = \frac{\text{SSB}}{\text{a} - 1}$	
TOTAL	$SST = \sum_{i=1}^{a}\sum_{j=1}^{n}(x_{ij} - \bar{x})^2$	$N - 1$		

Bảng 2: Tóm tắt phép tính ANOVA

Định lý: Tổng các chênh lệch bình phương toàn bộ bằng tổng các chênh lệch bình phương nội bộ cộng với tổng các chênh lệch bình phương giữa các nhóm.

$$SST = SSW + SSB$$

Kiểm định giả thuyết:

$$F_{test} = \frac{MSB}{MSW}$$

Nếu giá trị kiểm định lớn nghĩa là có sự khác biệt lớn giữa μ của các nhóm so với sự biến động bên trong mỗi nhóm. Khi giá trị F_{test} vượt qua giá trị nhất định ở mức ý nghĩa nhất định, ta có thể bác bỏ H_0.
Có rất nhiều cách để so sánh với mỗi cách đem đến những ý nghĩa khác nhau như tương phản trực quan, chênh lệch nhỏ nhất, phạm vi bội số mới,... nhưng trong bài báo cáo lần này chỉ nhắc đến phương pháp chênh lệch nhỏ nhất của Fisher (LSD).

$$LSD = t_{(\frac{\alpha}{2};(N-a))} \times \sqrt{2MSW \times \left(\frac{1}{n_i} + \frac{1}{n_j}\right)}$$

- Nếu $|\overline{x_i} - \overline{x_j}| < LSD$ thì $\mu_1 = \mu_2$
- Nếu $|\overline{x_i} - \overline{x_j}| > LSD$ thì $\mu_1 \neq \mu_2$

$\rightarrow$ Ước lượng sự chênh lệch giữa 2 tổng thể:

$$\mu_1 - \mu_2 \in (\overline{x_1} - \overline{x_2}) \pm t_{(\frac{\alpha}{2};(N-a))} \times \sqrt{2MSW \times \left(\frac{1}{n_1} + \frac{1}{n_2}\right)}$$

2.2.5 Hồi quy tuyến tính bội

Hồi quy tuyến tính bội (Multiple linear regression - MLR) là một kĩ thuật thống kê được sử dụng để mô hình hóa mối quan hệ giữa một biến phụ thuộc (mục tiêu) và nhiều biến độc lập (dự báo hoặc đặc trưng) nhằm dự đoán kết quả của một biến phản ứng. Các giả định của MLR được bao gồm:

- Đồng nhất phương sai
- Độc lập của các quan sát (Independence of observations)
- Phân phối chuẩn (Normality)
- Quan hệ tuyến tính (Linearity)
- Không có đa cộng tuyến (No multicollinearity)

Phương trình hồi quy tuyến tính đa biến tổng quát

$$Y = \beta_0 X_1 + \beta_1 X_2 + \cdots + \beta_n X_n + \varepsilon$$

Với các giá trị $i = 1, 2, \ldots, n$ là các giá trị quan sát:

- Y: Biến phụ thuộc.
- $X_1, X_2, \ldots, X_n$: Các biến độc lập.
- β_0: Hệ số tự do (y-intercept) là giá trị của Y khi tất cả các biến độc lập bằng 0.
- β_i: Các hệ số góc tương ứng với biến giải thích X_i.
- ε: sai số ngẫu nhiên của mô hình (residuals).

Một số tham số điển hình trong MLR

1. S_{xx}

$$S_{xx} = \sum_{i=1}^{n} (x_i - \bar{x})^2 \overset{stat}{=} n \times \sigma_x^2 \equiv (n-1) \times s_x^2$$

2. S_{yy}: tương tự S_{xx}

3. S_{xy}

$$S_{xy} = \sum_{i=1}^{n} (x_i - \bar{x})^2 \times (y_i - \bar{y})^2$$

4. Hiệp phương sai (Covariance)

$$Cov(X, Y) = \frac{S_{xy}}{n-1}$$

Nếu:

- $Cov(X, Y) > 0$: Hai tổng thể tương quan cùng chiều.
- $Cov(X, Y) = 0$: Hai tổng thể không liên quan.
- $Cov(X, Y) < 0$: Hai tổng thể tương quan ngược chiều.

5. Hệ số tương quan mẫu r_{xy} là một ước lượng của hệ số tương quan ρ giữa X, Y

$$r = \frac{s_{xy}}{\sqrt{s_{xx}.s_{yy}}}$$

- $| r_{xy} | \leq 0.3$: không có hồi quy tuyến tính hoặc rất yếu.
- $0.3 < | r_{xy} | \leq 0.5$: hồi quy tuyến tính yếu.
- $0.5 < | r_{xy} | \leq 0.8$: hồi quy tuyến tính trung bình.
- $0.8 < | r_{xy} |$: hồi quy tuyến tính mạnh.

6. Hệ số xác định: Để tính được hệ số xác định R^2, ta cần tìm được sai số khác biệt giữa các đường hồi quy mẫu và trung bình của $Y (SSR)$ và được tính bằng công thức

$$SSR = \widehat{\beta_1}^2 . S_{xx} = \sum_{i=1}^{n} (\widehat{y_i} - \overline{y})^2$$

7. Sai số khác biệt giữa các quan sát với giá trị dự đoán trước (SSE). Đây được xem là sai số do yếu tố ngẫu nhiên:

$$SSE = s_{yy} - \widehat{\beta_1} . s_{xy} = \sum_{i=1}^{n} (y_i - \widehat{y_i})^2$$

8. SST (sum of square total)

$$SST = SSR + SSE$$

9. Hệ số xác định R^2 (R squared)

$$R^2 = \frac{SSR \times 100\%}{SST} = 1 - \frac{SSE}{SST}$$

10. Hệ số xác định thích nghi (Adjusted R square) Dùng biến Adjusted R^2 thay cho R^2 vì nó khắc phục được nhược điểm của R^2 là khi có thêm biến mới không liên quan gì đến Y, thì R^2 không thay đổi, trong khi đó Adjusted R^2 sẽ tăng nếu biến mới này phù hợp với mô hình. Ngược lại, nếu Adjusted R^2 giảm thì biến mới thêm vào mô hình là không phù hợp.

$$Adjusted\_R^2 = 1 - \frac{\left(1 - R^2\right) \times (n-1)}{n - m - 1}$$

Với n, m lần lượt là số hàng và số cột (số lượng các biến độc lập) của mô hình.

3 Tiền xử lý dữ liệu

3.1 Đọc dữ liệu

Trước hết, nhóm tác giả sẽ bắt đầu bằng việc thông qua các dữ liệu được cung cấp từ nguồn. Một bản xem trước chứa vài dòng dữ liệu đầu sẽ được hiển thị để ta có thể làm quen với định dạng dữ liệu. Bảng dữ liệu được mô tả trong Hình 12 và được tổng kết qua Hình 2.

Product_Collection	Vertical_Segment	Processor_Number	Status	Launch_Date	Lithography
7th Generation Intel® Core™ i7 Processors	Mobile	i7-7Y75	Launched	Q3'16	14 nm
8th Generation Intel® Core™ i5 Processors	Mobile	i5-8250U	Launched	Q3'17	14 nm
8th Generation Intel® Core™ i7 Processors	Mobile	i7-8550U	Launched	Q3'17	14 nm
Intel® Core™ X-series Processors	Desktop	i7-3820	End of Life	Q1'12	32 nm
7th Generation Intel® Core™ i5 Processors	Mobile	i5-7Y57	Launched	Q1'17	14 nm
Intel® Celeron® Processor 3000 Series	Mobile	3205U	Launched	Q1'15	14 nm
Intel® Celeron® Processor N Series	Mobile	N2805	Launched	Q3'13	22 nm
Intel® Celeron® Processor J Series	Desktop	J1750	Launched	Q3'13	22 nm
Intel® Celeron® Processor G Series	Desktop	G1610	Launched	Q1'13	22 nm
Legacy Intel® Pentium® Processor	Mobile	518	End of Interactive Support		90 nm
Intel® Pentium® Processor 2000 Series	Mobile	2020M	Launched	Q3'12	22 nm
Legacy Intel® Pentium® Processor	Mobile	773	End of Interactive Support		90 nm
Intel® Pentium® Processor 3000 Series	Mobile	3825U	Launched	Q1'15	14 nm
Intel® Pentium® Processor 4000 Series	Mobile	4405U	Launched	Q3'15	14 nm
Intel® Pentium® Processor N Series	Mobile	N3710	Launched	Q1'16	14 nm
Intel® Quark™ SE C1000 Microcontroller Series	Embedded	C1000	Launched	Q4'15	

Hình 1: Bảng khái quát số liệu

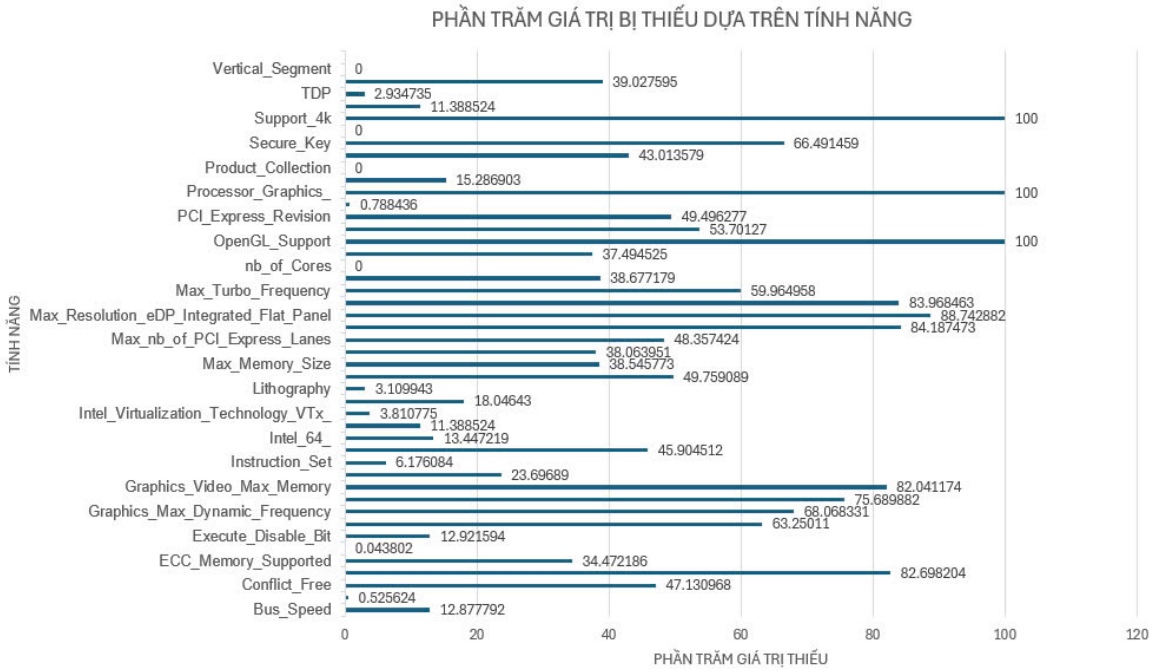
	count	unique	top	freq	mean	std	min	25%	50%	75%	max
Product_Collection	2283	75	Legacy Intel® Core™ Processors	375	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Vertical_Segment	2283	4	Mobile	760	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Processor_Number	1934	1892	330	4	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Status	2283	4	Launched	1043	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Launch_Date	1871	67	Q3'13	114	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Lithography	2212	10	22 nm	546	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Recommended_Customer_Price	1301	547	$281.00	39	NaN	NaN	NaN	NaN	NaN	NaN	NaN
nb_of_Cores	2283.0	NaN	NaN	NaN	4.066579	6.329884	1.0	1.0	2.0	4.0	72.0
nb_of_Threads	1427.0	NaN	NaN	NaN	8.728101	9.132518	1.0	4.0	4.0	8.0	56.0
Processor_Base_Frequency	2265	114	2.00 GHz	131	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Max_Turbo_Frequency	914	51	3.70 GHz	76	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Cache	2271	110	3 MB SmartCache	209	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Bus_Speed	1989	34	5 GT/s DMI	279	NaN	NaN	NaN	NaN	NaN	NaN	NaN
TDP	2216	250	35 W	238	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Embedded_Options_Available	2282	2	No	1724	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Conflict_Free	1207	1	Yes	1207	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Max_Memory_Size	1403	37	32 GB	404	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Memory_Types	1400	103	DDR3 1066/1333	143	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Max_nb_of_Memory_Channels	1414.0	NaN	NaN	NaN	2.615276	1.470327	1.0	2.0	2.0	3.0	16.0
Max_Memory_Bandwidth	1147	39	25.6 GB/s	390	NaN	NaN	NaN	NaN	NaN	NaN	NaN
ECC_Memory_Supported	1496	2	No	776	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Processor_Graphics_	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Graphics_Base_Frequency	839	28	350 MHz	238	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Graphics_Max_Dynamic_Frequency	729	23	1.10 GHz	143	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Graphics_Video_Max_Memory	410	8	2 GB	134	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Graphics_Output	555	17	eDP/DP/HDMI/DVI	139	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Support_4k	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Max_Resolution_HDMI	366	19	4096x2304@24Hz	181	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Max_Resolution_DP	361	8	4096x2304@60Hz	163	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Max_Resolution_eDP_Integrated_Flat_Panel	257	8	4096x2304@60Hz	149	NaN	NaN	NaN	NaN	NaN	NaN	NaN
DirectX_Support	395	6	12	165	NaN	NaN	NaN	NaN	NaN	NaN	NaN
OpenGL_Support	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
PCI_Express_Revision	1153.0	NaN	NaN	NaN	2.584562	0.493011	2.0	2.0	3.0	3.0	3.0
PCI_Express_Configurations_	1057	55	x4, x8, x16	190	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Max_nb_of_PCI_Express_Lanes	1179.0	NaN	NaN	NaN	20.399491	12.868963	0.0	16.0	16.0	32.0	48.0
T	2023	259	100°C	431	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Intel_Hyper_Threading_Technology_	2023	2	Yes	1138	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Intel_Virtualization_Technology_VTx_	2196	3	Yes	1631	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Intel_64_	1976	2	Yes	1703	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Instruction_Set	2142	3	64-bit	1692	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Instruction_Set_Extensions	1235	26	SSE4.1/4.2, AVX 2.0	332	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Idle_States	1742	2	Yes	1443	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Thermal_Monitoring_Technologies	1392	2	Yes	1289	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Secure_Key	765	2	Yes	718	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Execute_Disable_Bit	1988	2	Yes	1816	NaN	NaN	NaN	NaN	NaN	NaN	NaN

Hình 2: Tổng kết dữ liệu

3.2 Đánh giá phần trăm dữ liệu khuyết

Để củng cố cho phân tích dữ liệu, nhóm tác giả sẽ chuẩn hóa các giá trị bị thiếu bằng cách chuyển chúng thành "NA". Hiện tại, các dữ liệu khuyết này xuất hiện dưới nhiều dạng khác nhau như chuỗi rỗng (""), ký tự xuống dòng ("\n"), và các định dạng khác.

