

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



## Xác suất thống kê MT2013

---

Báo cáo bài tập lớn

# Phân tích dữ liệu GPU

---

Advisor(s): Advisor h

Student(s): Student 1 ID 1

Student 2 ID 2

Student 3 ID 3

HO CHI MINH CITY, NOVEMBER 2025



# **Mục lục**

## **Danh sách hình vẽ**

## **Danh sách bảng**

## **Danh sách đoạn mã**

1	Đọc dữ liệu từ file trong R . . . . .	12
2	Thay thế giá trị rác thành NA . . . . .	13
3	Loại bỏ các cột có tỷ lệ NA > 10% . . . . .	13
4	Loại bỏ các cột có tỷ lệ NA > 10% . . . . .	14
5	Chuẩn hoá các đơn vị đo trong dữ liệu . . . . .	15
6	Chuyển biến phân loại trong R . . . . .	17
7	Kiểm tra cấu trúc dữ liệu . . . . .	17
8	Kiểm tra thông số cơ bản của các biến . . . . .	17
9	Xây dựng mô hình ANOVA hai yếu tố trong R . . . . .	18
10	Kiểm định chuẩn hoá phần dư bằng Shapiro-Wilk test . . . . .	18
11	Kiểm định phương sai đồng nhất bằng Levene's test . . . . .	19



## 1 Tổng quan dữ liệu

Trong thời đại công nghệ phát triển nhanh chóng, bộ xử lý đồ họa (GPU – Graphics Processing Unit) đã trở thành một trong những thành phần quan trọng nhất của máy tính hiện đại. Ban đầu, GPU được thiết kế với mục đích chính là xử lý hình ảnh và đồ họa trong các trò chơi điện tử, phần mềm thiết kế và dựng phim. Tuy nhiên, cùng với sự tiến bộ của công nghệ, vai trò của GPU đã vượt xa khỏi phạm vi đồ họa thuần túy. Ngày nay, GPU đóng vai trò cốt lõi trong các lĩnh vực như trí tuệ nhân tạo (AI), học sâu (Deep Learning), mô phỏng khoa học, và xử lý dữ liệu lớn. Nhờ vào cấu trúc song song mạnh mẽ với hàng nghìn lõi xử lý, GPU có thể thực hiện hàng loạt phép tính phức tạp cùng lúc, giúp rút ngắn đáng kể thời gian xử lý so với CPU truyền thống. Dưới đây là tập dữ liệu khảo sát các yếu tố về GPU cung cấp những thông tin chi tiết như tốc độ xung nhịp, nhiệt độ tối đa, mức tiêu thụ điện năng, kích thước khuôn chip, ngày phát hành, giá bán, và nhiều đặc trưng kỹ thuật khác. Việc phân tích các dữ liệu này giúp ta hiểu rõ hơn về sự phát triển của công nghệ GPU qua các giai đoạn, từ đó đánh giá xu hướng tiến hóa về hiệu năng, giá thành, và mức độ tối ưu năng lượng. Bên cạnh đó, người nghiên cứu có thể khám phá mối quan hệ giữa giá thành và hiệu suất hoạt động, tìm hiểu xem liệu có nhà sản xuất nào nổi bật trong một phân khúc nhất định hay không. Thông qua việc khai thác và phân tích dữ liệu GPU, chúng ta có thể dự đoán xu hướng của các thế hệ GPU tương lai, phục vụ cho các ứng dụng như học máy, đồ họa máy tính, và tính toán hiệu năng cao.

Tập dữ liệu All\_GPUs, bao gồm 34 thông số của 3406 bộ xử lý đồ họa (GPU) khác nhau đến từ 3 nhà sản xuất chính là NVIDIA, AMD và Intel. Dữ liệu được quan sát và thu thập từ trang web [Kaggle](#). Tập dữ liệu này có tương đối nhiều thông số, trong đó có thể kể đến một số thông số được nêu ra trong bảng ?? dưới đây:



STT	Tên biến	Đơn vị	Mô tả
1	Manufacturer		Hãng của sản xuất GPU
2	Release_Date		Năm sản xuất của GPU
3	Memory_Bandwidth	Gigabyte/giây (GB/s)	Lượng dữ liệu tối đa mà bộ nhớ GPU có thể truyền tải trong mỗi giây
4	Memory_Speed	MHz	Dộ rộng của bus bộ nhớ, ảnh hưởng đến tốc độ truy cập và hiệu suất của bộ nhớ GPU
5	L2_Cache	KB	Bộ nhớ đệm cấp 2 giúp GPU truy cập nhanh hơn vào dữ liệu được sử dụng thường xuyên, tối ưu hóa hiệu suất
6	Memory_Bus	Bit	Dộ rộng của kênh truyền dữ liệu trong RAM (Memory).
7	Memory		Dung lượng bộ nhớ đồ họa (VRAM) của GPU, quyết định khả năng xử lý và lưu trữ dữ liệu hình

Table 1.1: Bảng mô tả một vài thông số quan trọng của tập dữ liệu GPU

Như đã đề cập ở trên, tập dữ liệu có tổng cộng 34 thông số khác nhau, tuy nhiên nếu liệt kê hết ở bảng ?? thì sẽ rất dài. Ngoài ra, trong các thông số đó, có nhiều thông số mang tính kỹ thuật cao và không phổ biến, những dữ liệu này sẽ được xử lý sau để dễ dàng hơn trong việc phân tích và trực quan hóa dữ liệu.



## 2 Kiến thức nền

### 2.1 Thống kê mô tả và thống kê suy diễn

**Thống kê mô tả (descriptive statistics):** là quá trình thu thập, biểu diễn, tổng hợp và xử lý dữ liệu để biến đổi dữ liệu thành thông tin.

**Thống kê suy diễn (Inferential statistics):** xử lý các thông tin có được từ thống kê mô tả, từ đó đưa ra các cơ sở cho những dự đoán (predictions), dự báo (forecasts) và các ước lượng (estimations).

### 2.2 Các đặc trưng của tổng thể và mẫu

#### 2.2.1 Khái niệm

**Tổng thể thống kê (population):** là tập hợp các phần tử thuộc đối tượng nghiên cứu, cần được quan sát, thu thập và phân tích theo một hoặc một số đặc trưng nào đó. Các phần tử tạo thành tổng thể thống kê được gọi là đơn vị tổng thể.

**Mẫu (sample):** là một số đơn vị được chọn ra từ tổng thể theo một phương pháp lấy mẫu nào đó. Các đặc trưng mẫu được sử dụng để suy rộng ra các đặc trưng của tổng thể nói chung.

**Đặc điểm thống kê (dấu hiệu nghiên cứu):** là các tính chất quan trọng liên quan trực tiếp đến nội dung nghiên cứu và khảo sát cần thu thập dữ liệu trên các đơn vị tổng thể. Người ta chia làm 2 loại: đặc điểm thuộc tính và đặc điểm số lượng.

#### 2.2.2 Tỷ lệ

Với một tổng thể có  $N$  phần tử và  $M$  phần tử mang tính chất A nào đó. Tỷ lệ tổng thể (kí hiệu:  $p$ ) được tính bởi công thức:

$$p = \frac{M}{N}$$

Với một mẫu có  $n$  phần tử và có  $m$  phần tử mang tính chất A nào đó. Tỷ lệ mẫu (kí hiệu:  $f$  hay  $\bar{p}$ ) được tính bởi công thức:

$$p = f = \frac{m}{n}$$

#### 2.2.3 Trung bình

**Trung bình (mean):** là đại lượng thường được sử dụng nhất để đo giá trị trung tâm của dữ liệu (Trung bình bị ảnh hưởng bởi các giá trị ngoại lai). Với một tổng thể có  $N$  phần tử,



trung bình tổng thể (kí hiệu:  $\mu$  hay  $\bar{X}$ ) tính bởi công thức:

$$\mu = \frac{1}{N} \sum_{i=1}^N X_i = \frac{X_1 + X_2 + \cdots + X_N}{N}$$

Với một mẫu có  $n$  phần tử, trung bình mẫu (kí hiệu:  $\bar{x}$ ) tính bởi công thức:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i = \frac{x_1 + x_2 + \cdots + x_n}{n}$$

Trong trường hợp X có bảng phân phối tần số như sau:

X	$x_1$	$x_2$	$x_3$	...	$x_k$
Tần số	$n_1$	$n_2$	$n_3$	...	$n_k$

Ta lại có trung bình mẫu tính bởi công thức:

$$\bar{x} = \frac{\sum_{i=1}^k x_i n_i}{n} = \frac{x_1 n_1 + x_2 n_2 + \cdots + x_k n_k}{n}$$

#### 2.2.4 Phương sai, độ lệch chuẩn

**Phương sai (Variance):** là trung bình của bình phương độ lệch các giá trị so với trung bình. Phương sai phản ánh độ phân tán hay sự biến thiên của dữ liệu.

