ĐẠI 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

THỐNG KÊ DỮ LIỆU BÁN LẺ GIAO DỊCH CỦA CỬA HÀNG ĐIỆN TỬ

GVHD: Cô Nguyễn Kiều Dung

SV thực hiện:	Bùi Nguyễn Hoàng Thọ	2333017
	Trần Văn Được	2210782
	Trịnh Duy Nghiêm	2312256

Tp. Hồ Chí Minh, Tháng 8/2025

Tên nhóm: MT01

STT	Họ tên SV	MSSV	Tên lớp	Ngành / Khoa
1	**Bùi Nguyễn Hoàng Thọ**	2333017	DT03	KH-KT máy tính
2	Trần Văn Được	2210782	DT03	Điện - Điện tử
3	Trịnh Duy Nghiêm	2312256	DT03	KH-KT máy tính

Mục lục

Giới thiệu

Trong thời đại khoa học công nghệ phát triển mạnh mẽ, nhu cầu sử dụng linh kiện và thiết bị điện tử ngày càng gia tăng. Điều này đã đặt ra yêu cầu cao hơn về chất lượng sản phẩm và dịch vụ, buộc các cửa hàng điện tử phải đối mặt với sự cạnh tranh khốc liệt. Để thu hút và giữ chân khách hàng, nhiều chương trình khuyến mãi cùng các chính sách ưu đãi được triển khai liên tục, đáp ứng tốt hơn nhu cầu và nâng cao chất lượng đời sống người tiêu dùng.

Để đạt được những mục tiêu này, các cửa hàng điện tử không ngừng đầu tư vào việc phân tích dữ liệu giao dịch bán lẻ. Đây là một bước quan trọng, giúp khảo sát nhu cầu khách hàng, từ đó xây dựng các chiến lược kinh doanh hiệu quả và tối ưu hóa phương thức tiếp cận thị trường.

Với mong muốn có một góc nhìn tổng quát hơn về hoạt động phân tích trong kinh doanh, chúng em đã chọn đề tài "Thống kê dữ liệu bán lẻ giao dịch của cửa hàng điện tử". Qua đó, chúng em hy vọng có thể tự mình thử nghiệm và áp dụng các kiến thức về xác suất và thống kê đã được tích lũy trong quá trình học tập.

Trong quá trình thực hiện báo cáo, nếu có bất kỳ thiếu sót nào, nhóm rất mong nhận được sự góp ý từ thầy để hoàn thiện tốt hơn.

1 Tóm Tắt Dữ Liệu

1.1 Ngữ cảnh

Bộ dữ liệu được sử dụng lấy từ Kaggle, mô phỏng các giao dịch bán lẻ tại một cửa hàng điện tử. Nó chứa thông tin chi tiết về đơn hàng, khách hàng, sản phẩm mua, và phản hồi của khách hàng.

1.2 Cách dữ liệu được thu thập

Bộ dữ liệu trên là tổng hợp những đơn hàng sản phẩm đã được thanh toán tại một cửa hàng điện tử. Nó đi kèm với ngày thanh toán, thông tin sản phẩm, và thông tin khách hàng.

1.3 Các loại biến và quan trắc

Nhóm đã tổng hợp lại được tổng cộng 1,000 giá trị quan trắc liên quan đến các giao dịch và 3 giá trị liên quan đến thông tin các kho hàng.

Trong quá trình tổng hợp, các biến trong bộ dữ liệu được đưa ra như sau:

- **Biến định tính (Categorical variables):**
  - `order_id`: Mã đơn hàng.
  - `customer_id`: ID khách hàng.
  - `nearest_warehouse`: Tên kho hàng gần nhất.
  - `season`: Mùa trong năm.
  - `latest_customer_review`: Nhận xét từ khách hàng.
- **Biến định lượng (Numerical variables):**
  - `order_price`: Giá trị đơn hàng trước khuyến mãi.
  - `delivery_charges`: Phí giao hàng.
  - `coupon_discount`: Giá trị khuyến mãi áp dụng.
  - `order_total`: Giá trị đơn hàng sau khuyến mãi.
  - `customer_lat` và `customer_long`: Vĩ độ và kinh độ của khách hàng.
  - `distance_to_nearest_warehouse`: Khoảng cách đến kho gần nhất.
- **Biến Boolean:**
  - `is_expedited_delivery`: Có giao hàng nhanh hay không.
  - `is_happy_customer`: Khách hàng hài lòng hay không.