Ta có thể dễ dàng thể hiện phần trăm các giá trị khuyết dựa trên chức năng bằng biểu đồ cột ngang trong Hình 3.



Hình 3: Phần trăm Dữ liệu Khuyết dựa theo Tính năng

Như đã thấy, một số tính năng như *Support_4k, Processor_Graphics,...* thiếu dữ liệu hoặc có phần trăm thiếu dữ liệu cao. Vì vậy ta sẽ chỉ giữ lại những tính năng có phần trăm thiếu dữ liệu thấp hoặc không khuyết cho các phần tiếp theo. Đây là những thành phần tính năng có khả năng cao ảnh hưởng đến giá thành CPU cuối cùng như (Product_Collection, Vertical_Segment, Launch_Date, Lithography, Recommended_Customer_Price, nb_of_Cores, nb_of_Threads, Processor_Base_Frequency,Cache, Instruction_Set, TDP, Max_Memory_Size)

3.3 Chuyển đổi dữ liệu

Khi chuẩn bị dữ liệu cho thống kê mô tả và thống kê suy luận, việc tinh chỉnh và chuyển đổi các tính năng đã chọn là điều rất quan trọng để nâng cao tính khả dụng của chúng. Nhóm sẽ phân đoạn dữ liệu trong mỗi đặc

trưng thành các nhóm hoặc khoảng giá trị riêng biệt, có ý nghĩa. Quá trình chuyển đổi và phân loại dữ liệu này giúp việc trực quan hóa và phân tích trở nên dễ dàng hơn. Bằng cách tổ chức dữ liệu thành các nhóm hoặc khoảng rõ ràng, ta có thể tạo ra các biểu đồ, đồ thị và hình vẽ hiệu quả hơn, giúp truyền đạt rõ ràng các khuynh hướng và mô hình trong dữ liệu. Ngoài ra ta cũng sẽ dùng một hàm hỗ trợ để chuyển đổi dữ liệu bất đối xứng cho chúng phân phối đối xứng và trực quan hơn. Hàm hỗ trợ mà nhóm dùng là hàm **Chuyển đổi Logarithm** định nghĩa như sau:

$$L = log(X + 1)$$

Ý nghĩa của việc cộng thêm một đơn vị vào phía sau là để đảm bảo không có kết quả bằng 0.

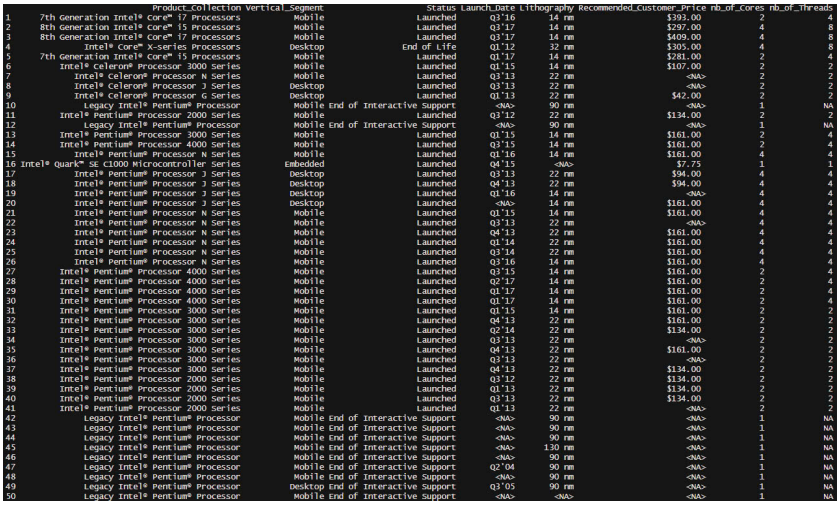
- **Biến Product_Collection**:
  Đối với biến này, ta sẽ phân tách theo các dòng máy chính bao gồm: Celeron, Pentium, Quark, Atom, Itanium, Xeon, Core. Nếu sản phẩm có chứa tên "Legacy", nhóm sẽ thêm từ này vào trước tên sản phẩm như một tiền tố.

- **Biến Launch_Date**:
  Lấy 2 ký tự cuối và chuyển qua cột mới **Year**. Các hàng không có giá trị ngày phát hành hoặc NA sẽ được chắt lọc.

- **Biến Lithography**:
  Chuyển về kiểu dữ liệu số bằng cách bỏ đi đơn vị đo. Sau đó đối với các giá trị khuyết ta tiến hành lắp vào bằng giá trị **median** dựa theo 2 đặc trưng **Product_Collection** và **Year**.

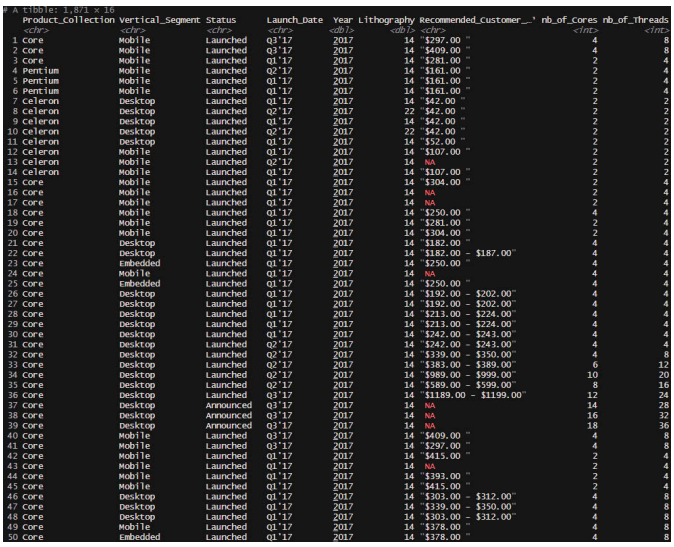
- **Biến Nb_of_Threads**:
  Ta sẽ có hướng xử lý tương tự như biến **Lithography**, bỏ đơn vị và thay các biến NA bằng **median** dựa theo **Product_Collection**.

Dữ liệu trước và sau khi thực hiện chuyển đổi (1):



	Product_Collection	Vertical_Segment	Status	Launch_Date	Lithography	Recommended_Customer_Price	nb_of_Cores	nb_of_Threads
1	7th Generation Intel® Core™ i7 Processors	Mobile	Launched	Q3'16	14 nm	$393.00	2	4
2	8th Generation Intel® Core™ i5 Processors	Mobile	Launched	Q3'17	14 nm	$297.00	4	8
3	8th Generation Intel® Core™ i7 Processors	Mobile	Launched	Q3'17	14 nm	$409.00	4	8
4	Intel® Core™ X-series Processors	Desktop	End of Life	Q1'12	32 nm	$305.00	4	8
5	7th Generation Intel® Core™ i5 Processors	Mobile	Launched	Q1'17	14 nm	$281.00	2	4
6	Intel® Celeron® Processor 3000 Series	Mobile	Launched	Q1'15	14 nm	$107.00	2	2
7	Intel® Celeron® Processor N Series	Mobile	Launched	Q3'13	22 nm	<NA>	2	2
8	Intel® Celeron® Processor J Series	Desktop	Launched	Q3'13	22 nm	<NA>	2	2
9	Intel® Celeron® Processor G Series	Desktop	Launched	Q1'13	22 nm	$42.00	2	2
10	Legacy Intel® Pentium® Processor	Mobile	End of Interactive Support	<NA>	90 nm	<NA>	1	NA
11	Intel® Pentium® Processor 2000 Series	Mobile	Launched	Q3'12	22 nm	$134.00	2	2
12	Legacy Intel® Pentium® Processor	Mobile	End of Interactive Support	<NA>	90 nm	<NA>	1	NA
13	Intel® Pentium® Processor 3000 Series	Mobile	Launched	Q1'15	14 nm	$161.00	2	4
14	Intel® Pentium® Processor 4000 Series	Mobile	Launched	Q3'15	14 nm	$161.00	2	4
15	Intel® Pentium® Processor N Series	Mobile	Launched	Q1'16	14 nm	$161.00	4	4
16	Intel® Quark™ SC C1000 Microcontroller Series	Embedded	Launched	Q4'15	<NA>	$7.75	1	1
17	Intel® Pentium® Processor J Series	Desktop	Launched	Q3'13	22 nm	$94.00	4	4
18	Intel® Pentium® Processor J Series	Desktop	Launched	Q4'13	22 nm	$94.00	4	4
19	Intel® Pentium® Processor J Series	Desktop	Launched	Q1'16	14 nm	<NA>	4	4
20	Intel® Pentium® Processor J Series	Desktop	Launched	<NA>	14 nm	$161.00	4	4
21	Intel® Pentium® Processor N Series	Mobile	Launched	Q1'15	14 nm	$161.00	4	4
22	Intel® Pentium® Processor N Series	Mobile	Launched	Q3'13	22 nm	<NA>	4	4
23	Intel® Pentium® Processor N Series	Mobile	Launched	Q4'13	22 nm	$161.00	4	4
24	Intel® Pentium® Processor N Series	Mobile	Launched	Q1'14	22 nm	$161.00	4	4
25	Intel® Pentium® Processor N Series	Mobile	Launched	Q3'14	22 nm	$161.00	4	4
26	Intel® Pentium® Processor N Series	Mobile	Launched	Q3'16	14 nm	$161.00	4	4
27	Intel® Pentium® Processor 4000 Series	Mobile	Launched	Q3'15	14 nm	$161.00	2	4
28	Intel® Pentium® Processor 4000 Series	Mobile	Launched	Q2'17	14 nm	$161.00	2	4
29	Intel® Pentium® Processor 4000 Series	Mobile	Launched	Q1'17	14 nm	$161.00	2	4
30	Intel® Pentium® Processor 4000 Series	Mobile	Launched	Q1'17	14 nm	$161.00	2	4
31	Intel® Pentium® Processor 3000 Series	Mobile	Launched	Q1'15	14 nm	$161.00	2	2
32	Intel® Pentium® Processor 3000 Series	Mobile	Launched	Q4'13	22 nm	$161.00	2	2
33	Intel® Pentium® Processor 3000 Series	Mobile	Launched	Q2'14	22 nm	$134.00	2	2
34	Intel® Pentium® Processor 3000 Series	Mobile	Launched	Q3'13	22 nm	<NA>	2	2
35	Intel® Pentium® Processor 3000 Series	Mobile	Launched	Q4'13	22 nm	$161.00	2	2
36	Intel® Pentium® Processor 3000 Series	Mobile	Launched	Q3'13	22 nm	<NA>	2	2
37	Intel® Pentium® Processor 3000 Series	Mobile	Launched	Q4'13	22 nm	$134.00	2	2
38	Intel® Pentium® Processor 2000 Series	Mobile	Launched	Q3'12	22 nm	$134.00	2	2
39	Intel® Pentium® Processor 2000 Series	Mobile	Launched	Q1'13	22 nm	$134.00	2	2
40	Intel® Pentium® Processor 2000 Series	Mobile	Launched	Q3'13	22 nm	$134.00	2	2
41	Intel® Pentium® Processor 2000 Series	Mobile	Launched	Q1'13	22 nm	<NA>	2	2
42	Legacy Intel® Pentium® Processor	Mobile	End of Interactive Support	<NA>	90 nm	<NA>	1	NA
43	Legacy Intel® Pentium® Processor	Mobile	End of Interactive Support	<NA>	90 nm	<NA>	1	NA
44	Legacy Intel® Pentium® Processor	Mobile	End of Interactive Support	<NA>	90 nm	<NA>	1	NA
45	Legacy Intel® Pentium® Processor	Mobile	End of Interactive Support	<NA>	130 nm	<NA>	1	NA
46	Legacy Intel® Pentium® Processor	Mobile	End of Interactive Support	<NA>	90 nm	<NA>	1	NA
47	Legacy Intel® Pentium® Processor	Mobile	End of Interactive Support	Q2'04	90 nm	<NA>	1	NA
48	Legacy Intel® Pentium® Processor	Mobile	End of Interactive Support	<NA>	90 nm	<NA>	1	NA
49	Legacy Intel® Pentium® Processor	Desktop	End of Interactive Support	Q3'05	90 nm	<NA>	1	NA
50	Legacy Intel® Pentium® Processor	Mobile	End of Interactive Support	<NA>	<NA>	<NA>	1	NA