**Độ lệch chuẩn (Standard deviation):** Độ lệch chuẩn (Standard deviation) là căn bậc hai của phương sai. Độ lệch chuẩn dùng để đo sự biến thiên, biểu diễn sự biến thiên xung quanh trung bình và cũng có cùng đơn vị đo với dữ liệu gốc.

Với một tổng thể có  $N$  phần tử, phương sai tổng thể (kí hiệu:  $\sigma^2$ ) tính bởi công thức:

$$\sigma^2 = \frac{\sum_{i=1}^N (x_i - \mu)^2}{N} = \frac{\sum_{i=1}^N x_i^2 - N\mu^2}{N} = \frac{\sum_{i=1}^N x_i^2}{N} - \mu^2$$

Khi đó:  $\sigma$  được gọi là độ lệch chuẩn của tổng thể.

Với một mẫu có  $n$  phần tử, phương sai mẫu (kí hiệu:  $s^2$ ) tính bởi công thức:

$$s^2 = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n-1} = \frac{\sum_{i=1}^n x_i^2 - n\bar{x}^2}{n-1} = \frac{\sum_{i=1}^n x_i^2}{n-1} - \frac{n\bar{x}^2}{n-1}$$

Trong trường hợp X có bảng phân phối tần số như sau:

X	$x_1$	$x_2$	$x_3$	...	$x_k$
Tần số	$n_1$	$n_2$	$n_3$	...	$n_k$



Ta lại có phương sai mẫu tính bởi công thức:

$$s^2 = \frac{1}{n-1} \sum_{i=1}^k n_i(x_i - \bar{x})^2 = \frac{1}{n-1} [n_1(x_1 - \bar{x})^2 + n_2(x_2 - \bar{x})^2 + \cdots + n_k(x_k - \bar{x})^2]$$

Khi đó:  $s$  được gọi là độ lệch mẫu.

### 2.2.5 Các đặc trưng khác

**Yếu vị (Mode):** là giá trị của phần tử có số lần xuất hiện lớn nhất trong mẫu. Yếu vị không bị ảnh hưởng bởi các điểm ngoại lai.

**Hệ số biến thiên (Coefficient of variation):** đo lường mức độ biến động tương đối của mẫu dữ liệu, được dùng khi người ta muốn so sánh mức độ biến động của các mẫu không cùng đơn vị đo. Đơn vị tính bằng %.

$$CV(\text{tongthe}) = \frac{\sigma}{\mu} \times 100\%$$

$$CV(\text{mau}) = \frac{s}{\bar{x}} \times 100\%$$

**Sai số chuẩn (Standard Error):** là giá trị đại diện cho độ lệch chuẩn của giá trị trung bình trong tập dữ liệu. Nó phục vụ như một thước đo biến động cho các biến ngẫu nhiên hay độ lượn độ phân tán. Độ phân tán càng nhỏ, dữ liệu càng chính xác.

$$SE(\text{tongthe}) = \frac{\sigma}{\sqrt{N}}$$

$$SE(\text{mau}) = \frac{s}{\sqrt{n}}$$

**Trung vị (Median):** Giả sử  $X$  có  $N$  quan sát, sắp các quan sát này theo thứ tự tăng dần. Trung vị là giá trị nằm chính giữa dãy số này và chia nó thành 2 phần bằng nhau. Cụ thể:

Giả sử mẫu có kích thước  $n$  được sắp xếp tăng dần theo giá trị được khảo sát:

$$x_1 \leq x_2 \leq \cdots \leq x_{n-1} \leq x_n$$

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

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

Trung vị không bị ảnh hưởng bởi các điểm ngoại lai (outliers).

**Tứ phân vị (Quartiles):** Giá trị trung vị chia mẫu dữ liệu đã sắp thứ tự thành 2 tập có số phần tử bằng nhau. Trung vị của tập dữ liệu nhỏ hơn là  $Q_1$  (gọi là tứ phân vị dưới) và trung vị của tập dữ liệu lớn hơn là  $Q_3$  (gọi là tứ phân vị trên).  $Q_2$  được lấy bằng giá trị trung vị. Độ tráoi giữa, hay là khoảng tứ phân vị  $IQR = Q_3 - Q_1$ .

**Điểm Outlier:** còn gọi là điểm dị biệt, điểm ngoại lệ, điểm ngoại lai.... Đó là các phần tử



của mẫu có giá trị nằm ngoài khoảng

$$(Q_1 - 1.5 \times IQR; Q_3 + 1.5 \times IQR)$$

## 2.3 Ước lượng

Lý thuyết ước lượng là một nội dung trọng tâm trong thống kê và nghiên cứu khoa học, tập trung vào việc xác định giá trị của các tham số (parameters) của quần thể dựa trên những mẫu (samples) được chọn ra từ quần thể đó. Mục tiêu chính của ước lượng là tìm ra các giá trị gần đúng cho những đại lượng đặc trưng của quần thể như trung bình tổng thể ( $\mu$ ), phương sai tổng thể ( $\sigma^2$ ), và tỷ lệ phần tử có đặc điểm nhất định trong quần thể ( $p$ ).

- **Khoảng Tin Cậy (Confidence Interval - CI):** Là một loại ước lượng khoảng được sử dụng để chỉ ra phạm vi mà ta tin rằng tham số của tổng thể nằm trong đó. Khoảng tin cậy thường được xác định bởi hai giới hạn: giới hạn dưới và giới hạn trên. Ví dụ, một khoảng tin cậy 95% cho trung bình tổng thể có thể là (20, 30), nghĩa là ta tin rằng với độ tin cậy 95%, trung bình thực sự của tổng thể nằm trong khoảng từ 20 đến 30.
- **Mức Ý Nghĩa (Significance Level -  $\alpha$ ):** Là ngưỡng mà ta chọn để quyết định ý nghĩa. Ví dụ mức ý nghĩa thường được chọn ở mức 0.05 - nghĩa là khả năng kết quả quan sát sự khác biệt được nhìn thấy trên số liệu là ngẫu nhiên chỉ là 5%.
- **Độ Tin Cậy (Confidence Level):** Được biểu thị dưới dạng một tỷ lệ phần trăm chỉ mức độ tin tưởng hoặc sự chắc chắn mà khoảng tin cậy ước lượng của chúng ta bao gồm tham số tổng thể thực sự. Ví dụ: nếu ta xây dựng khoảng tin cậy với mức tin cậy 95%, ta tin chắc rằng 95 trên 100 lần ước tính sẽ nằm giữa giá trị trên và giá trị dưới được chỉ định bởi khoảng tin cậy.

$$\gamma = 1 - \alpha$$

Có hai phương pháp ước lượng thường được sử dụng là ước lượng điểm (point estimation) và ước lượng khoảng (interval estimation).

- **Ước lượng điểm (Point Estimation):** là dùng một tham số thống kê mẫu đơn lẻ để ước lượng giá trị tham số của tổng thể.

$$\mu \approx \bar{x}$$

$$\sigma^2 \approx s^2$$

$$p \approx f$$



- **Ước lượng bằng khoảng tin cậy (Interval Estimation):** - Ước lượng bằng khoảng tin cậy chính là tìm ra khoảng ước lượng  $(G_1; G_2)$  cho tham số  $\theta$  trong tổng thể sao cho ứng với độ tin cậy (confidence) bằng  $\gamma$  cho trước,  $P(G_1 < \theta < G_2) = \gamma$ .