2 Cơ sở lý thuyết (Kiến thức nền)

2.1 Giới thiệu về công cụ sử dụng thống kê

2.1.1 Ngôn ngữ R

R là một công cụ rất mạnh cho học máy, thống kê và phân tích dữ liệu. Nó là một ngôn ngữ lập trình, có thể sử dụng trên bất kỳ hệ điều hành nào do tính chất độc lập nền tảng (*platform-independent*). R có khả năng tích hợp với các ngôn ngữ khác như C, C++ và cho phép tương tác với nhiều nguồn dữ liệu cũng như các gói thống kê như SAS, SPSS.

2.1.2 Công cụ RStudio

RStudio là một môi trường phát triển tích hợp (IDE - Integrated Development Environment) dành riêng cho R. RStudio cung cấp giao diện trực quan và các chức năng thuận tiện để làm việc với R mà không cần chạy R trực tiếp.

2.2 Các khái niệm về thành phần và phương pháp sử dụng thống kê

2.2.1 Biến định lượng

Biến định lượng là những biến có thể đo lường được bằng các con số và thực hiện các phép toán số học như cộng, trừ, nhân, chia. Biến định lượng cung cấp thông tin về "mức độ"hoặc "số lượng"của một đặc điểm cụ thể.

2.2.2 Phân phối chuẩn

Khái niệm: Phân phối chuẩn, còn gọi là phân phối Gauss (theo tên nhà toán học Carl Friedrich Gauss), là một loại phân phối liên tục. Đặc điểm nổi bật của phân phối chuẩn là:

- Đối xứng qua giá trị trung bình.
- Có hình dạng chuông.
- Các giá trị được phân bố đều xung quanh trung bình và giảm dần khi xa trung bình.

2.2.3 Kiến thức thống kê mô tả

Các chỉ số đo lường xu hướng tập trung:

- **Trung bình (Mean):** Là giá trị trung tâm của bộ dữ liệu, được tính bằng công thức:

$$\text{Mean} = \frac{\sum x_i}{n}$$

trong đó x_i là các giá trị quan sát, n là số lượng giá trị.

- **Trung vị (Median):** Là giá trị nằm ở giữa khi các giá trị được sắp xếp theo thứ tự tăng dần.
- **Mốt (Mode):** Là giá trị xuất hiện nhiều nhất trong bộ dữ liệu.

Các chỉ số đo lường sự phân tán

- **Phương sai (Variance):** Đo lường mức độ phân tán của dữ liệu xung quanh giá trị trung bình, được tính bằng công thức:
$$\text{Variance} = \frac{\sum(x_i - \bar{x})^2}{n}$$
- **Độ lệch chuẩn (Standard Deviation):** Là căn bậc hai của phương sai:
$$\text{SD} = \sqrt{\text{Variance}}$$
- **Khoảng (Range):** Là hiệu giữa giá trị lớn nhất và giá trị nhỏ nhất trong bộ dữ liệu.