Hình 4: Bảng dữ liệu trước quá trình chuyển đổi



A tibble: 1,871 x 16

	Product_Collection	Vertical_Segment	Status	Launch_Date	Year	Lithography	Recommended_Customer_...	nb_of_Cores	nb_of_Threads
	<chr>	<chr>	<chr>	<chr>	<dbl>	<dbl>	<chr>	<int>	<int>
1	Core	Mobile	Launched	Q3'17	2017	14	"$297.00 "	4	8
2	Core	Mobile	Launched	Q3'17	2017	14	"$409.00 "	4	8
3	Core	Mobile	Launched	Q1'17	2017	14	"$281.00 "	2	4
4	Pentium	Mobile	Launched	Q2'17	2017	14	"$161.00 "	2	4
5	Pentium	Mobile	Launched	Q1'17	2017	14	"$161.00 "	2	4
6	Pentium	Mobile	Launched	Q1'17	2017	14	"$161.00 "	2	4
7	Celeron	Desktop	Launched	Q1'17	2017	14	"$42.00 "	2	2
8	Celeron	Desktop	Launched	Q2'17	2017	22	"$42.00 "	2	2
9	Celeron	Desktop	Launched	Q1'17	2017	14	"$42.00 "	2	2
10	Celeron	Desktop	Launched	Q2'17	2017	22	"$42.00 "	2	2
11	Celeron	Desktop	Launched	Q1'17	2017	14	"$52.00 "	2	2
12	Celeron	Mobile	Launched	Q1'17	2017	14	"$107.00 "	2	2
13	Celeron	Mobile	Launched	Q2'17	2017	14	NA	2	2
14	Celeron	Mobile	Launched	Q1'17	2017	14	"$107.00 "	2	2
15	Core	Mobile	Launched	Q1'17	2017	14	"$304.00 "	2	4
16	Core	Mobile	Launched	Q1'17	2017	14	NA	2	4
17	Core	Mobile	Launched	Q1'17	2017	14	NA	2	4
18	Core	Mobile	Launched	Q1'17	2017	14	"$250.00 "	4	4
19	Core	Mobile	Launched	Q1'17	2017	14	"$281.00 "	2	4
20	Core	Mobile	Launched	Q1'17	2017	14	"$304.00 "	2	4
21	Core	Desktop	Launched	Q1'17	2017	14	"$182.00 "	4	4
22	Core	Desktop	Launched	Q1'17	2017	14	"$182.00 - $187.00"	4	4
23	Core	Embedded	Launched	Q1'17	2017	14	"$250.00 "	4	4
24	Core	Mobile	Launched	Q1'17	2017	14	NA	4	4
25	Core	Embedded	Launched	Q1'17	2017	14	"$250.00 "	4	4
26	Core	Desktop	Launched	Q1'17	2017	14	"$192.00 - $202.00"	4	4
27	Core	Desktop	Launched	Q1'17	2017	14	"$192.00 - $202.00"	4	4
28	Core	Desktop	Launched	Q1'17	2017	14	"$213.00 - $224.00"	4	4
29	Core	Desktop	Launched	Q1'17	2017	14	"$213.00 - $224.00"	4	4
30	Core	Desktop	Launched	Q1'17	2017	14	"$242.00 - $243.00"	4	4
31	Core	Desktop	Launched	Q2'17	2017	14	"$242.00 - $243.00"	4	4
32	Core	Desktop	Launched	Q2'17	2017	14	"$339.00 - $350.00"	4	8
33	Core	Desktop	Launched	Q2'17	2017	14	"$383.00 - $389.00"	6	12
34	Core	Desktop	Launched	Q2'17	2017	14	"$989.00 - $999.00"	10	20
35	Core	Desktop	Launched	Q2'17	2017	14	"$589.00 - $599.00"	8	16
36	Core	Desktop	Launched	Q3'17	2017	14	"$1189.00 - $1199.00"	12	24
37	Core	Desktop	Announced	Q3'17	2017	14	NA	14	28
38	Core	Desktop	Announced	Q3'17	2017	14	NA	16	32
39	Core	Desktop	Announced	Q3'17	2017	14	NA	18	36
40	Core	Mobile	Launched	Q3'17	2017	14	"$409.00 "	4	8
41	Core	Mobile	Launched	Q3'17	2017	14	"$297.00 "	4	8
42	Core	Mobile	Launched	Q1'17	2017	14	"$415.00 "	2	4
43	Core	Mobile	Launched	Q1'17	2017	14	NA	2	4
44	Core	Mobile	Launched	Q1'17	2017	14	"$393.00 "	2	4
45	Core	Mobile	Launched	Q1'17	2017	14	"$415.00 "	2	4
46	Core	Desktop	Launched	Q1'17	2017	14	"$303.00 - $312.00"	4	8
47	Core	Desktop	Launched	Q1'17	2017	14	"$339.00 - $350.00"	4	8
48	Core	Desktop	Launched	Q1'17	2017	14	"$303.00 - $312.00"	4	8
49	Core	Mobile	Launched	Q1'17	2017	14	"$378.00 "	4	8
50	Core	Embedded	Launched	Q1'17	2017	14	"$378.00 "	4	8

Hình 5: Bảng dữ liệu sau quá trình chuyển đổi

- **Biến Processor_Base_Frequency**:
  Do biến này có hai đơn vị đo tần số là MHz và GHz, ta sẽ chuyển đổi thành đơn vị chung là MHz (1 GHz = 1000 MHz). Các biến không có

giá trị sẽ được lồng vào bằng **median** theo biến **Vertical Segment**.

- **Biến Cache**:
  Biến này sẽ được nhóm phân thành 2 cột dữ liệu với một cột chứa kích thước và cột còn lại chứa kiểu Cache. Các chuỗi "NA" sẽ được loại bỏ ở mỗi cột sau khi đã tách dữ liệu. Và tương tự với biến **Processor_Base_Frequency** ta cũng sẽ chuyển đổi đơn vị đo kích thước thành 1 đơn vị thống nhất là KB. Bên cạnh đó do dữ liệu có sự lệch nên ta sẽ dùng hàm chuyển đổi log đã nêu trước và chuyển đổi các dữ liệu này thành một cột mới **Normalized_Cache**.

- **Biến Instruction_Set**:
  Biến này sẽ được loại bỏ các ký tự chữ và giữ nguyên số liệu số học. Các dữ liệu khuyết sẽ được ghép vào bằng xu hướng của chúng theo dữ liệu **Product_Collection**.

- **Biến TDP**:
  Đối với biến này ta cũng sẽ chắt lọc và chỉ lấy những dữ liệu số. Còn đối với dữ liệu khuyết ta sẽ trực tiếp loại bỏ vì tỉ lệ khuyết ở biến này khá thấp và không ảnh hưởng nhiều tới dữ liệu phân tích chung.

- **Biến Max_Memory_Size**:
  Đơn vị của biến sẽ được chuẩn hóa về cùng 1 đơn vị đo là GB. Những dữ liệu khuyết sẽ được bù vào bằng **median** dựa theo **Product_Collection**, **Vertical_Segment** và **Year**. Ngoài ra biến này cũng được chuyển đổi theo hàm log.

- **Biến Max_nb_of_Memory_Channels** và
  Biến Max_Memory_Bandwidth:
  Đối với các dữ liệu này chỉ cần xử lý dữ liệu khuyết bằng cách thêm vào chỗ trống các **median** dựa theo tính năng **Product_Collection** và **Vertical_Segment**. Riêng biến **Max_Memory_Bandwidth** thì cần tiền xử lý xóa đơn vị.

- **Biến Recommended_Customer_Price**:
  Đầu tiên ta xóa ký tự **$** để thực hiện các thao tác xử lý số liệu số học. Kế tiếp ta sẽ tính dữ liệu trung bình **mean** cho các biến **Recommended_Customer_Price** có dạng vùng để đảm bảo kết quả cho ra sẽ chỉ là 1 số liệu duy nhất. Đến bước xử lý dữ liệu khuyết ta sẽ dựa theo biến **Product_Collection** và sẽ dùng phương pháp lấp vào giá trị gần nhất không khuyết theo từng nhóm **Product_Collection** theo cả hai chiều giá thấp hơn và giá cao hơn. Cuối cùng, ta sẽ đổi kiểu dữ liệu thành dạng **double** để thuận tiện hơn cho các tính toán ở phần sau.

Dữ liệu trước và sau khi thực hiện chuyển đổi (2):

	Recommended_Customer_Price	Processor_Base_Frequency	Cache	Instruction_Set	TDP	Max_Memory_Size	Max_nb_of_Memory_Channels	Max_Memory_Bandwidth
1	$393.00	1.30 GHz	4 MB SmartCache	64-bit	4.5 W	16 GB	2	29.8 GB/s
2	$297.00	1.60 GHz	6 MB SmartCache	64-bit	15 W	32 GB	2	34.1 GB/s
3	$409.00	1.80 GHz	8 MB SmartCache	64-bit	15 W	32 GB	2	34.1 GB/s
4	$305.00	3.60 GHz	10 MB SmartCache	64-bit	130 W	64.23 GB	4	51.2 GB/s
5	$281.00	1.20 GHz	4 MB SmartCache	64-bit	4.5 W	16 GB	2	29.8 GB/s
6	$107.00	1.50 GHz	2 MB	64-bit	15 W	16 GB	2	25.6 GB/s
7	<NA>	1.46 GHz	1 MB	64-bit	4.3 W	4 GB	1	<NA>
8	<NA>	2.41 GHz	1 MB L2	64-bit	10 W	8 GB	2	<NA>
9	$42.00	2.60 GHz	2 MB SmartCache	64-bit	55 W	32 GB	2	21 GB/s
10	<NA>	2.80 GHz	1 MB L2	32-bit	88 W	<NA>	NA	<NA>
11	$134.00	2.40 GHz	2 MB SmartCache	64-bit	35 W	32 GB	2	25.6 GB/s
12	<NA>	1.30 GHz	2 MB L2	32-bit	5.5 W	<NA>	NA	<NA>
13	$161.00	1.90 GHz	2 MB	64-bit	15 W	16 GB	2	25.6 GB/s
14	$161.00	2.10 GHz	2 MB SmartCache	64-bit	15 W	32 GB	2	34.1 GB/s
15	$161.00	1.60 GHz	2 MB L2	64-bit	6 W	8 GB	2	<NA>
16	$7.75	32 MHz	8 KB	32-bit	<NA>	<NA>	NA	<NA>
17	$94.00	2.41 GHz	2 MB L2	64-bit	10 W	8 GB	2	<NA>
18	$94.00	2.41 GHz	2 MB L2	64-bit	10 W	8 GB	2	21.3 GB/s
19	<NA>	1.60 GHz	2 MB L2	64-bit	6.5 W	8 GB	2	<NA>
20	$161.00	1.50 GHz	2 MB	64-bit	10 W	8 GB	2	<NA>
21	$161.00	1.60 GHz	2 MB L2	64-bit	6 W	8 GB	2	<NA>
22	<NA>	2.00 GHz	2 MB	64-bit	7.5 W	8 GB	2	<NA>
23	$161.00	2.17 GHz	2 MB	64-bit	7.5 W	8 GB	2	<NA>
24	$161.00	2.16 GHz	2 MB	64-bit	7.5 W	8 GB	2	<NA>
25	$161.00	2.16 GHz	2 MB L2	64-bit	7.5 W	8 GB	2	21.32 GB/s
26	$161.00	1.10 GHz	2 MB L2	64-bit	6 W	8 GB	2	<NA>
27	$161.00	1.50 GHz	2 MB SmartCache	64-bit	6 W	16 GB	2	29.8 GB/s
28	$161.00	1.60 GHz	2 MB SmartCache	64-bit	6 W	16 GB	2	29.8 GB/s
29	$161.00	1.50 GHz	2 MB SmartCache	64-bit	6 W	16 GB	2	29.8 GB/s
30	$161.00	2.30 GHz	2 MB SmartCache	64-bit	15 W	32 GB	2	34.1 GB/s
31	$161.00	1.90 GHz	2 MB	64-bit	15 W	16 GB	2	25.6 GB/s
32	$161.00	1.20 GHz	2 MB SmartCache	64-bit	11.5 W	16 GB	2	25.6 GB/s
33	$134.00	2.40 GHz	2 MB	64-bit	37 W	32 GB	2	25.6 GB/s
34	<NA>	1.20 GHz	2 MB SmartCache	64-bit	11.5 W	16 GB	2	25.6 GB/s
35	$161.00	1.70 GHz	2 MB SmartCache	64-bit	15 W	16 GB	2	25.6 GB/s
36	<NA>	1.70 GHz	2 MB SmartCache	64-bit	15 W	16 GB	2	25.6 GB/s
37	$134.00	2.30 GHz	2 MB SmartCache	64-bit	37 W	32 GB	2	25.6 GB/s
38	$134.00	1.80 GHz	2 MB SmartCache	64-bit	17 W	32 GB	2	25.6 GB/s
39	$134.00	2.50 GHz	2 MB SmartCache	64-bit	35 W	32 GB	2	25.6 GB/s
40	$134.00	1.90 GHz	2 MB SmartCache	64-bit	17 W	32 GB	2	25.6 GB/s
41	<NA>	1.10 GHz	2 MB SmartCache	64-bit	10 W	32 GB	2	25.6 GB/s
42	<NA>	1.20 GHz	2 MB L2	32-bit	5.5 W	<NA>	NA	<NA>
43	<NA>	1.10 GHz	2 MB L2	32-bit	5 W	<NA>	NA	<NA>
44	<NA>	1.00 GHz	2 MB L2	32-bit	5 W	<NA>	NA	<NA>
45	<NA>	1.10 GHz	1 MB L2	32-bit	7 W	<NA>	NA	<NA>
46	<NA>	1.60 GHz	2 MB L2	32-bit	7.5 W	<NA>	NA	<NA>
47	<NA>	1.40 GHz	2 MB L2	32-bit	7.5 W	<NA>	NA	<NA>
48	<NA>	1.50 GHz	2 MB L2	32-bit	7.5 W	<NA>	NA	<NA>
49	<NA>	3.80 GHz	1 MB L2	64-bit	115 W	<NA>	NA	<NA>
50	<NA>	800 MHz	512 KB L2	32-bit	5.9 W	<NA>	NA	<NA>