### 2.3.1 Ước lượng điểm (Point Estimation)

Một **ước lượng (estimator)** của một tham số (của tổng thể): là một biến ngẫu nhiên có giá trị phụ thuộc vào thông tin của mẫu, giá trị của nó là một xấp xỉ cho tham số chưa biết của tổng thể. Một giá trị cụ thể của biến ngẫu nhiên này gọi là một **giá trị ước lượng điểm**.

Xét đại lượng ngẫu nhiên  $X$  có phân phối  $F(x; \theta)$  với tham số  $\theta$  chưa biết.

Chọn một mẫu ngẫu nhiên cỡ  $n$  từ  $X_1, X_2, \dots, X_n$ .

Thống kê  $\hat{\theta} = h(X_1, X_2, \dots, X_n)$  gọi là một ước lượng điểm cho  $\theta$ .

Với một mẫu cụ thể  $(x_1, x_2, \dots, x_n)$ , ta gọi  $\hat{\theta} = h(x_1, x_2, \dots, x_n)$  là một giá trị ước lượng điểm cụ thể cho  $\theta$ .

## 2.4 Ước lượng bằng khoảng tin cậy (Interval Estimation)

Cho tham số  $\theta$  của tổng thể và  $X_1, X_2, \dots, X_n$  là các quan sát ngẫu nhiên. Ta gọi khoảng  $(c, d)$  là khoảng ước lượng (hay khoảng tin cậy) của tham số  $\theta$  với độ tin cậy  $\gamma$  nếu:  $P(c < \theta < d) = \gamma$ . Có thể nói, độ tin cậy  $\gamma$  cho khoảng ước lượng của tham số  $\theta$  chính là xác suất để ta đúng khi ước lượng tham số  $\theta$  bằng khoảng  $(c, d)$ . Ngược lại, xác suất mà ta cho phép sai khi ước lượng  $\theta$  được gọi là mức ý nghĩa. Kí hiệu là  $\alpha$ . Ta có  $\alpha + \gamma = 1$ .

Xác định khoảng ước lượng đối xứng của trung bình tổng thể dựa vào một mẫu đã cho, với kích thước là  $n$ , trung bình mẫu là  $\bar{x}$  và phương sai mẫu là  $s^2$  hoặc phương sai tổng thể là  $\sigma^2$ . Mục tiêu là xác định một khoảng ước lượng đối xứng xung quanh  $\mu$  của mẫu với một mức độ tin cậy cụ thể.

Để giải quyết vấn đề này, ta cần đi đến việc xác định epsilon ( $\epsilon$ ) - sai số ước lượng, dựa trên các thông tin đã biết về mẫu. Khoảng tin cậy sẽ được biểu diễn bằng khoảng  $\bar{x} \pm \epsilon$ . Tùy thuộc vào giả định về phân phối của dữ liệu và các thông tin đã biết về phương sai, cách tính  $\epsilon$  sẽ thay đổi như sau:



Dạng	Giả định	Loại	Nguồn sai số	Khoảng tin cậy
Tỷ lệ	$n \geq 30$	Đối xứng	$\epsilon = z_{\alpha/2} \sqrt{\frac{f(1-f)}{n}}$	$f - \epsilon < p < f + \epsilon$
		Bên phải	$\epsilon = z_\alpha \sqrt{\frac{f(1-f)}{n}}$	$0 < p < f + z_\alpha \sqrt{\frac{f(1-f)}{n}}$
		Bên trái	$\epsilon = z_\alpha \sqrt{\frac{f(1-f)}{n}}$	$f - z_\alpha \sqrt{\frac{f(1-f)}{n}} < p < 1$
Trung bình	$X \sim N(\mu, \sigma^2)$ Đã biết $\sigma$	Đối xứng	$\epsilon = z_{\alpha/2} \frac{\sigma}{\sqrt{n}}$	$\bar{x} - \epsilon < \mu < \bar{x} + \epsilon$
		Bên phải	$\epsilon = z_\alpha \frac{\sigma}{\sqrt{n}}$	$-\infty < \mu < \bar{x} + z_\alpha \frac{\sigma}{\sqrt{n}}$
		Bên trái	$\epsilon = z_\alpha \frac{\sigma}{\sqrt{n}}$	$\bar{x} - z_\alpha \frac{\sigma}{\sqrt{n}} < \mu < \infty$
	$X \sim N(\mu, \sigma^2)$ Chưa biết $\sigma$	Đối xứng	$\epsilon = t_{\alpha/2;n-1} \frac{s}{\sqrt{n}}$	$\bar{x} - \epsilon < \mu < \bar{x} + \epsilon$
		Bên phải	$\epsilon = t_{\alpha;n-1} \frac{s}{\sqrt{n}}$	$-\infty < \mu < \bar{x} + t_{\alpha;n-1} \frac{s}{\sqrt{n}}$
		Bên trái	$\epsilon = t_{\alpha;n-1} \frac{s}{\sqrt{n}}$	$\bar{x} - t_{\alpha;n-1} \frac{s}{\sqrt{n}} < \mu < \infty$
Phân phối tùy ý, mẫu lớn ( $n \geq 30$ ). Nếu chưa biết $\sigma$ thì thay bằng $s$		Đối xứng	$\epsilon = z_{\alpha/2} \frac{\sigma}{\sqrt{n}}$	$\bar{x} - \epsilon < \mu < \bar{x} + \epsilon$
		Bên phải	$\epsilon = z_\alpha \frac{\sigma}{\sqrt{n}}$	$-\infty < \mu < \bar{x} + z_\alpha \frac{\sigma}{\sqrt{n}}$
		Bên trái	$\epsilon = z_\alpha \frac{\sigma}{\sqrt{n}}$	$\bar{x} - z_\alpha \frac{\sigma}{\sqrt{n}} < \mu < \infty$

Table 2.1: Phân phối XS của trung bình mẫu và tỷ lệ mẫu

## 2.5 Kiểm định giả thuyết thống kê

### 2.5.1 Khái niệm chung về kiểm định

Trong thống kê, **kiểm định** (hypothesis testing) là quá trình đánh giá một giả thuyết về dữ liệu để xác định xem liệu có đủ bằng chứng để chấp nhận hay bác bỏ giả thuyết đó. Mục tiêu của kiểm định là đưa ra quyết định dựa trên dữ liệu mẫu có sẵn để rút ra những kết luận về tổng thể.

Quá trình kiểm định thường bắt đầu bằng việc xây dựng hai giả thuyết:

- **Giả thuyết không** (null hypothesis, ký hiệu  $H_0$ ): là giả thiết về yếu tố cần kiểm định của tổng thể ở trạng thái bình thường, không chịu tác động của các hiện tượng liên quan. Yếu tố trong  $H_0$  phải được xác định cụ thể.



- **Giả thuyết thay thế - giả thuyết đối (alternative hypothesis, ký hiệu  $H_1$ ):** là một mệnh đề mâu thuẫn với  $H_0$ ,  $H_1$  thể hiện xu hướng cần kiểm định.

**Miền Bác Bỏ (Rejection Region):** là miền số thực thỏa  $P(G \in RR | H_0 \text{ đúng}) = \alpha$ .  $\alpha$  là một số khá bé, thường không quá 10% và được gọi là mức ý nghĩa của kiểm định. Một ký hiệu khác của miền bác bỏ được dùng trong bài:  $W_\alpha$

**Miền Chấp Nhận (Acceptance Region):** phần bù của miền bác bỏ trong R

**Tiêu chuẩn kiểm định:** là hàm thống kê  $G = G(X_1, X_2, \dots, X_n, \theta_0)$ , xây dựng trên mẫu ngẫu nhiên  $W = (X_1, X_2, \dots, X_n)$  và tham số  $\theta_0$  liên quan đến  $H_0$ ; Điều kiện đặt ra với thống kê G là nếu  $H_0$  đúng thì quy luật phân phối xác suất của G phải hoàn toàn xác định.