Các chỉ số mô tả sự phân bố

- **Tứ phân vị (Quartiles):** Là các giá trị chia dữ liệu thành bốn phần bằng nhau.
- **Khoảng tứ phân vị hay độ khoảng giữa (Interquartile Range - IQR):** Là khoảng cách giữa tứ phân vị thứ ba (Q_3) và tứ phân vị thứ nhất (Q_1), được tính bằng:
$$\text{IQR} = Q_3 - Q_1$$

IQR là một chỉ số quan trọng giúp xác định sự phân tán dữ liệu và phát hiện các giá trị ngoại lai (outliers). Các giá trị được coi là ngoại lai nếu nằm ngoài khoảng:

$$[Q_1 - 1.5 \times \text{IQR}, Q_3 + 1.5 \times \text{IQR}]$$

Biểu đồ sử dụng thống kê

- **Biểu đồ cột (Barplot):** Dùng so sánh các giá trị khác nhau.
- **Biểu đồ hộp (Box plot):** Thể hiện các tứ phân vị và phát hiện giá trị ngoại lai.
- **Biểu đồ Q-Q (Quantile-Quantile plot):** Kiểm tra dữ liệu có tuân theo phân phối chuẩn hay không.
- **Biểu đồ corrplot**: thể hiện sự tương quan của dữ liệu.

2.3 Kiến thức thống kê suy luận (Tóm tắt)

2.3.1 Lý thuyết Kiểm định thống kê

Kiểm định là quá trình sử dụng dữ liệu mẫu để đưa ra quyết định về một giả thuyết nào đó liên quan đến tổng thể.

- **Giả thuyết không (Null Hypothesis - H_0):** Giả thuyết ban đầu cho rằng không có sự khác biệt hoặc mối quan hệ nào giữa các biến.
- **Giả thuyết thay thế (Alternative Hypothesis - H_1):** Giả thuyết cho rằng có sự khác biệt hoặc mối quan hệ giữa các biến.
- **Mức ý nghĩa (Significance Level - α):** Xác định ngưỡng để bác bỏ giả thuyết không, thường là 0.05 hoặc 0.01.
- **Giá trị p (p-value):** Xác suất để quan sát được dữ liệu mẫu nếu giả thuyết không là đúng. Nếu p-value nhỏ hơn mức ý nghĩa α, ta bác bỏ giả thuyết không.

2.3.2 Kiểm định z-test cho 1 mẫu

2.3.3 Kiểm định t-test cho 2 mẫu độc lập

- **Định nghĩa:** Kiểm định t-test là một phương pháp thống kê dùng để so sánh trung bình của hai nhóm độc lập hoặc hai mẫu liên quan nhằm xác định xem có sự khác biệt có ý nghĩa thống kê giữa chúng hay không.
- **Ý nghĩa:**
  - Phù hợp cho dữ liệu có phân phối tuỳ ý và mẫu lớn ($n > 30$)
  - Giúp xác định xem hai nhóm có khác biệt về trung bình hay không.

Công thức tổng quát cho hai mẫu độc lập:

$$T = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}}$$

Trong đó:

- $\bar{X}_1, \bar{X}_2$: trung bình mẫu của hai nhóm (ví dụ: mùa xuân và mùa hè)
- s_1, s_2: độ lệch chuẩn mẫu của hai nhóm
- n_1, n_2: kích thước mẫu của hai nhóm

Bậc tự do (degrees of freedom) được ước lượng theo công thức Welch–Satterthwaite:

$$df = \frac{\left(\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}\right)^2}{\frac{\left(\frac{s_1^2}{n_1}\right)^2}{n_1 - 1} + \frac{\left(\frac{s_2^2}{n_2}\right)^2}{n_2 - 1}}$$

So sánh giá trị $|T|$ với ngưỡng $t_{\alpha/2,df}$ để đưa ra kết luận.

2.3.4 Kiểm định Kruskal-Wallis

- **Định nghĩa:** Kiểm định phi tham số dùng để so sánh trung vị của **ba hoặc nhiều nhóm độc lập** nhằm xác định xem có sự khác biệt đáng kể giữa các nhóm khi dữ liệu không theo phân phối chuẩn.
- **Ý nghĩa:**
  - Phù hợp cho dữ liệu không chuẩn hoặc dữ liệu thứ bậc.
  - Kiểm tra sự khác biệt trung vị giữa các nhóm độc lập.