Hình 6: Bảng dữ liệu trước khi chuyển đổi

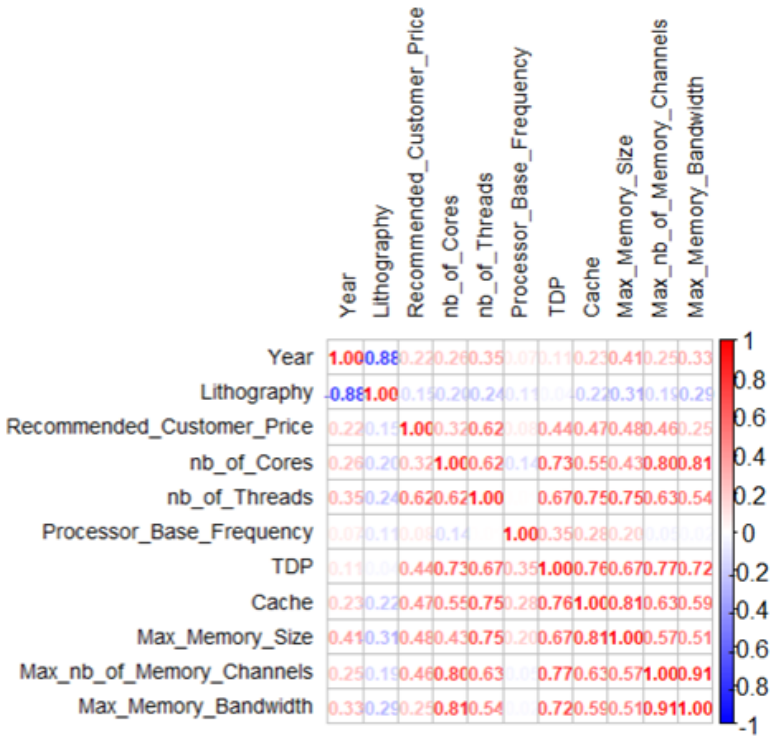
	Recommended_Customer_Price	Processor_Base_Frequency	Cache	Normalized_Cache	Instruction_Set	TDP	Max_Memory_Size	Max_nb_of_Memory_Channels	Max_Memory_Bandwidth
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	297	1600	6144	8.72	64	15	3.50	2	34.1
2	409	1800	8192	9.01	64	15	3.50	2	34.1
3	281	1200	4096	8.32	64	4.5	2.83	2	29.8
4	161	1600	2048	7.63	64	6	2.83	2	29.8
5	161	1500	2048	7.63	64	6	2.83	2	29.8
6	161	2300	2048	7.63	64	15	3.50	2	34.1
7	42	2900	2048	7.63	64	51	4.17	2	34.1
8	42	2900	2048	7.63	64	54	4.17	2	34.1
9	42	2700	2048	7.63	64	35	4.17	2	34.1
10	42	2700	2048	7.63	64	35	4.17	2	34.1
11	52	3000	2048	7.63	64	51	4.17	2	34.1
12	107	1800	2048	7.63	64	15	3.50	2	34.1
13	107	1500	2048	7.63	64	6	2.83	2	29.8
14	107	2200	2048	7.63	64	15	3.50	2	34.1
15	304	2200	4096	8.32	64	15	3.50	2	34.1
16	250	3100	4096	8.32	64	28	3.50	2	34.1
17	250	3300	4096	8.32	64	28	3.50	2	34.1
18	250	2500	6144	8.72	64	45	4.17	2	34.1
19	281	2600	3072	8.03	64	15	3.50	2	34.1
20	304	2300	4096	8.32	64	15	3.50	2	34.1
21	182	3000	6144	8.72	64	65	4.17	2	34.1
22	184.	2400	6144	8.72	64	35	4.17	2	34.1
23	250	2900	6144	8.72	64	45	4.17	2	34.1
24	250	2800	6144	8.72	64	45	4.17	2	34.1
25	250	2100	6144	8.72	64	25	4.17	2	34.1
26	197	2700	6144	8.72	64	35	4.17	2	34.1
27	197	3400	6144	8.72	64	65	4.17	2	34.1
28	218.	3500	6144	8.72	64	65	4.17	2	34.1
29	218.	2800	6144	8.72	64	35	4.17	2	34.1
30	242.	3800	6144	8.72	64	91	4.17	2	34.1
31	242.	4000	6144	8.72	64	112	4.17	2	34.1
32	344.	4300	8192	9.01	64	112	4.17	2	34.1
33	409	1900	8192	9.01	64	15	3.50	2	34.1
34	297	1700	6144	8.72	64	15	3.50	2	34.1
35	415	2400	4096	8.32	64	15	3.50	2	34.1
36	393	3500	4096	8.32	64	28	3.50	2	34.1
37	393	2800	4096	8.32	64	15	3.50	2	34.1
38	415	2500	4096	8.32	64	15	3.50	2	34.1
39	308.	2900	8192	9.01	64	35	4.17	2	34.1
40	344.	4200	8192	9.01	64	91	4.17	2	34.1
41	308.	3600	8192	9.01	64	65	4.17	2	34.1
42	378	2800	6144	8.72	64	45	4.17	2	34.1
43	378	3000	8192	9.01	64	45	4.17	2	34.1
44	378	2900	8192	9.01	64	45	4.17	2	34.1
45	378	2900	8192	9.01	64	45	4.17	2	34.1
46	568	3100	8192	9.01	64	45	4.17	2	34.1
47	39	1500	1024	6.93	64	7	2.83	1	10.7
48	64	1400	2048	7.63	64	10	3.50	2	23.5
49	198	3000	8192	9.01	64	72	4.17	2	37.5
50	218.	3300	8192	9.01	64	73	4.17	2	37.5

Hình 7: Bảng dữ liệu sau khi chuyển đổi

4 Thống kê mô tả

Chia khung dữ liệu đã xử lý thành hai bảng Dữ liệu Phân loại và Dữ liệu số để xử lý. Bảng dữ liệu phân loại bao gồm: Product_Collection, Vertical_Segment, Status, Cache_Type, Instruction_Set và phần còn lại nằm trong bảng Dữ liệu Số. Đối với dữ liệu số, chúng tôi tính toán các thống kê khác nhau bao gồm số lượng, giá trị trung bình, độ lệch chuẩn, giá trị tối thiểu, tứ phân vị thứ nhất, trung vị, tứ phân vị thứ ba và giá trị tối đa cho từng biến số. Đối với dữ liệu phân loại, chúng tôi tính toán số lượng giá trị không phải NA, số lượng giá trị duy nhất, mode (giá trị xuất hiện thường xuyên nhất) và tần suất của mode cho từng biến phân loại.

4.1 Ma trận tương quan:



Hình 8: Bảng dữ liệu sau khi chuyển đổi

1. **Năm (Year)**
   - Tương quan âm mạnh với **Lithography** (-0.88): Khi công nghệ phát triển theo thời gian, kích thước transistor (Lithography) có xu hướng giảm (ví dụ: từ 32nm xuống 14nm).
   - **Giải thích**: Công nghệ sản xuất càng tiên tiến (Lithography nhỏ hơn) thường đi kèm với bộ nhớ đệm hiệu quả hơn.
2. **Giá Khuyến Nghị (Recommended Customer Price)**

- Giá có xu hướng tăng khi các đặc tính khác tăng, ngoại trừ Lithography.
- Tương quan mạnh nhất là 0.62 (với **nb_of_Thread**), các biến còn lại dao động từ 0.1 đến 0.62.
- **Giải thích**:
  - Cache và số lõi/luồng ảnh hưởng rõ rệt đến giá.
  - Lithography càng nhỏ (công nghệ tiên tiến) thường làm giá tăng, nhưng ma trận không phản ánh điều này do các yếu tố khác (ví dụ: phân khúc thị trường) chi phối.

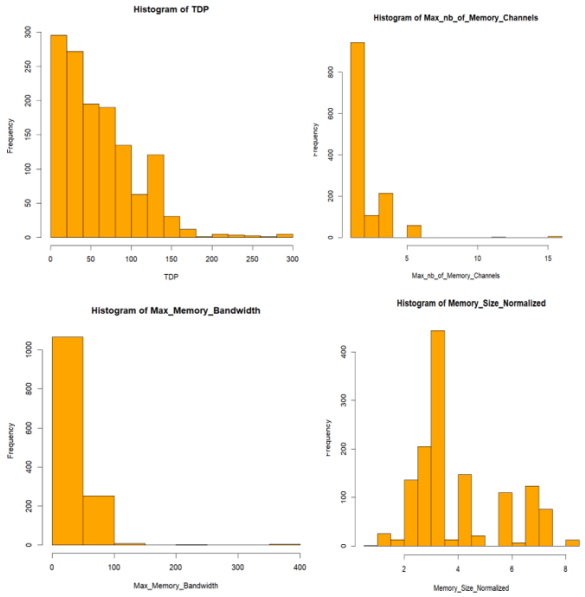
3. **Số Lõi (Cores) và Luồng (Threads)**
   - Tương quan nội tại mạnh (0.62): CPU nhiều lõi thường hỗ trợ đa luồng.
   - Tương quan với **Cache** (0.55 và 0.75): CPU nhiều lõi/luồng thường có bộ nhớ đệm lớn hơn để xử lý tác vụ song song.
   - **Giải thích**: Kiến trúc CPU hiện đại ưu tiên cân bằng giữa số lõi, luồng và bộ nhớ đệm.
4. **Hiệu Suất Nhiệt (TDP)**
   - Tương quan $> 35\%$ với hầu hết đặc tính (trừ **Year** và **Lithography**).
   - Xu hướng: CPU hiệu năng cao (nhiều lõi, băng thông lớn) tiêu thụ nhiều điện năng hơn.
   - **Giải thích**: TDP phản ánh sự đánh đổi giữa hiệu suất và tiêu thụ năng lượng.
5. **Tần Số Cơ Bản (Processor Base Frequency)**
   - Tương quan dương yếu với các biến khác.

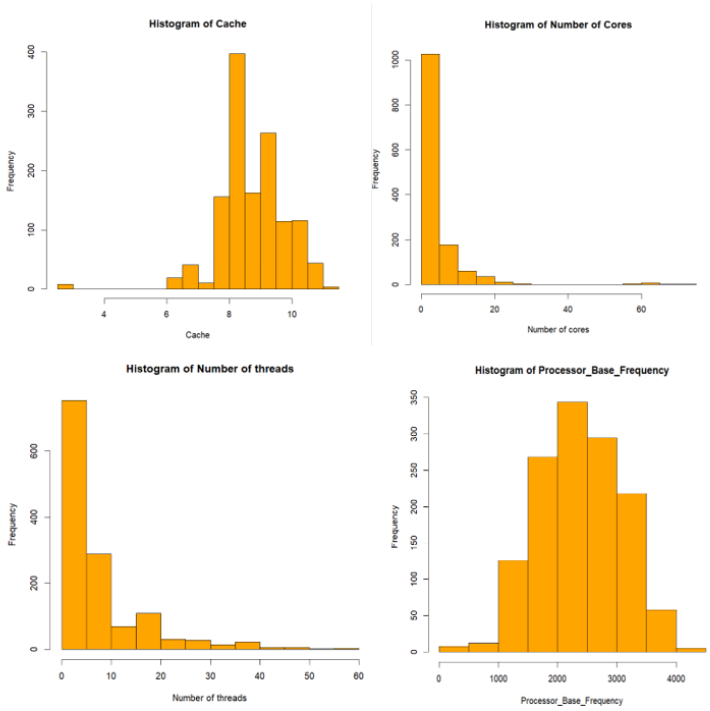
- **Nguyên nhân**: Tần số không phải yếu tố quyết định duy nhất đến hiệu năng (còn phụ thuộc vào kiến trúc, số lõi...).

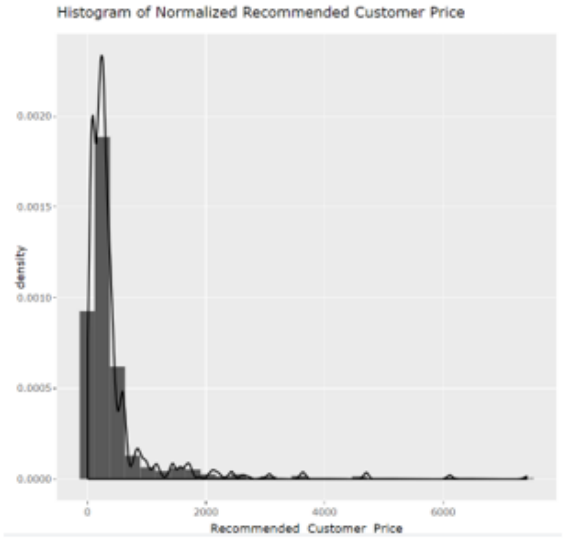
6. **Lithography và Cache**

- Lithography có tương quan âm với **Cache** (-0.22): Công nghệ sản xuất càng nhỏ (ví dụ: 7nm) cho phép tích hợp bộ nhớ đệm hiệu quả hơn trên cùng diện tích.
- **Memory (Bộ nhớ)**:
  - Các biến liên quan (kênh bộ nhớ, băng thông) tương quan dương mạnh với nhau, nhưng không liên quan đến Lithography.
- **Giải thích**:
  - Lithography ảnh hưởng đến thiết kế vật lý, trong khi bộ nhớ phụ thuộc vào kiến trúc tổng thể.
  - Cache được tối ưu để bù đắp cho độ trễ bộ nhớ.

4.2 Histogram:







Nhận Xét Histogram Các Đặc Tính CPU

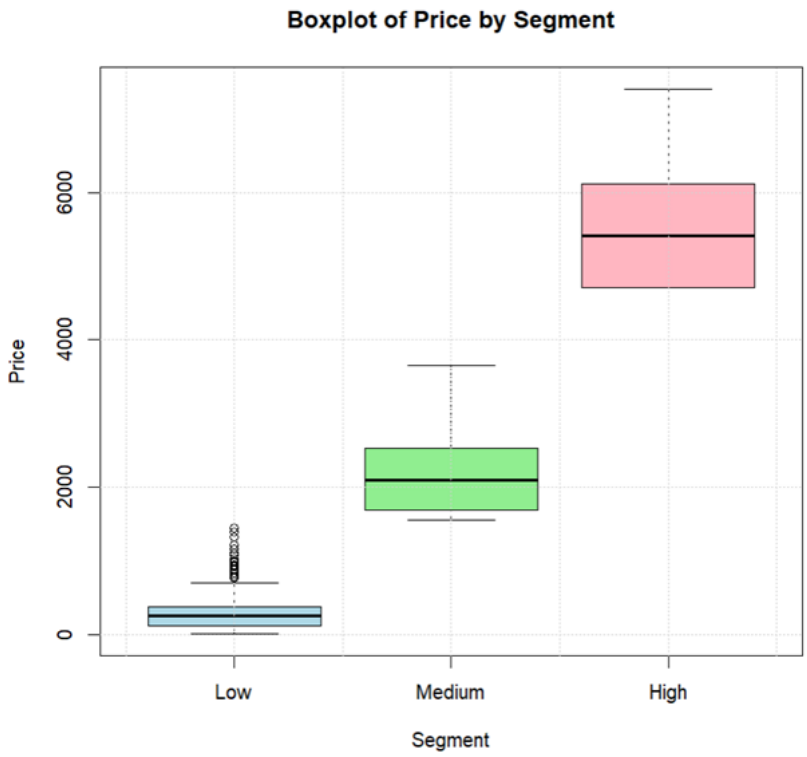
1. **Histogram của Số lõi (nb_of_cores) và Số luồng (nb_of_threads)**
   - Nhận xét: Cả hai đều lệch phải, phản ánh thị trường chủ yếu là

CPU 2-4 lõi (phổ biến cho nhu cầu cơ bản), một số ít CPU server/-workstation có nhiều lõi/luồng hơn kéo đuôi phân phối.