### 2.5.2 Quy tắc kiểm định:

Từ mẫu thực nghiệm, ta tính được một giá trị cụ thể của tiêu chuẩn kiểm định, gọi là **giá trị kiểm định thống kê**:

$$g_{qs} = G(x_1, x_2, \dots, x_n, \theta_0)$$

Theo nguyên lý xác suất bé, biến cỗ  $G \in RR$  có xác suất nhỏ nên với 1 mẫu thực nghiệm ngẫu nhiên, nó không thể xảy ra.

Do đó:

- + Nếu  $g_{qs} \in RR$  thì bác bỏ  $H_0$ , thừa nhận giả thiết  $H_1$ .
- + Nếu  $g_{qs} \notin RR$ : ta chưa đủ dữ liệu khẳng định  $H_0$  sai. Vì vậy ta chưa thể chứng minh được  $H_1$  đúng.

### 2.5.3 Các sai lầm trong bài toán kiểm định

Kết luận của một bài toán kiểm định có thể mắc các sai lầm sau:

- **Sai lầm loại I:** Bác bỏ giả thiết  $H_0$  trong khi  $H_0$  đúng. Xác suất mắc sai lầm này nếu  $H_0$  đúng chính bằng mức ý nghĩa  $\alpha$ . Nguyên nhân mắc phải sai lầm loại I thường có thể do kích thước mẫu quá nhỏ, có thể do phương pháp lấy mẫu ...
- **Sai lầm loại II:** Thừa nhận  $H_0$  trong khi  $H_0$  sai, tức là mặc dù thực tế  $H_1$  đúng nhưng giá trị thực nghiệm  $g_{qs}$  không thuộc RR.

Quyết định	Tình huống	
	$H_0$ đúng	$H_0$ sai
Bác bỏ $H_0$	Sai lầm loại I. Xác suất = $\alpha$	Quyết định đúng
Không bác bỏ $H_0$	Quyết định đúng	Sai lầm loại II. Xác suất = $\beta$



#### 2.5.4 Các bước thực hiện kiểm định

1. Phát biểu giả thuyết và đối thuyết của bài toán.
2. Tính giá trị thống kê kiểm định (tiêu chuẩn kiểm định) cho bài toán.
3. Xác định miền bác bỏ tốt nhất cho bài toán.
4. Dưa ra kết luận.

### 2.6 Các mô hình kiểm định được sử dụng trong báo cáo

#### 2.6.1 Phân tích phương sai (anova)

Phân tích phương sai là một mô hình dùng để xem xét sự biến động của một biến ngẫu nhiên định lượng X chịu tác động trực tiếp của một hay nhiều yếu tố nguyên nhân (định tính). Được làm hai loại là phân tích phương sai 1 yếu tố và phân tích phương sai 2 yếu tố.

#### 2.6.2 Phân tích phương sai 1 yếu tố

##### Giả thiết

- Các tổng thể có phân phối chuẩn  $N(\mu_i; \sigma_i^2)$ ,  $i = 1, 2, \dots, k$  với  $k$  là tổng thể (thông thường  $k \geq 3$ ).
- Phương sai các tổng thể chưa biết nhưng được giả định là bằng nhau ( $\sigma_1^2 = \sigma_2^2 = \dots = \sigma_k^2$ ).
- Các mẫu quan sát (từ  $k$  tổng thể) được lấy độc lập.

##### Các bước thực hiện

###### Bước 1: Đặt giả thiết kiểm định

$$H_0 : \mu_1 = \mu_2 = \dots = \mu_k,$$
$$H_1 : \exists \mu_i \neq \mu_j \quad (i \neq j)$$

**Bước 2:** Tính trung bình mẫu của các nhóm  $\bar{x}_1, \bar{x}_2, \dots, \bar{x}_k$  theo công thức:

$$\bar{x}_i = \frac{\sum_{j=1}^{n_i} x_{ij}}{n_i}, \quad (i = 1, 2, \dots, k)$$



**Bước 3:** Tính tổng các bình phương lệch (tổng bình phương):

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

**Bước 4:** Tính các phương sai:

$$MSW = \frac{SSW}{N - k}, \quad MSB = \frac{SSB}{k - 1}$$

**Bước 5:** Tính thông kê kiểm định (tiêu chuẩn kiểm định, giá trị quan sát):

$$F = \frac{MSB}{MSW}$$

**Bước 6:** Xác định miền bác bỏ của bài toán:

$$RR = (F_{\alpha; k-1; N-k}; +\infty)$$

Tìm giá trị  $F_{\alpha; k-1; N-k}$  tra bảng Fisher mức ý nghĩa  $\alpha$  và cột  $k - 1$  và dòng  $N - k$ .

**Bước 7:** Đưa ra kết luận:

- Nếu  $F > F_{\alpha; k-1; N-k} \iff F \in RR \Rightarrow$  Bác bỏ  $H_0$ , chấp nhận  $H_1$
- Nếu  $F < F_{\alpha; k-1; N-k} \iff F \notin RR \Rightarrow$  không bác bỏ  $H_0$  (chưa bác bỏ được  $H_0$ , chấp nhận  $H_0$ )

Table 2.2: Bảng tóm tắt ANOVA 1 yếu tố

Nguồn của sự biến thiên	SS	df	MS	F
Giữa các nhóm	SSB	k-1	MSB	$F = \frac{MSB}{MSW}$
Trong từng nhóm	SSW	N-k	MSW	
Toàn bộ	SST	N-1		



### 3 Tiết xử lý dữ liệu

#### 3.1 Đọc dữ liệu vào R

Ta đọc dữ liệu từ file dữ liệu đã cho dưới định dạng CSV vào R bằng hàm `read.csv()` như sau:

Listing 1: Đọc dữ liệu từ file trong R

```
1 df <- read.csv("./data_sets/All_GPUs.csv")
2 head(df, 5)
```

Sau khi đọc dữ liệu xong, ta có thể sử dụng hàm `head()` để hiển thị 5 dòng đầu tiên của dữ liệu nhằm kiểm tra xem dữ liệu đã được đọc đúng chưa (Hình ??).

```
> df <- read.csv("./data_sets/All_GPUs.csv")
> head(df, 5)
#> #> Architecture Best_Resolution Boost_Clock Core_Speed DVI_Connection Dedicated Direct_X
#> #> 1 Tesla G92b 738 MHz 2 Yes DX 10.0
#> #> 2 R600 XT \n- 2 Yes DX 10
#> #> 3 R600 PRO \n- 2 Yes DX 10
#> #> 4 RV630 \n- 2 Yes DX 10
#> #> 5 RV630 \n- 2 Yes DX 10
#> #> DisplayPort_Connection HDMI_Connection Integrated L2_Cache Manufacturer Max_Power Memory Memory_Bandwidth
#> #> 1 NA 0 No 0KB Nvidia 141 Watts 1024 MB 64GB/sec
#> #> 2 NA 0 No 0KB AMD 215 Watts 512 MB 106GB/sec
#> #> 3 NA 0 No 0KB AMD 200 Watts 512 MB 51.2GB/sec
#> #> 4 NA 0 No 0KB AMD 256 MB 36.8GB/sec
#> #> 5 NA 0 No 0KB AMD 45 Watts 256 MB 22.4GB/sec
#> #> Memory_Bus Memory_Speed Memory_Type Name Notebook_GPU Open_GL
#> #> 1 256 Bit 1000 MHz GDDR3 GeForce GTS 150 No 3.3
#> #> 2 512 Bit 828 MHz GDDR3 Radeon HD 2900 XT 512MB No 3.1
#> #> 3 256 Bit 800 MHz GDDR3 Radeon HD 2900 Pro No 3.1
#> #> 4 128 Bit 1150 MHz GDDR4 Radeon HD 2600 XT Diamond Edition No 3.3
#> #> 5 128 Bit 700 MHz GDDR3 Radeon HD 2600 XT No 3.1
#> #> PSU Pixel_Rate Power_Connector Process ROPs Release_Date Release_Price Resolution_WxH
#> #> 1 450 Watt & 38 Amps 12 GPixel/s None 55nm 16 \n01-Mar-2009 2560x1600
#> #> 2 550 Watt & 35 Amps 12 GPixel/s None 80nm 16 \n14-May-2007 2560x1600
#> #> 3 550 Watt & 35 Amps 10 GPixel/s None 80nm 16 \n07-Dec-2007 2560x1600
#> #> 4 400 Watt & 25 Amps 3 GPixel/s None 65nm 4 \n01-Jul-2007 2560x1600
#> #> 5 400 Watt & 25 Amps 3 GPixel/s None 65nm 4 \n28-Jun-2007 2560x1600
#> #> SLI_Crossfire Shader TMUs Texture_Rate VGA_Connection
#> #> 1 Yes 4 64 47 GTexel/s 0
#> #> 2 Yes 4 16 12 GTexel/s 0
#> #> 3 Yes 4 16 10 GTexel/s 0
#> #> 4 Yes 4 8 7 GTexel/s 0
#> #> 5 Yes 4 8 6 GTexel/s 0
```

Figure 3.1: Hiển thị 5 dòng đầu tiên của dữ liệu sau khi đọc vào R

Như vậy, ta đã hoàn thành việc đọc dữ liệu từ file CSV vào R và có thể tiến hành các bước tiền xử lý dữ liệu tiếp theo.