2.3.5 Kiểm định Wilcoxon-Mann-Whitney

- **Định nghĩa:** Kiểm định phi tham số dùng để so sánh **hai mẫu độc lập**, kiểm tra xem hai mẫu có cùng phân phối hay không.
- **Ý nghĩa:**
  - Thích hợp khi dữ liệu không chuẩn hoặc có ngoại lệ lớn.
  - Ứng dụng cho dữ liệu định lượng hoặc thứ bậc để kiểm tra sự khác biệt về phân phối.

2.3.6 Kiểm định Wilcoxon Signed-Rank

- **Định nghĩa:** Kiểm định phi tham số dùng để so sánh trung vị của **hai tập dữ liệu liên quan (cặp đôi)**, thay thế kiểm định tham số khi dữ liệu không tuân theo phân phối chuẩn.
- **Ý nghĩa:**

- Kiểm tra sự khác biệt giữa hai tập dữ liệu liên quan.
- Thích hợp cho dữ liệu định lượng không chuẩn hoặc dữ liệu thứ bậc.

2.3.7 Phân tích hậu kiểm

Post hoc test: được sử dụng sau khi có kết quả có ý nghĩa thống kê để xác định sự khác biệt giữa các nhóm. Kiểm định **Bonferroni** là phương pháp đơn giản, điều chỉnh mức ý nghĩa α bằng cách chia cho số lượng so sánh, giúp kiểm soát tỷ lệ lỗi loại I khi thực hiện các bài kiểm tra tính độc lập giữa các nhóm.