2. **Histogram của Băng thông bộ nhớ (Max_memory_Bandwidth) và Số kênh bộ nhớ (Memory_Channels)**
   - Nhận xét: Phân phối lệch phải, đa số CPU có băng thông và số kênh thấp (phục vụ nhu cầu phổ thông), một số ít CPU high-end kéo đuôi phân phối.
3. **Histogram của Tần số cơ bản (Processor Base Frequency)**
   - Nhận xét: Phân phối chuẩn (normal distribution) do mean ≈ median và dữ liệu tập trung đối xứng quanh trung tâm. Điều này phản ánh sự đa dạng cân đối của CPU từ hiệu năng thấp đến cao.
4. **Histogram của Giá khuyến nghị (Recommended Customer Price)**
   - Nhận xét: Phân phối lệch phải mạnh, phù hợp với thực tế thị trường: đa số người dùng mua CPU giá rẻ/phổ thông, chỉ một nhóm nhỏ cần CPU đắt tiền.
5. **Histogram của Bộ nhớ đệm (Cache)**
   - Nhận xét: Phân phối lệch trái nhẹ (left-skewed), cho thấy đa số CPU có cache xấp xỉ 8–8.4 KB, một số ít có cache nhỏ hơn kéo đuôi về bên trái.
6. **Histogram của Dung lượng bộ nhớ (Memory Size Normalized)**
   - Nhận xét: Đa số CPU hỗ trợ bộ nhớ khoảng 3 GB, phù hợp với nhu cầu cơ bản. Các giá trị cao hơn (4–8 GB) thường dành cho CPU chuyên dụng.
7. **Histogram của Công suất nhiệt (TDP)**

- Nhận xét: Phản ánh xu hướng ưu tiên CPU tiết kiệm năng lượng cho đại chúng, chỉ một số ít CPU hiệu năng cao có TDP lớn.

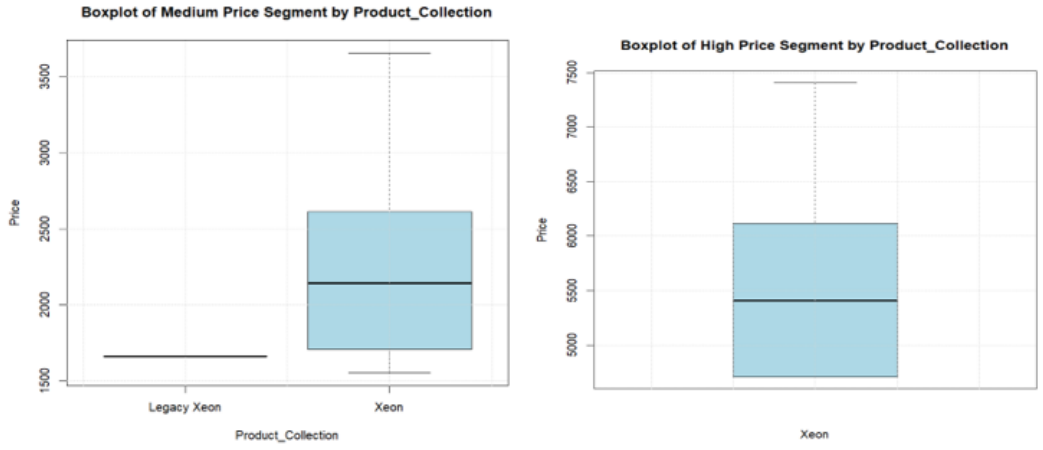
4.3 Box Plot:

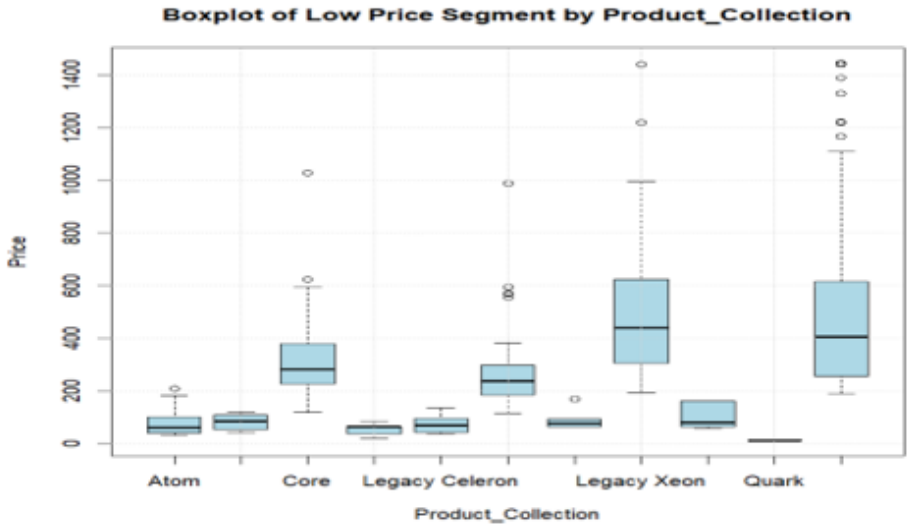


Hình trên làm nổi bật sự phân bố giá cả giữa ba phân khúc: Thấp, Trung bình và Cao.

Phân khúc Thấp (màu xanh dương) cho thấy mức giá tập trung ở vị trí thấp nhất, với biến động tối thiểu và một số ngoại lệ. Điều này cho thấy đa phần các sản phẩm trong nhóm này tập trung ở mức giá thấp, với một số ít trường hợp có giá cao hơn. Phân khúc Trung bình (màu xanh lá) thể hiện phạm vi giá ở mức trung bình, với độ phân tán rộng hơn so với phân khúc Thấp, phản ánh sự đa dạng lớn hơn trong nhóm này. Giá tập trung chủ yếu trong khoảng từ 2.000 đến 4.000 đơn vị. Phân khúc Cao (màu hồng) có phạm vi giá rộng nhất và các giá trị luôn ở mức cao, không có ngoại lệ. Điều

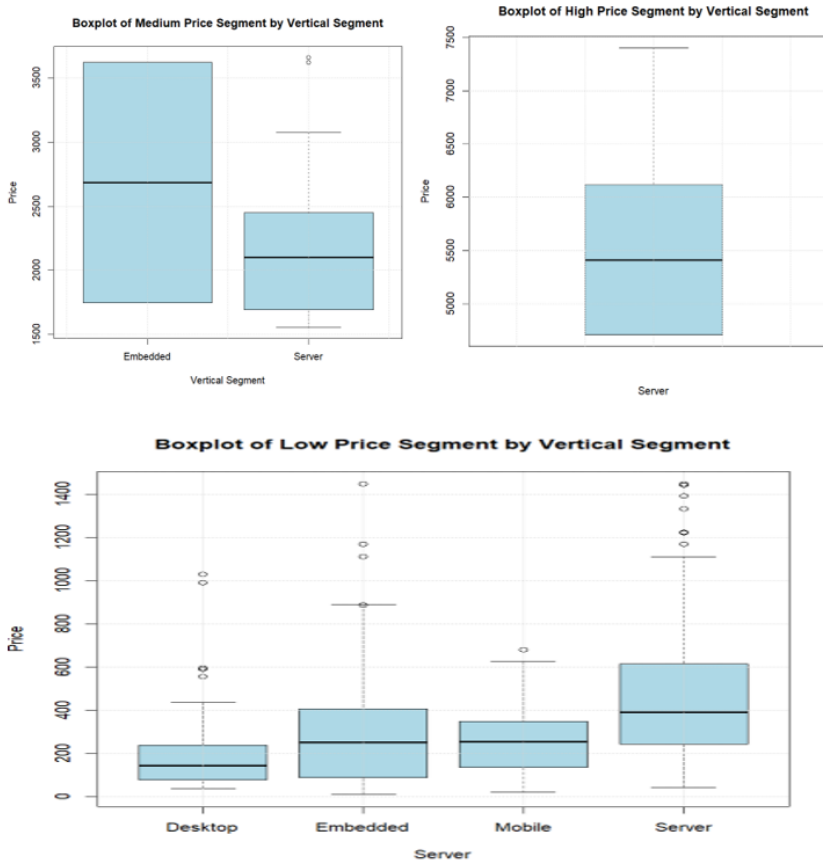
này cho thấy các sản phẩm thuộc phân khúc này thường có chất lượng cao và đồng nhất về cấu trúc giá. Tổng quan: Dữ liệu cho thấy sự phân tầng rõ rệt về giá giữa các phân khúc, trong đó các phân khúc cao hơn luôn đi kèm với mức giá và phạm vi giá lớn hơn.





Ba biểu đồ hộp trong hình trên thể hiện xu hướng giá cả giữa các phân khúc dọc được phân loại thành ba mức giá: trung bình, cao và thấp. Ở phân khúc trung bình, phân khúc dọc "Embedded" cho thấy biến động giá lớn

hơn và trung vị cao hơn so với "Server", điều này cho thấy sự đa dạng hơn về sản phẩm. Phân khúc giá cao tập trung hoàn toàn vào "Server", thể hiện sự dao động giá đáng kể với xu hướng nhẹ về giá cao hơn, được chỉ ra bởi sự phân tán đồng đều mà không có giá trị ngoại lệ. Ở phân khúc giá thấp, phân khúc dọc "Server" có trung vị cao hơn và biến động rộng hơn, chồng lấn với các phân khúc cao hơn, trong khi "Desktop" duy trì mức giá thấp và ổn định. "Embedded" và "Mobile" thể hiện mức giá trung bình với một số biến động, phản ánh chiến lược giá hỗn hợp. Nhìn chung, phân khúc "Server" nổi bật hơn ở các nhóm giá cao, trong khi "Desktop" thể hiện sự ổn định ở phân khúc thấp. Những xu hướng này minh họa sự khác biệt rõ ràng trong chiến lược định giá.



Các biểu đồ hộp trên so sánh các phân khúc giá giữa các nhóm sản phẩm

khác nhau, tiết lộ những xu hướng riêng biệt trong từng loại. Biểu đồ đầu tiên cho thấy nhóm "Embedded" có phạm vi giá rộng hơn trong phân khúc trung bình so với "Server" - vốn có phân phối tập trung hơn với một vài giá trị ngoại lệ. Biểu đồ thứ hai tập trung vào nhóm "Server" trong phân khúc giá cao, thể hiện độ phân tán vừa phải không có giá trị ngoại lệ rõ rệt, cho thấy mức giá khá ổn định. Biểu đồ thứ ba khảo sát phân khúc giá thấp cho thấy sự biến động lớn hơn giữa các nhóm sản phẩm. Nhóm "Desktop" có khoảng tứ phân vị hẹp nhất, trong khi "Server" thể hiện độ phân tán rộng hơn cùng các giá trị ngoại lệ, phản ánh sự dao động giá lớn hơn. Các nhóm "Embedded" và "Mobile" nằm ở giữa, với "Embedded" có giá trị nhỉnh hơn đôi chút so với "Mobile". Nhìn chung, các biểu đồ trực quan hóa dữ liệu này minh họa các chiến lược định giá đa dạng giữa các phân khúc, mỗi phân khúc đều có đặc điểm phân phối và các giá trị ngoại lệ riêng biệt.

5 Thống kê suy diễn

5.1 Bài toán kiểm định một mẫu

5.1.1 Đề bài

Kiểm định Tần số Turbo tối đa (cột Max_Turbo_Frequency) với mức ý nghĩa 5%:

Xét dòng sản phẩm **Intel® Atom™ Processor C Series** (giả định được nhà sản xuất cho là giá trị trung bình tổng thể là 2.4 GHz). Dựa vào tập dữ liệu hãy kiểm định xem có giống với kết quả mà nhà sản xuất đưa ra không.

- Giả thuyết H_0: trung bình tần số Turbo tối đa là 2.4 GHz
- Giả thuyết đối H_1: trung bình tần số Turbo tối đa không phải là 2.4 GHz

5.1.2 Bài giải

- Bước 1: Lọc ra những sản phẩm Atom C Series

```
atom_data <- subset(CPU_data, grepl("Intel Atom Processor C",
    Product_Collection, fixed = TRUE))
```

- Bước 2: Chuẩn hóa cột Max_Turbo_Frequency

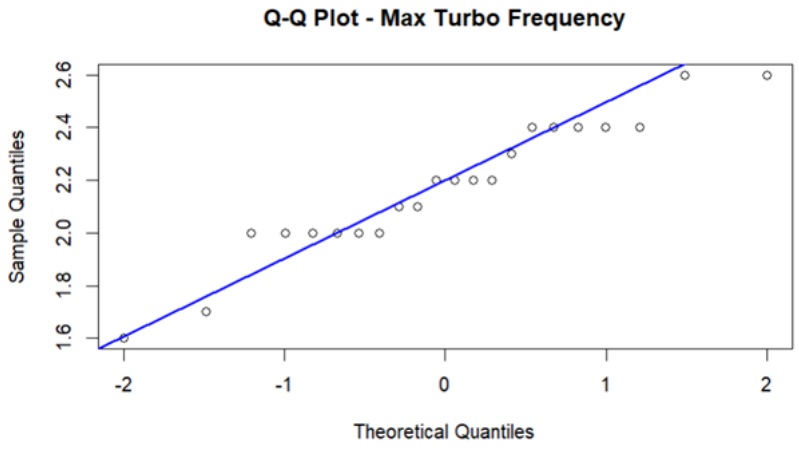
```
atom_data$Max_Turbo_Frequency <- as.numeric(gsub(" GHz", "",
    atom_data$Max_Turbo_Frequency, fixed = TRUE))
```

- Bước 3: Loại bỏ các giá trị khuyết

```
atom_data <- atom_data[!is.na(atom_data$Max_Turbo_Frequency),
    ]
```

- Bước 4: Xử lý giá trị ngoại lai

```
Q1 <- quantile(atom_data$Max_Turbo_Frequency, 0.25)
Q3 <- quantile(atom_data$Max_Turbo_Frequency, 0.75)
IQR <- Q3 - Q1
atom_data <- atom_data[atom_data$Max_Turbo_Frequency >= Q1 -
    1.5*IQR & atom_data$Max_Turbo_Frequency <= Q3 + 1.5*IQR, ]
# QQ Plot for Max_Turbo_Frequency
qqnorm(atom_data$Max_Turbo_Frequency,
        main = "Q-Q Plot - Max Turbo Frequency")
qqline(atom_data$Max_Turbo_Frequency, col = "blue", lwd = 2)
```



Hình 9: Biểu đồ QQ của Max_Turbo_Frequency)

Nhìn biểu đồ và phân tích nhóm thấy các điểm nhìn chung chênh lệch không quá so với đường chuẩn màu xanh, vì vậy có thể xem là phân phối xấp xỉ chuẩn, nhưng để chính xác hơn thì dùng Kiểm định Shapiro.