### 3.2 Làm sạch dữ liệu

Vì trong file dữ liệu ban đầu, có thể tồn tại các giá trị bị thiếu (NA) hoặc các giá trị không hợp lệ, ta cần thực hiện các bước làm sạch dữ liệu để đảm bảo tính chính xác và độ tin cậy của phân tích sau này.

Trước tiên, ta sẽ phải thay thế tất cả các giá trị rác, không hợp lệ thành giá trị *NA* trong R. Ví dụ, nếu một ô bất kì có giá trị là chuỗi rỗng "" hoặc ký tự đặc biệt như "N/A", ta sẽ thay thế chúng bằng *NA* như sau:

```

1 df <- df %>%
2   mutate(across(where(is.character), trimws))
3 df[df == ""] <- NA
4 df[df == "N/A"] <- NA
5 df[df == "NA "] <- NA
6 df[df == "-"] <- NA
7 df[df == "Unknown Release Date"] <- NA
8 # Chi lay nam san xuat, khong lay ngay cu the
9 df$Release_Date <- as.Date(df$Release_Date, format = "%d-%b-%Y")
10 df$Release_Date <- format(df$Release_Date, "%Y")

```

Listing 2: Thay thế giá trị rác thành NA

Sau khi thay thế các giá trị rác, nhóm nhận thấy rằng có rất nhiều yếu tố có số lượng *NA* lớn, điều này buộc nhóm phải lựa chọn giữa loại bỏ và chuẩn hóa. Trong nội dung bài báo cáo này, nhóm sẽ loại bỏ những đặc điểm (cột) có số lượng giá trị *NA* vượt quá 10% tổng số dòng dữ liệu. Để thực hiện việc này, ta có thể sử dụng đoạn mã sau:

```

1 # Dem so luong gia tri NA trong moi cot
2 missing_counts = freq.na(df)
3 # Ve do thi ty le du lieu khuyet
4 ggplot(missing_counts, aes(x = rownames(missing_counts), y =
  missing_counts[,2], )) +
  geom_bar(stat = "identity", fill = "cyan") +
  geom_text(aes(label = paste0(missing_counts[,2], "%")), vjust =
    -0.5, size = 2) +
  labs(title = "Missing rate", x = "Feature", y = "Rate (%)") +
  theme_minimal() +
  theme(axis.text.x = element_text(
    size = 10,

```

```

11    angle = 90,
12    hjust = 1
13  ))

```

Listing 3: Loại bỏ các cột có tỷ lệ NA > 10%

Kết quả trực quan hóa tỷ lệ dữ liệu khuyết thiêU được thể hiện trong Hình ??.

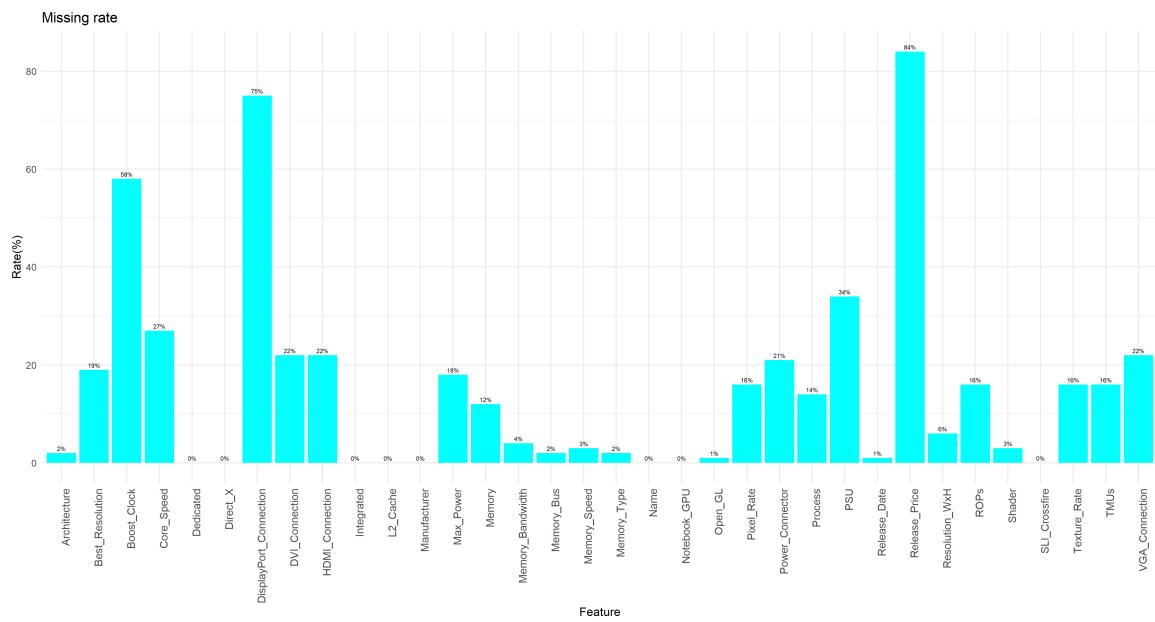


Figure 3.2: Tỷ lệ dữ liệu khuyết thiêU trong các đặc trưng

Tiếp theo sau đó, ta sẽ loại bỏ các cột có tỷ lệ giá trị NA vượt quá 10% tổng số dòng dữ liệu như sau:

```

1 missing_counts_df <- data.frame(
2   feature = rownames(missing_counts),
3   percent = missing_counts[,2]
4 )
5 cols_to_keep <- missing_counts_df$feature[missing_counts_df$
6   percent <= 10 & missing_counts_df$feature != "Architecture" &
7   missing_counts_df$feature != "Name"]
8 df_filtered <- df[, cols_to_keep, drop = FALSE]
9 head(df_filtered, 5)
10 df_filtered <- na.omit(df_filtered)

```

Listing 4: Loại bỏ các cột có tỷ lệ NA > 10%



Bằng câu lệnh `print(names(df_filtered))`, ta có thể kiểm tra lại các cột còn lại sau khi đã loại bỏ các cột có tỷ lệ giá trị *NA* vượt quá 10% (Hình ??).

```
> print(names(df_filtered))
[1] "Resolution_WxH"   "Memory_Bandwidth" "Shader"          "Memory_Speed"    "Memory_Bus"
[6] "Memory_Type"     "Open_GL"        "Release_Date"   "Dedicated"      "Integrated"
[11] "Direct_X"        "L2_Cache"       "Manufacturer"  "Notebook_GPU"  "SLI_Crossfire"
```