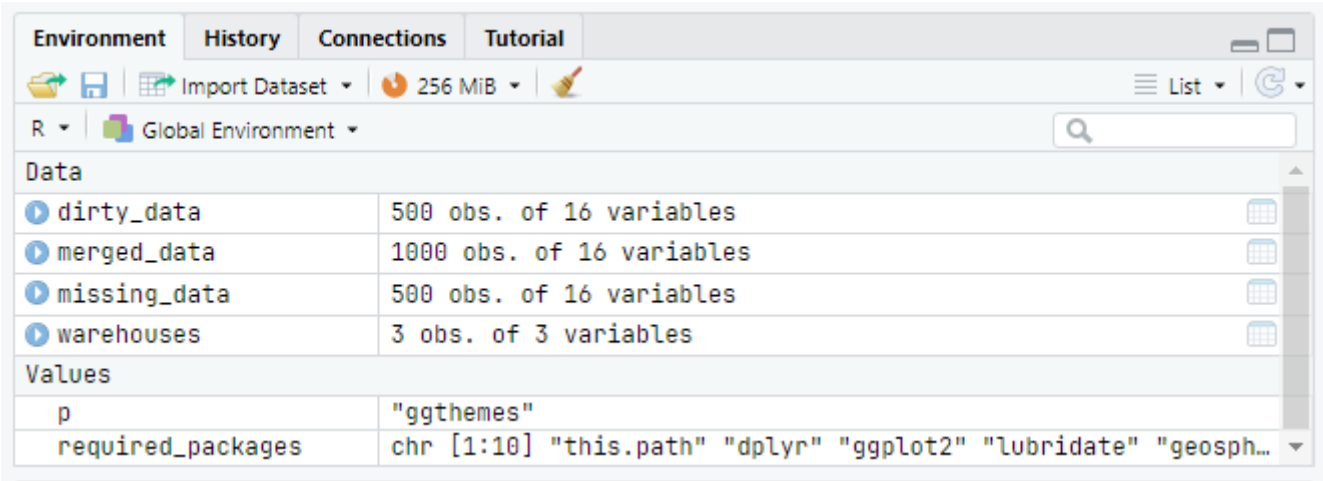
3 Tiền Xử Lý Dữ Liệu

3.1 Ghép dữ liệu

Trước hết, nhóm cần cài đặt các thư viện cần được sử dụng trong suốt quá trình làm sạch và xử lý dữ liệu.

```r
required_packages <- c("this.path", "dplyr", "ggplot2", "lubridate", "geosphere", "readr", "corrplot", "faraway", "car", "ggthemes","gt","nortest","knitr","FSA","ggcorrplot","dunn.test")
for (p in required_packages) {
  if (!require(p, character.only = TRUE)) install.packages(p)
  library(p, character.only = TRUE)
}
# Load đường dẫn hiện tại của thư mục data chứa các file CSV mẫu
setwd(this.path::here())
dirty_data <- read_csv("data/dirty_data.csv")
missing_data <- read_csv("data/missing_data.csv")
# Chuyển định dạng tháng ngày năm cột date
dirty_data$date <- parse_date_time(dirty_data$date, orders = c("mdy", "ymd", "dmy"))
missing_data$date <- parse_date_time(missing_data$date, orders = c("mdy", "ymd", "dmy"))
merged_data <- rbind(dirty_data, missing_data)
```

Listing 1: Ghép dữ liệu trong R



Hình 1: *Sau khi gộp 2 files CSV*

Giải thích: Thư viện **this.path** giúp tự động cập nhật đường dẫn thư mục gốc của dự án tại nơi chứa file R. Sau khi tải dữ liệu từ các file CSV mẫu, nhóm gộp hai file này thành một bảng duy nhất để thuận tiện xử lý.

3.2 Xác định Dữ liệu Khuyết

Trong quá trình phân tích dữ liệu mẫu, nhóm đã tiến hành kiểm tra sự tồn tại của các giá trị bị khuyết (NA) trong bộ dữ liệu. Kết quả cho thấy một số cột có chứa giá trị khuyết, cụ thể như sau:

```
> na_cout<- colSums(is.na(merged_data ))
> print(na_cout)
                    order_id                  customer_id                         date
                           0                            0                            0
           nearest_warehouse                shopping_cart                  order_price
                          10                            0                           10
            delivery_charges                 customer_lat                customer_long
                           0                           10                           10
             coupon_discount                  order_total                       season
                           0                           10                           10
       is_expedited_delivery distance_to_nearest_warehouse       latest_customer_review
                           0                           10                            0
           is_happy_customer
                          10
```

Listing 2: Các cột dữ liệu chứa giá trị bị khuyết

> Để phục vụ mục đích làm bài tập lớn và vẽ đồ thị, nhóm đã quyết định bỏ đi ba cột **distance_to_nearest_warehouse** , **customer_long** và **customer_lat** và **nearst_warehouses**.

3.3 Làm sạch dữ liệu

3.3.1 Order_price, Order_Total

Để xử lý các giá trị bị khuyết ở hai cột **order_price** và **order_total**, nhóm áp dụng công thức:

$$\textbf{order\_total} = \frac{\textbf{order\_price} \times (100 - \textbf{coupon\_discount})}{100} + \textbf{delivery\_charges} \tag{1}$$

$$\textbf{order\_price} = \frac{(\textbf{order\_total} - \textbf{delivery\_charges}) \times 100}{100 - \textbf{coupon\_discount}} \tag{2}$$

```
merged_data <- merged_data %>%
  mutate(
    order_total = ifelse(is.na(order_total), order_price * (100 - coupon_discount) / 100 +
    delivery_charges, order_total),
    order_price = ifelse(is.na(order_price), (order_total - delivery_charges) * 100 / (100 -
    coupon_discount), order_price)
  )
```

Listing 3: Xử lý cột **order_total** và **order_price**

Giải thích: Hàm **mutate** của thư viện **dplyr** được sử dụng để thêm hoặc cập nhật cột trong bảng dữ liệu.