- Bước 5: Kiểm định Shapiro

```
shapiro.test(atom_data$Max_Turbo_Frequency)
```

```
> # Bước 5. Kiểm định Shapiro (phân phối chuẩn)
> shapiro.test(atom_data$Max_Turbo_Frequency)

        Shapiro-Wilk normality test

data:  atom_data$Max_Turbo_Frequency
W = 0.94273, p-value = 0.2252
```

Hình 10: Giá trị trả về sau khi kiểm định Shapiro

Giá trị p-value = 0.2252 lớn hơn mức ý nghĩa 0.05 nên ta kết luận có thể xem phân phối của biến Max_Turbo_Frequency là phân phối chuẩn. Vậy đây là bài toán 1 mẫu có phân phối chuẩn và chưa biết phương sai tổng thể, sử dụng T-test để kiểm định.

- Bước 6: T-test với giả thuyết trung bình 2.4

```
t.test(atom_data$Max_Turbo_Frequency, mu = 2.4)
```

```
> # Bước 6. T-Test với giả thuyết trung bình = 2.4
> t.test(atom_data$Max_Turbo_Frequency, mu = 2.4)

        One Sample t-test

data:  atom_data$Max_Turbo_Frequency
t = -4.153, df = 21, p-value = 0.0004507
alternative hypothesis: true mean is not equal to 2.4
95 percent confidence interval:
 2.058919 2.286535
sample estimates:
mean of x
 2.172727
```

Hình 11: Giá trị trả về sau khi kiểm định T-test

Như vậy, với giá trị p-value = 0.0004507 $<$ 0.05 ta bác bỏ giả thuyết H_0. Kết luận trung bình tần số Turbo tối đa không phải là 2.4 GHz (trung bình của mẫu là 2.172727)

5.2 Bài toán kiểm định hai mẫu

5.2.1 Đề bài

Với mức ý nghĩa 5%, có thể kết luận rằng có sự khác biệt giữa giá thành đề xuất cho khách hàng (Recommended_Customer_Price) của các CPU có kích thước các thành phần trên chip (Lithography) khác nhau không?

Chia dữ liệu Lithography thành 2 thành phần:

- Nhóm 1: Kích thước các thành phần trên chip nhỏ (có kích thước < 32 nm)

- Nhóm 2: Kích thước các thành phần trên chip lớn (có kích thước ≥ 32 nm)

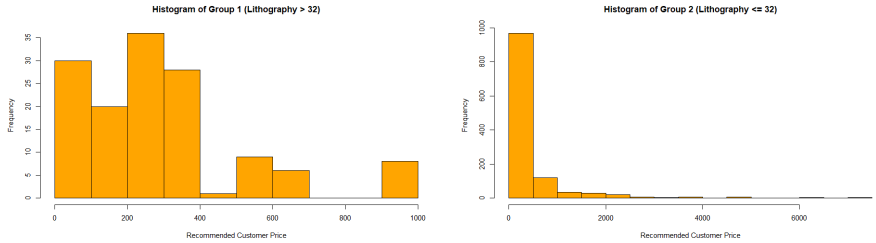
- Đặt giả thuyết H_0: Giá thành đề xuất của 2 nhóm kích thước bằng nhau, $\mu_1 = \mu_2$
- Đặt giả thuyết H_1: Giá thành đề xuất của 2 nhóm kích thước khác nhau, $\mu_1 \neq \mu_2$

5.2.2 Bài giải

- Bước 1: Tách dữ liệu Lithography thành 2 nhóm theo đề bài

```
group1 <- CPU_data$Recommended_Customer_Price[CPU_data$
    Lithography > 32]
group2 <- CPU_data$Recommended_Customer_Price[CPU_data$
    Lithography <= 32]
hist(group1, main = "Histogram of Group 1 (Lithography > 32)",
     xlab = "Recommended Customer Price", col = "orange")
print(group1)

hist(group2, main = "Histogram of Group 2 (Lithography <= 32)"
    , xlab = "Recommended Customer Price", col = "orange")
print(group2)
```



(a) Biểu đồ Histogram của nhóm 1 (b) Biểu đồ Histogram của nhóm 2

Hình 12: Dạng của 2 nhóm sau khi tách dữ liệu

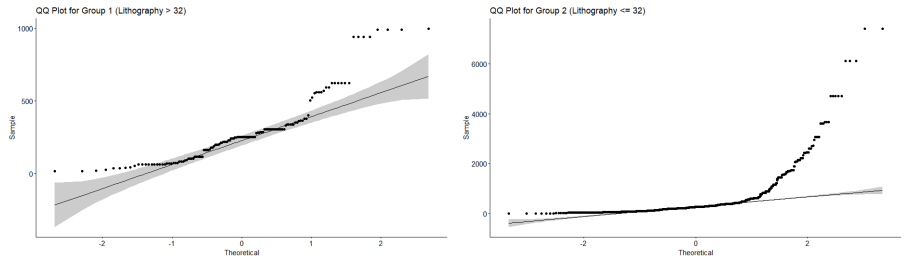
- Bước 2: Tính toán trung bình và độ lệch chuẩn của từng nhóm

```
# group1
mean_group1 <- mean(group1, na.rm = TRUE)
sd_group1 <- sd(group1, na.rm = TRUE)
n1 <- length(group1)
# group2
mean_group2 <- mean(group2, na.rm = TRUE)
sd_group2 <- sd(group2, na.rm = TRUE)
n2 <- length(group2)
```

- Bước 3: Vẽ biểu đồ QQ cho từng nhóm để dự đoán phân phối

```
# QQ-plot group1
qqplot1 <- ggqqplot(group1, main = "QQ Plot for Group 1 (
    Lithography > 32)")
print(qqplot1)

# QQ-plot group2
qqplot2 <- ggqqplot(group2, main = "QQ Plot for Group 2 (
    Lithography <= 32)")
print(qqplot2)
```



(a) Biểu đồ QQ của nhóm 1 (b) Biểu đồ QQ của nhóm 2

Hình 13: Biểu đồ QQ của các nhóm

Từ biểu đồ ta nhận thấy các điểm quan sát có nhiều điểm nằm lệch đi so với phân phối chuẩn ở cả 2 nhóm. Nên ta kết luận đây là bài toán 2 mẫu độc lập có phân phối tùy ý.

- Bước 4: Tính tiêu chuẩn kiểm định Z và kiểm định phương sai của 2 nhóm bằng F-test

```r
# z
Z <- (mean_group1 - mean_group2) / sqrt((sd_group1^2 / n1) + (sd_group2^2 / n2))
cat("Z =", Z, "\n")
# f-test
var_test_result <- var.test(group1, group2)
```

```
        F test to compare two variances

data:  group1 and group2
F = 0.1035, num df = 137, denom df = 1194, p-value < 2.2e-16
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.08155587 0.13469844
sample estimates:
ratio of variances
         0.1034959
```

Hình 14: Kết quả trả về từ kiểm định F-test

Ta thấy p-value $<$ 2.2e-16 nên bác bỏ H_0 của f-test, kết luận được phương sai của 2 nhóm kích thước là khác nhau. Thông tin này sẽ được sử dụng truyền vào bước kiểm định cuối.

- Bước 5: Kiểm định t cho 2 mẫu độc lập với phương sai khác nhau

```r
result <- t.test(group1, group2, var.equal = FALSE)
```

```
        Welch Two Sample t-test

data:  group1 and group2
t = -5.8196, df = 536.64, p-value = 1.015e-08
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -220.5581 -109.2371
sample estimates:
mean of x mean of y
 287.4275  452.3251
```

Hình 15: Kiểm định 2 mẫu độc lập

Như đã thấy từ kết quả in ra, p-value < 0.05 nên ta bác bỏ giả thuyết H_0. Như vậy, giá trị trung bình của 2 nhóm kích thước có sự khác nhau.

5.3 Bài toán anova một chiều

5.3.1 Mục tiêu

- Nắm được các khái niệm liên quan đến phân tích phương sai một yếu tố.
- Biết cách giải một bài toán phân tích phương sai một yếu tố.
- Tìm hiểu và hiện thực các lệnh của ngôn ngữ R để giải quyết bài toán.

5.3.2 Bài toán phân tích phương sai ANOVA

Để kiểm tra sự khác biệt về giá bán đề xuất trung bình (Recommended_Customer_Price) giữa các nhóm sản phẩm CPU Atom, Celeron, Pentium, Xeon, Core ta sử dụng phương pháp phân tích phương sai một nhân tố (One-way ANOVA).

Để kiểm tra xem mẫu có tuân theo phân phối chuẩn hay không, ta thực hiện kiểm định Shapiro-Wilk với các giả thuyết như sau:

- **Giả thuyết không** (H_0): Mẫu tuân theo phân phối chuẩn.
- **Giả thuyết đối** (H_1): Mẫu không tuân theo phân phối chuẩn.

Tiếp theo, để kiểm tra sự đồng nhất về phương sai giữa các nhóm, ta áp dụng kiểm định Levene với các giả thuyết sau:

- **Giả thuyết không** (H_0): Phương sai giữa các nhóm là đồng nhất.
- **Giả thuyết đối** (H_1): Có ít nhất hai nhóm có phương sai khác nhau.

Vì các mẫu được phân loại theo tên của chúng, ta giả định rằng các mẫu này là độc lập. Đối với giả thuyết chính của phân tích ANOVA, các giả thuyết được đặt ra như sau:

- **Giả thuyết không** (H_0): Trung bình giá tiền giữa các hãng là như nhau.

- **Giả thuyết đối** (H_1): Có ít nhất hai hãng có giá tiền trung bình khác nhau.

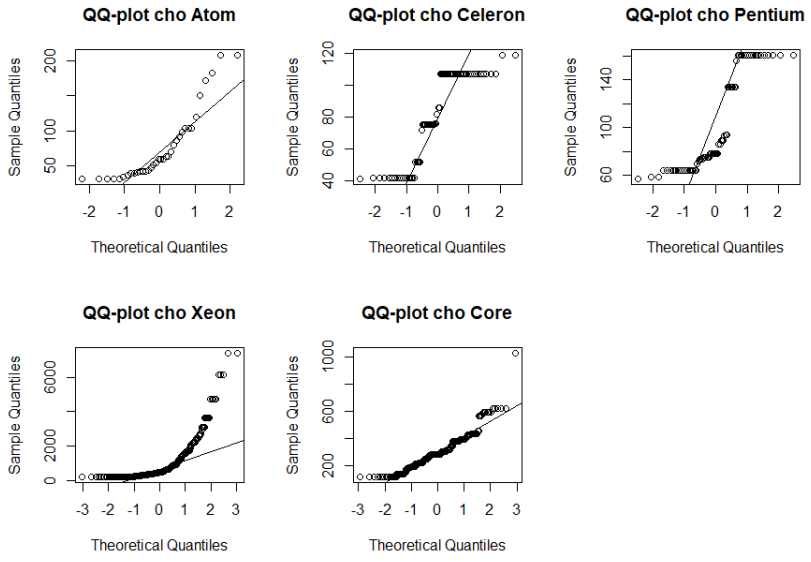
Sau khi thực hiện phân tích ANOVA, ta có thể sử dụng hàm `TukeyHSD()` để tiến hành kiểm tra so sánh cặp (post-hoc test). Các giả thuyết cho kiểm tra này như sau:

- **Giả thuyết không** (H_0): Không có sự khác biệt về giá bán trung bình giữa các nhóm sản phẩm ($\mu_1 = \mu_2 = \mu_3 = \mu_4$).
- **Giả thuyết đối** (H_1): Có ít nhất một cặp nhóm sản phẩm có sự khác biệt về giá bán trung bình.

5.3.3 Tiến hành phân tích ANOVA

a. Kiểm định phân phối chuẩn:

- **Kiểm tra bằng Q-Q Plot:** Một phương pháp trực quan để kiểm tra phân phối chuẩn là vẽ biểu đồ Q-Q (Quantile-Quantile Plot). Dưới đây là biểu đồ Q-Q Plot cho các nhóm sản phẩm:



Hình 16: Biểu đồ Q-Q Plot cho các nhóm sản phẩm: Atom, Celeron, Pentium, Xeon, Core.

- **Kiểm định Shapiro-Wilk:** Ngoài biểu đồ Q-Q Plot, chúng ta thực hiện kiểm định Shapiro-Wilk để kiểm tra tính chuẩn của dữ liệu. Dưới đây là kết quả kiểm định Shapiro-Wilk cho các nhóm:
  - Nhóm Atom: $p - value = 2.401e - 05$
  - Nhóm Celeron: $p - value = 5.172e - 09$
  - Nhóm Pentium: $p - value = 3.778e - 09$
  - Nhóm Xeon: $p - value < 2.2e - 16$
  - Nhóm Core: $p - value = 3.448e - 12$

  Vì tất cả các p-value đều nhỏ hơn 0.05, ta bác bỏ giả thuyết H_0 (dữ liệu tuân theo phân phối chuẩn). Do đó, dữ liệu giá bán không tuân theo phân phối chuẩn trong các nhóm.

b. Kiểm định tính đồng nhất phương sai

Dùng kiểm định Levene để kiểm tra giả thuyết:

- H_0: Các nhóm có phương sai bằng nhau.
- H_1: Có ít nhất một cặp nhóm có phương sai khác nhau.