Figure 3.3: Các cột còn lại sau khi loại bỏ các cột có tỷ lệ NA > 10%

Mặc dù đã làm sạch dữ liệu bằng cách loại bỏ các cột có tỷ lệ khuyết hoặc số lượng giá trị *NA* cao, vẫn còn một yếu tố khiến việc phân tích dữ liệu trở nên khó khăn, đó là các đơn vị đo, do đó ta cần chuẩn hoá các đơn vị đo trong dữ liệu bằng cách loại bỏ chúng.

```
1 # Chuan hoa don vi do trong cac cot
2 remove_unit_cols <- c("Memory_Bandwidth", "Memory_Speed", "Memory_Bus",
3                         "Direct_X")
4 main_df <- df_filtered
5 main_df[remove_unit_cols] <- lapply(df_filtered[remove_unit_cols],
6                                     function(x) {
7                                         as.numeric(gsub("[^0-9.]", "", x))
8                                     })
9 clean_cache <- function(x) {
10     main <- as.numeric(sub("KB.*", "", x))           # 2304
11     mult      <- as.numeric(sub(".*\\"\\(x([0-9]+)\\)\\)", "\\\\$1", x))  #
12     2
13     if (is.na(mult)) mult <- 1
14     return(main * mult)
15 }
16 main_df$L2_Cache <- sapply(main_df$L2_Cache, clean_cache)
```

Listing 5: Chuẩn hoá các đơn vị đo trong dữ liệu

Dữ liệu đã được làm sạch được lưu vào biến `main_df` và được hiện trong Hình ??.

```
> head(main_df, 5)
#>   Manufacturer Release_Date Memory_Bandwidth Memory_Speed L2_Cache Open_GL Memory_Type Resolution_WxH
#> 1      Nvidia    2009          64.0        1000       0    3.3    GDDR3  2560x1600
#> 2       AMD     2007         106.0        828       0    3.1    GDDR3  2560x1600
#> 3       AMD     2007         51.2         800       0    3.1    GDDR3  2560x1600
#> 4       AMD     2007         36.8        1150       0    3.3    GDDR4  2560x1600
#> 5       AMD     2007         22.4         700       0    3.1    GDDR3  2560x1600
#>
#>   Direct_X Shader Memory_Bus Dedicated Integrated Notebook_GPU SLI_Crossfire
#> 1      10      4     256     Yes     No     No     Yes
#> 2      10      4     512     Yes     No     No     Yes
#> 3      10      4     256     Yes     No     No     Yes
#> 4      10      4     128     Yes     No     No     Yes
#> 5      10      4     128     Yes     No     No     Yes
```

Figure 3.4: Dữ liệu sau khi làm sạch



## 4 Thống kê suy diễn

### 4.1 Phân tích ANOVA hai yếu tố về ảnh hưởng của Hạng sản xuất và Loại bộ nhớ lên Băng thông bộ nhớ

#### 4.1.1 Mục đích kiểm định

Mục đích của kiểm định này là đánh giá xem:

1. Băng thông bộ nhớ trung bình có khác nhau giữa các **hạng** hay không.
2. Băng thông bộ nhớ trung bình có khác nhau giữa các **loại GPU** (tích hợp so với chuyên dụng) hay không.
3. Có tồn tại **tác động tương tác** giữa Hạng và Loại bộ nhớ đối với băng thông bộ nhớ hay không.

#### 4.1.2 Giả thiết nghiên cứu

##### 1. Ảnh hưởng của hạng

- **Giả thiết không**  $H_0$ : Băng thông bộ nhớ trung bình của các hạng là như nhau.
- **Giả thiết đối**  $H_1$ : Có ít nhất hai hạng có băng thông bộ nhớ trung bình khác nhau.

##### 2. Ảnh hưởng của loại GPU

- **Giả thiết không**  $H_0$ : Băng thông bộ nhớ trung bình giữa các loại GPU là như nhau.
- **Giả thiết đối**  $H_1$ : Có sự khác biệt băng thông trung bình giữa ít nhất hai loại GPU.

##### 3. Tác động tương tác giữa hai yếu tố

- **Giả thiết không**  $H_0$ : Không tồn tại tương tác giữa hạng và loại GPU (tác động của một yếu tố không phụ thuộc vào yếu tố kia).
- **Giả thiết đối**  $H_1$ : Có tương tác giữa hạng và loại GPU (tác động của một yếu tố phụ thuộc vào yếu tố còn lại).



#### 4.1.3 Quy trình thực hiện ANOVA hai yếu tố

**Bước 1: Nhập dữ liệu** Chuyển biến Manufacturer(Hãng) và Dedicated(Loại GPU) thành biến phân loại trong R:

```
1 main_df$Manufacturer <- as.factor(main_df$Manufacturer)
2 main_df$Dedicated      <- as.factor(main_df$Dedicated)
```

Listing 6: Chuyển biến phân loại trong R

#### Bước 2: Kiểm tra cấu trúc dữ liệu

Đảm bảo Manufacturer và Dedicated là biến phân loại, sử dụng lệnh:

```
1 str(main_df)
```

Listing 7: Kiểm tra cấu trúc dữ liệu

```
> main_df$Manufacturer <- as.factor(main_df$Manufacturer)
> main_df$Dedicated <- as.factor(main_df$Dedicated)
> str(main_df$Manufacturer)
Factor w/ 4 levels "AMD","ATI","Intel",..: 4 1 1 1 1 1 1 1 ...
> str(main_df$Dedicated)
Factor w/ 2 levels "No","Yes": 2 2 2 2 2 2 2 2 ...
```

Figure 4.1: Kết quả kiểm tra kiểu dữ liệu với lệnh str() trong R

Để kiểm tra các thông số cơ bản của các biến cần kiểm định, sử dụng lệnh:

```
1 summary(main_df$Memory_Bandwidth)
2 table(main_df$Manufacturer)
3 table(main_df$Dedicated)
```

Listing 8: Kiểm tra thông số cơ bản của các biến

Kết quả:

```
> # Basic summary statistics
> summary(main_df$Memory_Bandwidth)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   2.0    29.9   112.0   144.1   219.4   1280.0
> table(main_df$Manufacturer)

   AMD    ATI    Intel    Nvidia
   1179     78    153    1618
> table(main_df$Dedicated)

   No    Yes
   216  2812
> |
```

Figure 4.2: Thống kê mô tả cơ bản của biến Memory\_Bandwidth(Băng thông bộ nhớ)



Tóm tắt thống kê cho biến **Memory\_Bandwidth**:

min = 2.0, Q1 = 29.9, Median = 112.0, Mean = 144.1, Q3 = 219.4, max = 1280.

Nhận xét:

- Bảng thông số có phân bố lệch phải mạnh (max = 1280).
- Số mẫu theo hãng rất lệch (AMD 1179, ATI 78, Intel 153, Nvidia 1618).
- GPU chuyên dụng (Yes) chiếm 2812/3028 mẫu, lệch phân bố so với GPU tích hợp (No = 216).

### Bước 3: Thực hiện thống kê ANOVA hai yếu tố

Memory\_Bandwidth ~ Manufacturer \* Dedicated

Xây dựng mô hình ANOVA hai yếu tố trong R:

```
1 anova_model <- aov(Memory_Bandwidth ~ Manufacturer * Dedicated ,  
                      data = main_df)
```

Listing 9: Xây dựng mô hình ANOVA hai yếu tố trong R

Kiểm tra giả định:

a) Kiểm định chuẩn hoá phần dư bằng Shapiro-Wilk test:

Giả thuyết không  $H_0$  : Phần dư có phân phối chuẩn  
Giả thuyết đối  $H_1$  : Phần dư không có phân phối chuẩn

Table 4.1: Kết quả kiểm định Shapiro-Wilk cho phần dư của mô hình

```
1 shapiro.test(residuals(anova_model))
```

Listing 10: Kiểm định chuẩn hoá phần dư bằng Shapiro-Wilk test

```
> # Normality of residuals  
> shapiro_test <- shapiro.test(residuals(anova_model))  
> print(shapiro_test)  
  
shapiro-wilk normality test  
  
data: residuals(anova_model)  
W = 0.85823, p-value < 2.2e-16  
|
```

Figure 4.3: Kết quả kiểm định Shapiro-Wilk cho phần dư của mô hình ANOVA



**Nhận xét:**Vì  $p-value < 0.05$  nên ta bác bỏ  $H_0$ , thừa nhận  $H_1$ , do đó phần dư không tuân theo phân phối chuẩn.