3.3.2 season

Ở cột **season**, nhóm nhận thấy một số giá trị không đồng nhất về cách viết hoa hoặc viết thường, dẫn đến dữ liệu thiếu đồng bộ được thể hiện dưới đây:

```
season_unique_before<-unique(merged_data$season)
print(season_unique_before)
```

Listing 4: Giá trị cột **season** ban đầu

```
merged_data$season <- tolower(merged_data$season) # Đổi tất cả giá trị mùa dạng viết thường
month_value <- month(merged_data$date) # lấy tháng trong cột $date
merged_data <- merged_data %>%
  # Toán tử %>% : Truyền kết quả của phép toán hoặc hàm vào hàm tiếp theo .
  # mutate: Tạo ra cột mới hoặc thay đổi giá trị các cột trong data frame
  mutate(season = case_when(
    !is.na(season) ~ season,
    month_value %in% c(12, 1, 2) ~ "winter",
    month_value %in% c(3, 4, 5) ~ "spring",
    month_value %in% c(6, 7, 8) ~ "summer",
    TRUE ~ "autumn"
  ))
```

Listing 5: Xử lý cột **season**

Giải thích: Nhóm đã lấy giá trị **tháng** trong cột **date** của dữ liệu mẫu và sử dụng hàm **case_when** xuất ra mùa và thế vào các giá trị bị khuyết.

```
> season_unique_after<-unique(merged_data$season)
> print(season_unique_after)
[1] "winter" "summer" "autumn" "spring"
```

Listing 6: Dữ liệu cột **season** sau khi xử lý

3.3.3 is_happy_customer

```
median_happy_customer <- round(median(merged_data$is_happy_customer, na.rm = TRUE), digits =
    0)
merged_data$is_happy_customer[is.na(merged_data$is_happy_customer)] <- median_happy_customer
```

Listing 7: Làm tròn trung vị

Giải thích: Lí do nhóm sử dụng giá trị trung vị cho cột is_happy_customer để **phản ánh độ hài lòng của khách hàng**, chúng chỉ có rơi vào một trong hai trường hợp là 0 và 1 (Hài lòng hoặc không hài lòng) và không ảnh hưởng bởi các giá trị ngoại lai .

3.3.4 Kiểm tra dữ liệu sau khi làm sạch

Sau khi dữ liệu được làm sạch, kiểm tra có cột nào còn dính giá trị bị khuyết nữa không.

```
> na_cout<- colSums(is.na(merged_data ))
> print(na_cout)
                      order_id                    customer_id                           date
                             0                              0                              0
             nearest_warehouse                  shopping_cart                    order_price
                            10                              0                              0
              delivery_charges                   customer_lat                  customer_long
                             0                             10                             10
               coupon_discount                    order_total                         season
                             0                              0                              0
         is_expedited_delivery distance_to_nearest_warehouse         latest_customer_review
                             0                             10                              0
             is_happy_customer
                             0
>
```