Kết quả kiểm định Levene từ R:

Result

```
Levene's Test for Homogeneity of Variance (center = median)
       Df F value    Pr(>F)
group   4  35.021 < 2.2e-16 ***
      881
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Vì p-value rất nhỏ (< 0.05), ta bác bỏ H_0. Phương sai giữa các nhóm không đồng nhất.

c. Kiểm định ANOVA

Ta sử dụng hàm aov() , kết hợp với hàm summary() cùng với những tham số phù hợp để tiến hành áp dụng mô hình phân tích phương sai một yếu tố cho mẫu Recommended_Customer_Price:

Result

```
> # 3. Execute ANOVA modelWS
                     Df    Sum Sq  Mean Sq F value Pr(>F)
Product_Collection    4 105704702 26426175    50.6 <2e-16 ***
Residuals           881 460059037   522201
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

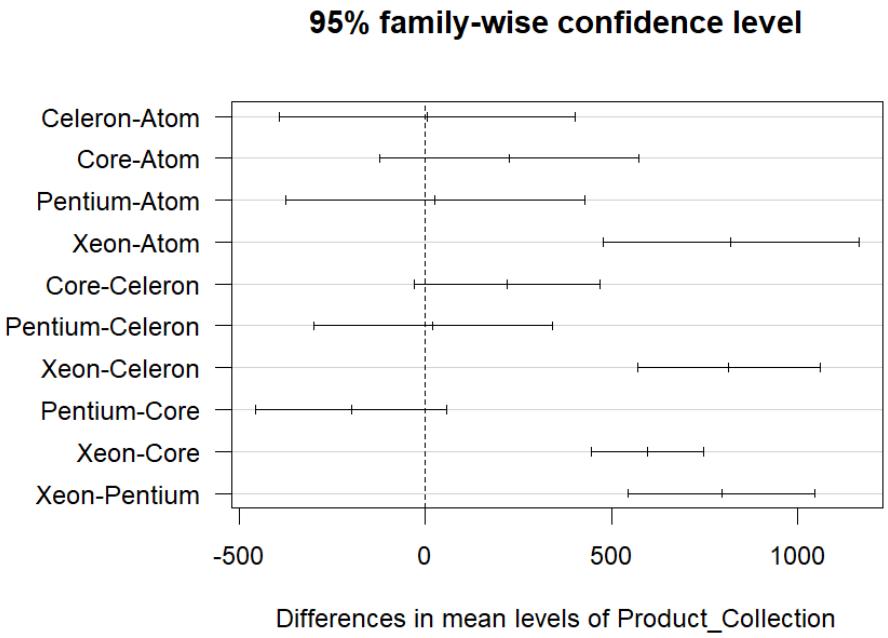
Qua kết quả ta thấy rằng giá trị F = 50.6 là cao, cho thấy trung bình của nhóm không bằng nhau. Và giá trị p-value rất nhỏ, ta bác bỏ H_0. Điều này cho thấy có sự khác biệt đáng kể về giá bán trung bình giữa ít nhất hai nhóm sản phẩm trong các nhóm sản phẩm được quan sát là có ý nghĩa thống kê.

d. Kiểm tra sự khác biệt giữa các nhóm (Post-hoc Tukey Test)

Phân tích hậu kiểm Tukey so sánh cặp giữa các nhóm ta nhận được kết quả sau:

Result

```
$Product_Collection
                         diff        lwr        upr     p adj
Celeron-Atom         5.354701 -392.64380  403.35320 0.9999996
Core-Atom          224.783242 -123.31640  572.88289 0.3947750
Pentium-Atom        25.213851 -377.06586  427.49356 0.9998070
Xeon-Atom          821.479131  477.55517 1165.40309 0.0000000
Core-Celeron       219.428541  -31.19954  470.05662 0.1180771
Pentium-Celeron     19.859150 -301.80844  341.52674 0.9998183
Xeon-Celeron       816.124431  571.32909 1060.91977 0.0000000
Pentium-Core      -199.569391 -456.94188   57.80309 0.2125930
Xeon-Core          596.695889  446.04624  747.34554 0.0000000
Xeon-Pentium       796.265281  544.56919 1047.96137 0.0000000
```



Hình 17: Biểu đồ kiểm tra sự khác nhau giữa các nhóm

Từ kết quả trên, ta nhận thấy phép so sánh nào có khoảng tin cậy 95% nằm lệch về một phía (trái hoặc phải) của giá trị 0 thì sự khác biệt đó có ý nghĩa thống kê. Tức là sự khác biệt giữa Xeon và Atom, Xeon và Celeron, Xeon và Core, Xeon và Pentium có ý nghĩa thống kê.

5.3.4 Nhận xét kết quả:

- Nhóm Xeon có giá bán trung bình cao hơn đáng kể so với các nhóm khác.
- Không có sự khác biệt đáng kể giữa các nhóm Atom,Celeron, Core, và Pentium.

5.3.5 Kết luận

Có sự khác biệt ý nghĩa thống kê về giá bán trung bình giữa các nhóm CPU (Atom, Core, Pentium, và Xeon). Nhóm Xeon có giá bán cao hơn rõ rệt so với các nhóm khác. Dữ liệu không tuân theo phân phối chuẩn và

không đồng nhất phương sai.

Trong bài toán này, nhóm đã thành công trong việc nghiên cứu các lý thuyết về phân tích phương sai ANOVA, tìm hiểu các lệnh của R để thực hiện việc tính toán, áp dụng mô hình ANOVA cho một yếu tố cụ thể, kiểm tra được giả thuyết về trung bình giữa các nhóm của yếu tố, đưa ra được kết luận cuối cùng cho bài toán.Tuy nhiên, kết quả nhóm trình bày có thể chưa thật sự chính xác. Nguyên nhân có thể đến từ nhiều yếu tố, cụ thể như sau:

- Ảnh hưởng của việc loại bỏ các quan sát khuyết trong quá trình tiền xử lý số liệu.
- Tổng thể chưa thoả giả thiết của mô hình, ví dụ như phân phối chưa thật sự chuẩn, phương sai của các nhóm không bằng nhau.
- Sự chênh lệch lớn về số lượng quan sát giữa các nhóm.

Chính vì những hạn chế trên, nhìn chung kết quả mà nhóm tìm được có thể có sai khác so với kết quả chính xác

5.4 Bài toán hồi quy tuyến tính bội

5.4.1 Tổng quan

Mục tiêu của phần này là xây dựng một mô hình hồi quy tuyến tính bội nhằm dự đoán giá CPU dựa trên các đặc trưng kỹ thuật và thông tin thị trường được thu thập trong bộ dữ liệu `Intel_CPUs.csv`.

Biến phụ thuộc: Y = *Recommended_ Customer_ Price* (giá CPU). Vì giá cả thường có phân phối lệch, ta sẽ áp dụng chuyển đổi log.
Ví dụ: $Y = \log(\text{Recommended\_Customer\_Price} + 1)$

```
CPU_data $ Recommended_Customer_Price <- log(CPU_data $
    Recommended_Customer_Price + 1)
```

→ Nhằm ổn định phương sai và cải thiện tính tuyến tính của quan hệ.

Biến độc lập: Các đặc trưng kỹ thuật và thị trường được lựa chọn bao gồm:

Product_Collection	**Vertical_Segment**	**Year**
Status	**Lithography**	**nb_of_Cores**
nb_of_Threads	**Processor_Base_Frequency**	**Cache**
Cache_Type	**Instruction_Set**	**TDP**
Max_Memory_Size	**Max_nb_of_Memory_Channels**	**Max_Memory_Bandwidth**

Các biến độc lập được sử dụng

Trong mô hình, các cột định tính như `Product_Collection`, `Vertical_Segment`, `Status` và `Cache_Type` sẽ được chuyển sang kiểu định lượng (factor).

5.4.2 Công thức mô hình và giả thuyết

Sau khi áp dụng chuyển đổi log cho giá, mô hình hồi quy tuyến tính bội được đặt theo công thức:

$$Y = \beta_0 + \beta_1 \,\text{Product\_Collection} + \beta_2 \,\text{Vertical\_Segment} + \beta_3 \,\text{Year} + \beta_4 \,\text{Lithography} + \beta_5 \,\text{Status} + \beta_6 \,\text{nb\_of\_Cores} + \beta_7 \,\text{nb\_of\_Threads} + \beta_8 \,\text{Processor\_Base\_Frequency} + \beta_9 \,\text{Cache} + \beta_{10} \,\text{Cache\_Type} + \beta_{11} \,\text{Instruction\_Set} + \beta_{12} \,\text{TDP} + \beta_{13} \,\text{Max\_Memory\_Size} + \beta_{14} \,\text{Max\_nb\_of\_Memory\_Channels} + \beta_{15} \,\text{Max\_Memory\_Bandwidth} + \varepsilon$$

Trong đó:

- Y là giá CPU sau chuyển đổi log.
- β_0 là hằng số (intercept) và β_i $(i = 1, 2, \ldots, 15)$ là các hệ số hồi quy.
- ε là sai số ngẫu nhiên, độc lập và có phân phối chuẩn $N(0; \sigma^2)$.

Giả thuyết kiểm định:

- **H_0:** Tất cả các hệ số $\beta_i = 0$ (không có mối liên hệ tuyến tính giữa các biến độc lập với giá CPU).
- **H_1:** Ít nhất một hệ số $\beta_i \neq 0$, cho thấy có mối liên hệ giữa các biến độc lập với giá CPU.

5.4.3 Triển khai mô hình và đánh giá

Quá trình triển khai mô hình bao gồm các bước sau:

1. Lọc các cột từ CPU_Data để phân tích hồi quy

   → Tất cả các cột hiện tại đều được chọn, trừ Launch Date.

```
CPU_regression <- CPU_data %>%
  select(
  Recommended_Customer_Price, Product_Collection, Vertical_
     Segment, Year, Lithography, Status,
  nb_of_Cores, nb_of_Threads, Processor_Base_Frequency, Cache,
      Cache_Type, Instruction_Set, TDP,
  Max_Memory_Size, Max_nb_of_Memory_Channels, Max_Memory_
     Bandwidth
)
```

2. Chuyển đổi các cột định tính thành định lượng bằng factor. Cụ thể là các cột:
   - Product_Collection
   - Vertical_Segment
   - Status
   - Cache_Type
3. Chia dữ liệu ban đầu thành 80% để huấn luyện (train), 20% để kiểm tra (test).

```
train_indices <- createDataPartition(CPU_regression $
   Recommended_Customer_Price, p = 0.8, list = FALSE)
#------------ split data into train and test ---------------
train_data <- CPU_regression[train_indices, ]
test_data <- CPU_regression[-train_indices, ]
```

4. Huấn luyện mô hình để tìm các hệ số của đường hồi quy

   - Xây dựng mô hình hồi quy tuyến tính dựa trên tập dữ liệu của train_data.

Result

```
Call:
lm(formula = Recommended_Customer_Price ~ ., data = train_data)

Residuals:
    Min      1Q  Median      3Q     Max
-2.3192 -0.2217  0.0170  0.2207  1.5332

Coefficients:
                                 Estimate Std. Error t value Pr(>|t|)
(Intercept)                    -8.463e+01  3.698e+01  -2.289 0.022293 *
Product_CollectionCeleron      -1.223e-01  1.179e-01  -1.038 0.299725
Product_CollectionCore          6.043e-01  1.227e-01   4.923 9.90e-07 ***
Product_CollectionLegacy Atom   2.397e-02  1.935e-01   0.124 0.901428
Product_CollectionLegacy Celeron -2.931e-01  1.356e-01  -2.161 0.030927 *
Product_CollectionLegacy Core   4.431e-01  1.232e-01   3.596 0.000338 ***
Product_CollectionLegacy Pentium -2.293e-01  1.379e-01  -1.663 0.096619 .
Product_CollectionLegacy Xeon   5.756e-01  1.439e-01   3.999 6.82e-05 ***
Product_CollectionPentium       1.212e-02  1.234e-01   0.098 0.921835
Product_CollectionQuark         9.921e-01  3.706e-01   2.677 0.007538 **
Product_CollectionXeon          5.853e-01  1.324e-01   4.421 1.09e-05 ***
Vertical_SegmentEmbedded        4.267e-01  7.378e-02   5.784 9.66e-09 ***
Vertical_SegmentMobile          6.243e-01  5.259e-02  11.872  < 2e-16 ***
Vertical_SegmentServer          3.941e-01  9.006e-02   4.376 1.33e-05 ***
Year                            4.115e-02  1.832e-02   2.247 0.024871 *
Lithography                     2.075e-03  4.083e-03   0.508 0.611419
StatusEnd of Interactive Support 9.218e-01  2.039e-01   4.520 6.89e-06 ***
StatusEnd of Life               9.280e-01  1.989e-01   4.665 3.49e-06 ***
StatusLaunched                  9.455e-01  1.935e-01   4.885 1.19e-06 ***
nb_of_Cores                    -1.284e-02  4.385e-03  -2.928 0.003484 **
nb_of_Threads                   1.667e-02  3.446e-03   4.839 1.51e-06 ***
Processor_Base_Frequency        1.676e-04  3.750e-05   4.469 8.70e-06 ***
Cache                           5.415e-01  5.458e-02   9.921  < 2e-16 ***
Cache_TypeL3                   -1.535e-01  1.003e-01  -1.530 0.126415
Cache_TypeLast Level Cache     -1.108e+00  1.598e-01  -6.933 7.22e-12 ***
Cache_TypeNormal               -2.596e-01  7.448e-02  -3.486 0.000511 ***
Cache_TypeSmartCache           -5.598e-02  6.899e-02  -0.811 0.417273
Instruction_Set                 6.613e-03  6.668e-03   0.992 0.321604
TDP                            -5.333e-04  1.037e-03  -0.514 0.607149
Max_Memory_Size                -7.212e-02  2.326e-02  -3.101 0.001980 **
Max_nb_of_Memory_Channels       4.082e-01  3.515e-02  11.614  < 2e-16 ***
Max_Memory_Bandwidth           -1.869e-02  1.705e-03 -10.963  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4423 on 1036 degrees of freedom
Multiple R-squared:  0.8029,    Adjusted R-squared:  0.797
F-statistic: 136.1 on 31 and 1036 DF,  p-value: < 2.2e-16
```

   - Dựa vào tập dữ liệu này, ta có thể nhận xét được sự ảnh hưởng của các

biến độc lập được chọn đối với giá cả của CPU bằng cách đánh giá giá trị của p-value (`Pr(>|t|)`) của từng biến.