- b) Kiểm định phương sai đồng nhất bằng Levene's test:

```
1 library(car)
2 leveneTest(Memory_Bandwidth ~ Manufacturer * Dedicated, data
= main_df)
```

Listing 11: Kiểm định phương sai đồng nhất bằng Levene's test



Bảng ANOVA từ R:

```
> # Two-way ANOVA model
> anova_model <- aov(Memory_Bandwidth ~ Manufacturer * Dedicated, data = main_df)
> # Show ANOVA summary table
> summary(anova_model)
      Df  Sum Sq Mean Sq F value    Pr(>F)
Manufacturer          3 2798908 932969  54.088 < 2e-16 ***
Dedicated             1   879716  879716  51.001 1.15e-12 ***
Manufacturer:Dedicated 2   83757   41878   2.428   0.0884 .
Residuals            3021 52109247   17249
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
>
```

Figure 4.4: Bảng ANOVA hai yếu tố được tạo bởi mô hình trong R

Ta có thể vẽ lại bảng như sau:

Table 4.2: Bảng phân tích phương sai hai yếu tố đối với biến *Memory\_Bandwidth*

Nguồn biến thiên	SS	df	MS	F	p-value
Manufacturer	2,798,908	3	932,969	54.088	$< 2 \times 10^{-16}$
Dedicated	879,716	1	879,716	51.001	$1.15 \times 10^{-12}$
Tương tác (Manufacturer $\times$ Dedicated)	83,757	2	41,878	2.428	0.0884
Sai số	52,109,247	3021	17,249	—	—

Nhận xét:

- **Manufacturer** có ảnh hưởng rất đáng kể đến băng thông bộ nhớ.
- **Dedicated** có ảnh hưởng rất đáng kể.
- **Tương tác Manufacturer  $\times$  Dedicated không có ý nghĩa thống kê** ( $p = 0.0884 > 0.05$ ).



#### Bước 4: Kiểm định giả định

(a) **Kiểm định chuẩn hoá phần dư** Từ kết quả kiểm định Shapiro-Wilk mà đoạn code trên tạo ra, ta thấy rằng:

$$W = 0.85823, \quad p < 2.2 \times 10^{-16}$$

Và vẽ được biểu đồ Q-Q bằng cách sử dụng lệnh:

```
qqnorm(residuals(anova_model))
qqline(residuals(anova_model), col = "red")
```

Kết quả như sau:

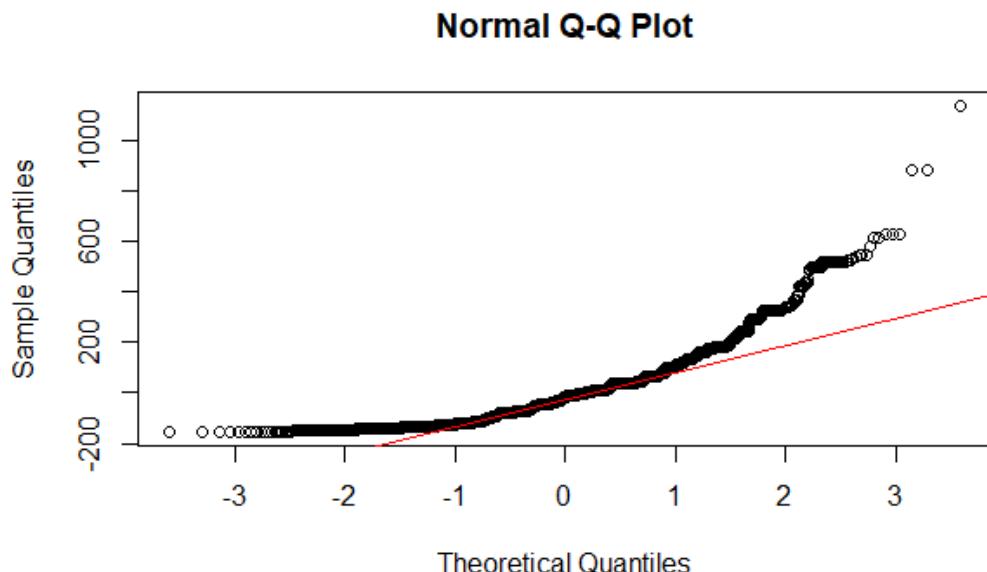


Figure 4.5: Biểu đồ Q-Q kiểm tra phân phối chuẩn của phần dư

**Nhận xét:** Phần dư **không** tuân theo phân phối chuẩn. (Biểu đồ Q-Q cũng cho thấy lệch phải mạnh.)



```
> # Homogeneity of variances
> levene_test <- LeveneTest(Memory_Bandwidth ~ Manufacturer * Dedicated, data = main_df)
> print(levene_test)
Levene's Test for Homogeneity of Variance (center = median)
  Df F value    Pr(>F)
group      6   34.63 < 2.2e-16 ***
               3021
---
signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> |
```

Figure 4.6: Kết quả kiểm định Levene về tính đồng nhất phương sai

(b) **Kiểm định phương sai đồng nhất (Levene)** Từ kết quả kiểm định Levene mà đoạn code trên tạo ra, ta thấy rằng:

$$F = 34.63, \quad p < 2.2 \times 10^{-16}$$

**Nhận xét:** Phương sai giữa các nhóm **không đồng nhất**.

**Kết luận về giả định:**

- Cả hai giả định quan trọng của ANOVA đều bị vi phạm.
- Tuy ANOVA vẫn thường “mạnh” với kích thước mẫu lớn ( $n = 3028$ ), nhưng kết luận cần diễn giải thận trọng.

**Bước 5: Kiểm định hậu nghiệm** Sử dụng lệnh:

TukeyHSD(anova\_model)

\$Manufacturer	diff	lwr	upr	p adj
ATI-AMD	74.35187	34.882820	113.82092	0.0000080
Intel-AMD	-110.80248	-139.812145	-81.79281	0.0000000
Nvidia-AMD	19.77607	6.849207	32.70294	0.0004986
Intel-ATI	-185.15435	-232.122796	-138.18590	0.0000000
Nvidia-ATI	-54.57579	-93.711178	-15.44041	0.0019448
Nvidia-Intel	130.57855	102.024516	159.13259	0.0000000

Figure 4.7: Kết quả kiểm định Tukey HSD cho yếu tố Hãng sản xuất(Manufacturer)

(1) **Hãng sản xuất** Từ kết quả trên, ta thấy rằng tất cả các cặp hãng đều khác biệt có ý nghĩa thống kê, ví dụ:

$$\text{Intel - AMD} : -110.80, \quad p < 10^{-7}$$

$$\text{ATI - AMD} : +74.35, \quad p = 8 \cdot 10^{-6}$$



Nvidia - AMD : +19.78,  $p = 0.00049$

**Nhận xét:** Băng thông bộ nhớ trung bình khác biệt đáng kể giữa các hãng.

```
$ Dedicated
      diff      lwr      upr     p adj
Yes-No 38.93941 20.75716 57.12165 2.76e-05
```

Figure 4.8: Kết quả kiểm định Tukey HSD cho yếu tố loại GPU (Dedicated)