- Cụ thể, nếu p-value có giá trị càng gần 0 thì mức độ ảnh hưởng của biến độc lập tới giá cả của CPU càng lớn, các biến có giá trị $p - value < 0.05$ thì có thể xem như đủ dữ kiện để kết luận các biến này có ảnh hưởng tới giá CPU.
- Ngoài ra, tập dữ liệu trên còn có cột biểu diễn hệ số $(\beta_0, \beta_1, \dots)$ tương ứng của các biến độc lập (`Estimate`) với sai số chuẩn là `Std. Error`.

$$Y = -84.63 - 0.1223.X_1 + 0.6043.X_2 + ... + 0.4082.X_{30} - 0.01869.X_{31}$$

- Tính toán giá trị của bình phương sai số trung bình (MSE)

$$MSE = \frac{SSE}{n - num_{var}} = \frac{\sum_{i=1}^{n} (y_i - \widehat{y_i})^2}{n - num_{var}}$$

  - Trong đó, num_{var} là số biến

  → Nếu giá trị này càng nhỏ thì mô hình càng chính xác. MSE thay đổi đáng kể khi sự chênh lệch đáng kể giữa giá trị thực tế so với giá trị hồi quy dự báo tương ứng càng lớn. Cụ thể MSE của mô hình này có giá trị là 0.1898; giá trị này nằm trong miền có thể chấp nhận được.

5. Lược bỏ bớt một số biến ít hoặc không gây ảnh hưởng đến Y.
   - Ta sẽ lọc lại dữ liệu bằng cách lược bỏ đi các biến độc lập có $p - value > 0.05$, cụ thể là `Lithography, Instruction_Set, TDP`.
   - Không loại bỏ các phân loại của `Product_Collection, Cache`.

Result

```
Call:
lm(formula = Recommended_Customer_Price ~ ., data = train_data)

Residuals:
     Min       1Q   Median       3Q      Max
-2.32029 -0.22602  0.01858  0.21974  1.53979

Coefficients:
```

```
                                  Estimate Std. Error t value Pr(>|t|)
(Intercept)                     -7.517e+01  2.561e+01  -2.935 0.003407 **
Product_CollectionCeleron       -1.341e-01  1.169e-01  -1.148 0.251294
Product_CollectionCore           6.005e-01  1.224e-01   4.907 1.07e-06 ***
Product_CollectionLegacy Atom   -8.378e-02  1.524e-01  -0.550 0.582729
Product_CollectionLegacy Celeron -2.983e-01  1.332e-01  -2.240 0.025324 *
Product_CollectionLegacy Core    4.384e-01  1.229e-01   3.569 0.000375 ***
Product_CollectionLegacy Pentium -2.299e-01  1.363e-01  -1.686 0.092116 .
Product_CollectionLegacy Xeon    5.871e-01  1.424e-01   4.122 4.06e-05 ***
Product_CollectionPentium        6.361e-03  1.231e-01   0.052 0.958808
Product_CollectionQuark          7.496e-01  2.896e-01   2.589 0.009764 **
Product_CollectionXeon           5.882e-01  1.319e-01   4.461 9.04e-06 ***
Vertical_SegmentEmbedded         4.242e-01  7.315e-02   5.799 8.84e-09 ***
Vertical_SegmentMobile           6.285e-01  5.186e-02  12.120  < 2e-16 ***
Vertical_SegmentServer           3.856e-01  8.897e-02   4.334 1.61e-05 ***
Year                             3.673e-02  1.268e-02   2.896 0.003860 **
StatusEnd of Interactive Support 9.321e-01  2.026e-01   4.601 4.72e-06 ***
StatusEnd of Life                9.317e-01  1.986e-01   4.692 3.07e-06 ***
StatusLaunched                   9.463e-01  1.933e-01   4.896 1.13e-06 ***
nb_of_Cores                     -1.424e-02  3.683e-03  -3.866 0.000118 ***
nb_of_Threads                    1.666e-02  3.441e-03   4.841 1.49e-06 ***
Processor_Base_Frequency         1.557e-04  2.846e-05   5.472 5.58e-08 ***
Cache                            5.339e-01  5.188e-02  10.290  < 2e-16 ***
Cache_TypeL3                    -1.615e-01  9.981e-02  -1.618 0.106029
Cache_TypeLast Level Cache      -1.120e+00  1.591e-01  -7.038 3.53e-12 ***
Cache_TypeNormal                -2.712e-01  7.367e-02  -3.681 0.000244 ***
Cache_TypeSmartCache            -6.890e-02  6.737e-02  -1.023 0.306705
Max_Memory_Size                 -7.188e-02  2.305e-02  -3.119 0.001866 **
Max_nb_of_Memory_Channels        4.040e-01  3.447e-02  11.720  < 2e-16 ***
Max_Memory_Bandwidth            -1.873e-02  1.700e-03 -11.019  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.442 on 1039 degrees of freedom
Multiple R-squared:  0.8026,    Adjusted R-squared:  0.7973
F-statistic: 150.9 on 28 and 1039 DF,  p-value: < 2.2e-16
```

- Có thể nhận thấy, sau khi loại bỏ các biến có $p - value > 0.05$ thì $F - statistic$ có cải thiện hơn được một chút. Chứng tỏ rằng việc loại bỏ các biến trên là cần thiết.

- Giá trị F và p ($F - statistic/p - value$): sử dụng để kiểm tra tính đáng tin cậy của mô hình hồi quy tuyến tính. $F - statistic$ đo lường sự khác biệt giữa mô hình tuyến tính và mô hình không có biến độc lập. Giá trị $F - statistic$ càng cao và p-value càng nhỏ, càng có mối quan hệ tuyến tính giữa biến phụ thuộc và biến độc lập. Mô hình có $F - statistic$ khá cao (150.9) và $p - value$ rất nhỏ ($< 2.2e - 16$) điều này cho thấy mô hình có mối quan hệ tuyến tính giữa biến phụ thuộc và các biến độc lập.

5.4.4 Đánh giá mô hình

Ta đánh giá độ chính xác của mô hình hồi quy bằng cách tính Mean Squared Error (MSE), trung bình của bình phương các sai số giữa giá trị dự đoán và giá trị thực tế và Adjusted_R - hệ số xác định thích nghi.

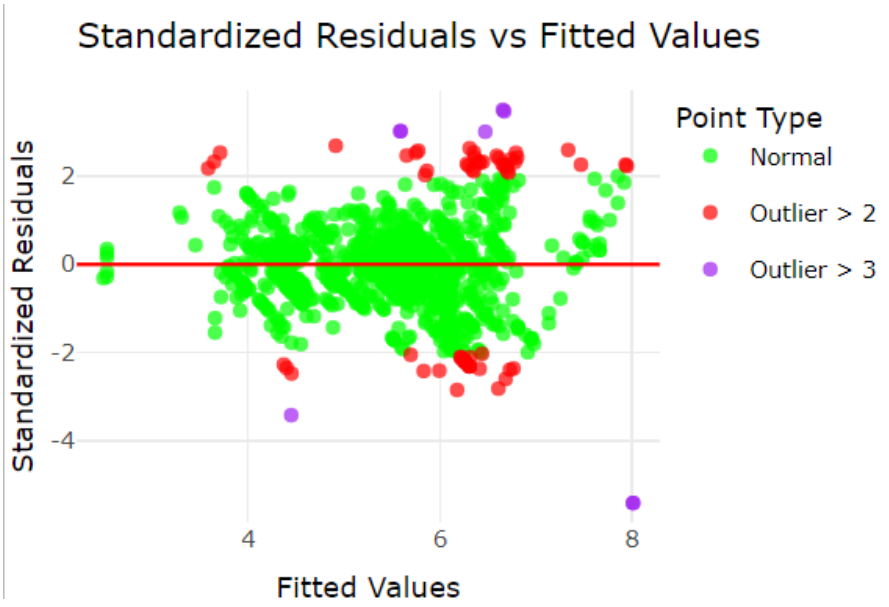
```
Console
train_MSE: 0.1953641
train_R-squared: 0.8026
train_Adjusted R-squared: 0.7973
test_MSE: 0.2301497
test_R-squared: 0.7819091
test_Adjusted R-squared: 0.5928761
```

- Dựa vào kết quả trên, ta có thể thấy giá trị R_Squared của tập huấn luyện và tập test gần như bằng nhau, còn giá trị MSE của tập train và test có sự khác nhau nhẹ. Từ đó, ta có thể rút ra nhận xét là mô hình dự đoán tốt với dữ liệu chưa được huấn luyện.
- Sở dĩ Adjusted R_Squared của tập huấn luyện và tập kiểm thử có sự chênh lệch lớn như vậy là do số lượng các mẫu của từng tiêu chí độc lập của hai tập huấn luyện và kiểm thử có sự chêch lệch lớn. Cụ thể là phân bố theo tỉ lệ 80 - 20. Mà Adjusted R_Squared phụ thuộc vào số hàng cũng như số cột nên mới dẫn đến sự khác biệt giá trị này.

5.4.5 Kết quả và Kiểm định các Giả định

Vẽ đồ thị biểu diễn

1. Biểu đồ Scatter



Hình 18: Biểu đồ scatter biểu diễn sự chênh lệch

Biểu đồ này biểu diễn sai số chuẩn giữa giá CPU do mô hình dự đoán với giá CPU thực tế tương ứng.

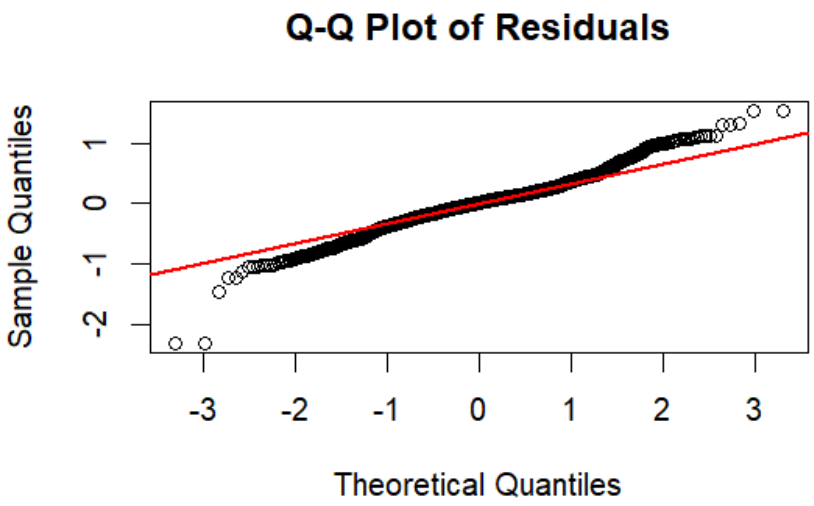
- Các điểm màu xanh lá: Biểu diễn các điểm có sai số chuẩn trong mức bình thường (≤ 2).
- Các điểm màu đỏ: là các điểm ngoại lai có sai số chuẩn > 2.
- Các điểm màu tím: là các điểm ngoại lai có sai số chuẩn > 3.

2. Biểu đồ Q-Q

Giả sử các phần dư chuẩn hóa sắp xếp theo thứ tự tăng dần là $r_1, r_2, \ldots, r_n$

Tính $q_i = \Phi^{-1}\left(\frac{i-0.5}{n}\right)$, trong đó Φ^{-1} là hàm phân vị ngược của phân phối chuẩn tắc, ví dụ $\Phi^{-1}(0.6)$ trả về giá trị x sao cho tích phân từ âm vô cực đến x của phân phối chuẩn tắc là 0.6.

Lập các cặp toạ độ $(q_1, r_1); (q_2, r_2); \ldots; (q_n, r_n)$ rồi vẽ đồ thị (trục hoành q, trục tung r).



Hình 19: Biểu đồ Q-Q kiểm định sự chính xác của mô hình

Đường màu đỏ là đường hồi quy của các cái điểm này, các điểm này càng gần đường màu đỏ thì phân phối của phần dư càng giống phân phối chuẩn.

Kết luận: Như vậy, quá trình phân tích suy diễn trong bài toán này không chỉ giúp hiểu sâu mối quan hệ giữa các đặc trưng kỹ thuật với giá CPU mà còn cung cấp cơ sở để dự đoán giá sản phẩm trong tương lai dựa trên các thông số kỹ thuật cho trước.

6 Thảo luận và mở rộng

Qua quá trình nghiên cứu đề tài nhóm có rút ra được một số vấn đề như sau:

- Ở phần kiểm định trung bình 1 mẫu và 2 mẫu, việc không biết được độ lệch chuẩn của tổng thể khiến cho việc lựa chọn phương pháp kiểm định có thể xảy ra sai sót. Cũng như dù loại bỏ được nhiều yếu tố không

mong muốn thì các sai lầm kiểm định (loại I và II) vẫn khả năng cao đã tồn tại kết quả.

- Mô hình hồi quy tuyến tính bội có nhiều điểm khả quan, tuy nhiên vấn đề phương sai không hằng là đáng lưu tâm, để giải quyết thì nhóm nghiên cứu có đề xuất sử dụng mô hình hồi quy Robust (một mô hình hồi quy khi gặp hiện tượng phương sai của sai số có bị thay đổi nhiều).

7 Nguồn dữ liệu và nguồn code

Tại đây!

8 Tài liệu tham khảo

[1] Douglas C. Montgomery, Applied Statistics and Probability for Engineers (Third Edition)
[2] Introductory Statistics with R, J Jambers - D.Hand - W.Hardle.
[3] Peter Dalgaard. 2008. Introductory Statistics with R. Springer.
[4] Kutner, M. H., Nachtsheim, C. J., Neter, J., & Li, W. (2005). Applied Linear Statistical
Models (5th ed.). McGraw-Hill Education.
[5] Moore, D. S., McCabe, G. P., & Craig, B. A. (2017). Introduction to the Practice of Statistics (9th ed.). W.H. Freeman and Company.
[6] Sách: Phân tích dữ liệu với R (Nguyễn Văn Tuấn 2022)