**(2) Loại GPU** Từ kết quả trên, ta thấy rằng:

$$\text{Yes - No} = 38.94, \quad p = 2.76 \times 10^{-5}$$

**Nhận xét:** GPU chuyên dụng có băng thông cao hơn GPU tích hợp.



\$`Manufacturer:Dedicated`					
	diff	lwr	upr	p	adj
ATI:No-AMD:No	NA	NA	NA	NA	NA
Intel:No-AMD:No	3.410937	-56.183216	63.005091	0.9999998	
Nvidia:No-AMD:No	-12.989062	-218.290051	192.311926	0.9999995	
AMD:Yes-AMD:No	120.344996	69.143099	171.546892	0.0000000	
ATI:Yes-AMD:No	188.164143	120.980515	255.347770	0.0000000	
Intel:Yes-AMD:No	-8.864063	-193.836118	176.107993	0.9999999	
Nvidia:Yes-AMD:No	133.951613	83.181189	184.722037	0.0000000	
Intel:No-ATI:No	NA	NA	NA	NA	NA
Nvidia:No-ATI:No	NA	NA	NA	NA	NA
AMD:Yes-ATI:No	NA	NA	NA	NA	NA
ATI:Yes-ATI:No	NA	NA	NA	NA	NA
Intel:Yes-ATI:No	NA	NA	NA	NA	NA
Nvidia:Yes-ATI:No	NA	NA	NA	NA	NA
Nvidia:No-Intel:No	-16.400000	-218.244773	185.444773	0.9999973	
AMD:Yes-Intel:No	116.934058	82.085088	151.783029	0.0000000	
ATI:Yes-Intel:No	184.753205	129.017612	240.488798	0.0000000	
Intel:Yes-Intel:No	-12.275000	-193.403346	168.853346	0.9999992	
Nvidia:Yes-Intel:No	130.540675	96.328799	164.752552	0.0000000	
AMD:Yes-Nvidia:No	133.334058	-66.194094	332.862211	0.4637330	
ATI:Yes-Nvidia:No	201.153205	-3.061117	405.367527	0.0570290	
Intel:Yes-Nvidia:No	4.125000	-263.091225	271.341225	1.0000000	
Nvidia:Yes-Nvidia:No	146.940675	-52.477191	346.358541	0.3309924	
ATI:Yes-AMD:Yes	67.819147	21.164806	114.473487	0.0002874	
Intel:Yes-AMD:Yes	-129.209058	-307.752188	49.334071	0.3549605	
Nvidia:Yes-AMD:Yes	13.606617	-1.905431	29.118665	0.1353020	
Intel:Yes-ATI:Yes	-197.028205	-380.793421	-13.262989	0.0255646	
Nvidia:Yes-ATI:Yes	-54.212530	-100.392929	-8.032131	0.0089720	
Nvidia:Yes-Intel:Yes	142.815675	-35.604197	321.235548	0.2281507	

Figure 4.9: Kết quả kiểm định Tukey HSD cho tương tác Hạng × Loại GPU

(3) **Tương tác** Từ kết quả trên, ta thấy rằng hầu hết các cặp đều không có ý nghĩa thống kê (nhiều trường hợp NA do thiếu dữ liệu).

**Nhận xét:** Không tìm thấy bằng chứng thuyết phục rằng tác động giữa hạng và loại GPU ảnh hưởng bằng thông theo cách khác biệt đáng kể.

Bước 6: Vẽ đồ thị Biểu đồ tương tác giữa Hãng và Loại GPU:

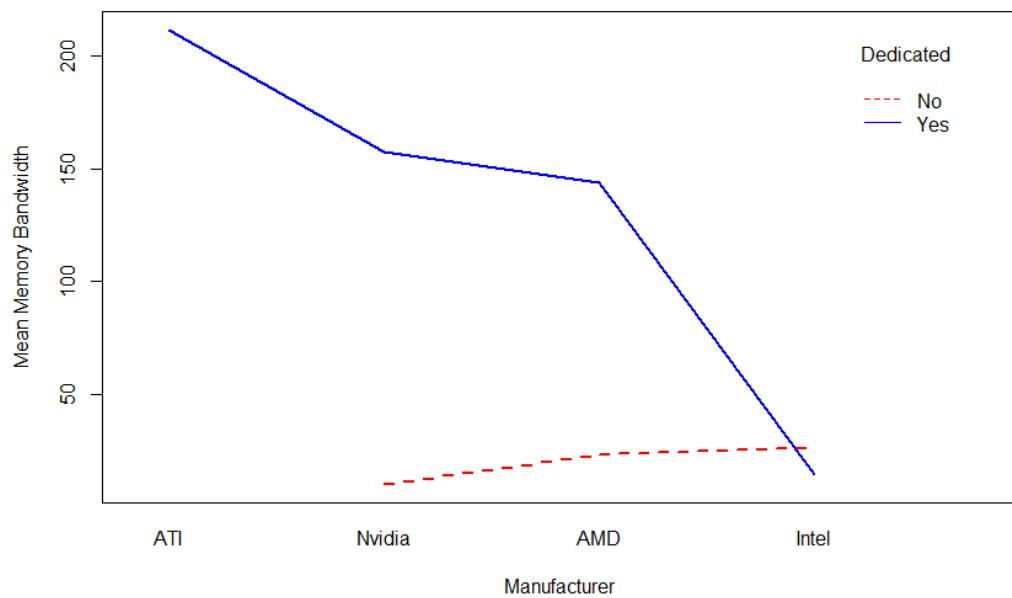


Figure 4.10: Biểu đồ tương tác giữa Hãng và Loại GPU



Biểu đồ trực quan hóa băng thông bộ nhớ theo Hãng và Loại GPU:

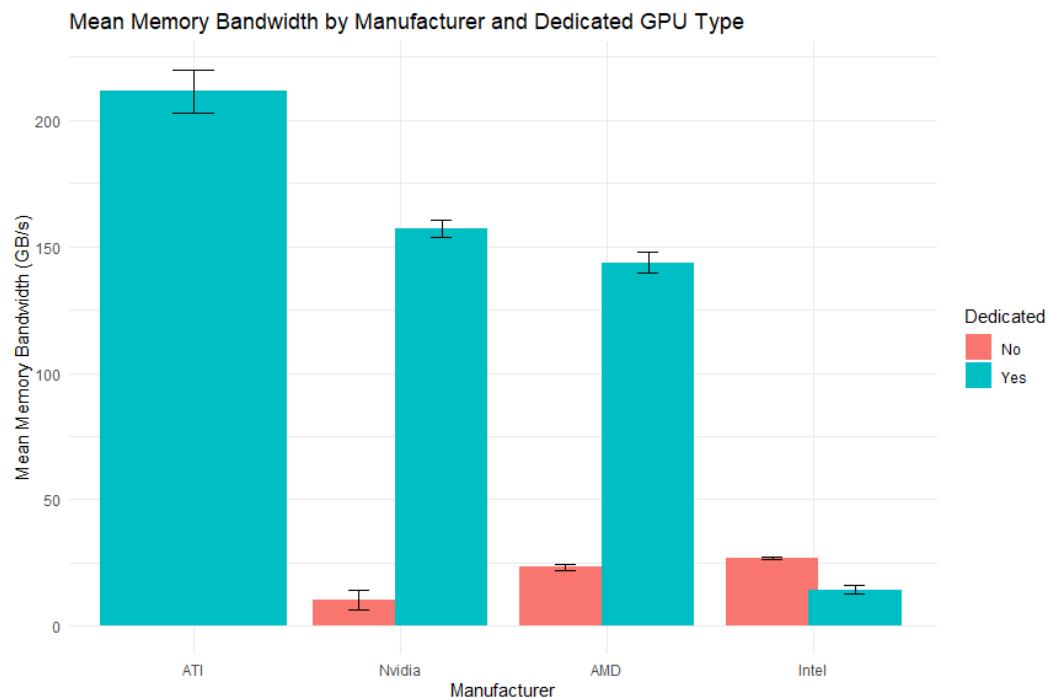


Figure 4.11: Biểu đồ trực quan hóa băng thông bộ nhớ theo Hãng và Loại GPU



## 4.2 Kết luận tổng hợp

Dựa trên kết quả phân tích:

- **Hãng sản xuất có ảnh hưởng rất mạnh** đến băng thông bộ nhớ ( $p < 2e-16$ ).
- **Loại GPU (Chuyên dụng/Tích hợp) cũng có ảnh hưởng đáng kể** ( $p = 1.15e-12$ ).
- **Không có tương tác rõ rệt** giữa hai yếu tố ( $p = 0.0884$ ).
- Tuy nhiên, **giả định chuẩn hoá phần dư và đồng nhất phương sai đều bị vi phạm**. Điều này không vô hiệu hoá ANOVA do mẫu rất lớn, nhưng kết luận nên được diễn giải cẩn trọng.
- Phân tích hậu nghiệm cho thấy tất cả hãng đều khác nhau có ý nghĩa, và GPU chuyên dụng vượt trội GPU tích hợp.

**Tóm tắt:** *Hãng sản xuất và loại GPU đều ảnh hưởng mạnh đến băng thông bộ nhớ, nhưng không có tương tác giữa hai yếu tố. Tuy nhiên, phân bố phần dư cho thấy ANOVA có thể bị ảnh hưởng bởi vi phạm giả định, do đó có thể xem xét thêm biến đổi dữ liệu (log-transform) trong phân tích mở rộng.*