Listing 8: Kiểm tra dữ liệu bị khuyết sau khi làm sạch

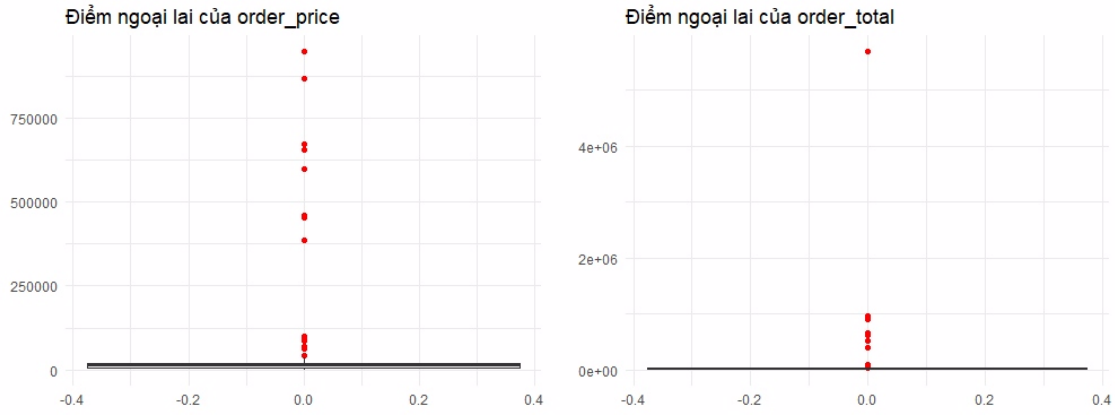
4 Mô Tả Thống Kê

4.1 Xử lý các điểm Outlier

Sau khi xử lý dữ liệu gốc, nhóm phát hiện rằng một số cột trong bộ dữ liệu xuất hiện những điểm bất thường, dẫn đến sai lệch nghiêm trọng. Điều này có thể bắt nguồn từ lỗi trong bộ dữ liệu gốc hoặc phát sinh trong quá trình xử lý. Những giá trị này được nhận diện là outliers. Để xác định các cột chứa giá trị ngoại lai, nhóm sử dụng *Boxplot* để trực quan hóa dữ liệu.

```
ggplot(data = merged_data, aes(y = order_price)) +
  geom_boxplot(outlier.shape = 16, outlier.colour = "red", outlier.fill = "red") +
  theme_minimal() +
  labs(
    title = "Điểm ngoại lai của order_price",
    y = ""
  )
ggplot(data = merged_data, aes(y = order_total)) +
  geom_boxplot(outlier.shape = 16, outlier.colour = "red", outlier.fill = "red") +
  theme_minimal() +
  labs(
    title = "Điểm ngoại lai của order_total",
    y = ""
  )
```

Listing 9: Phát hiện giá trị outlier



Hình 2: *Điểm ngoại lai được đánh dấu đỏ*

Qua phân tích, các cột được phát hiện có chứa outliers bao gồm:

- `order_price`
- `order_total`

Để loại bỏ điểm ngoại lai này, nhóm áp dụng phương pháp tính Độ trải giữa (*Interquartile Range*, viết tắt là IQR). Cụ thể, nhóm đặt cận trên *Whisker Upper* và *cận dưới* theo công thức sau:

- **Cận trên**

$$(Q3 + 1.5) \times \text{IQR (độ trải giữa)}$$

- **Cận dưới**

$$(Q1 - 1.5) \times \text{IQR (độ trải giữa)}$$

Những giá trị vượt qua cận trên hoặc dưới sẽ được thay thế bằng chính giá trị tương ứng của những cận này.

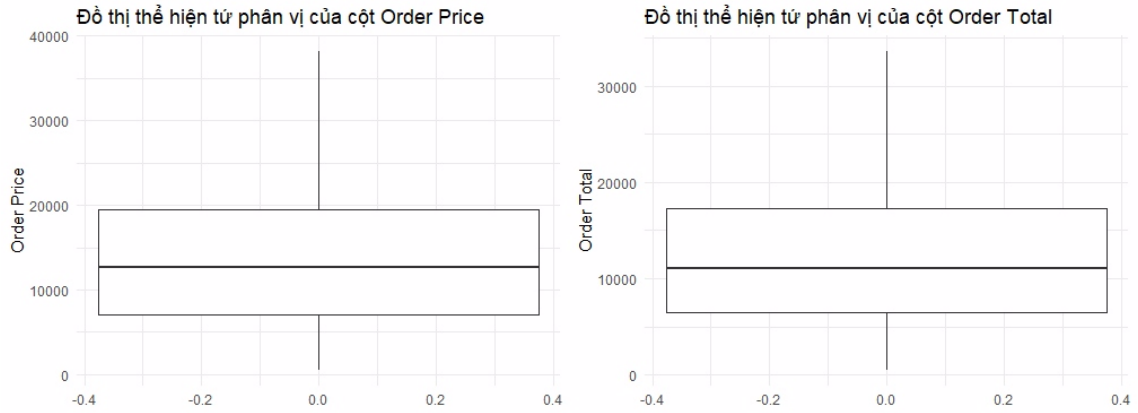
```r
# Tính toán các tứ phân vị cho các cột order_price và order_total
quantiles_price <- quantile(merged_data$order_price)
quantiles_total <- quantile(merged_data$order_total)
# Trong R , hàm quanlite trả về 5 giá trị cùng [index]
# [1]: 0%
# [2]: 25% [Q1]
# [3]: 50%
# [4]: 75% [Q3]
# [5]: 100%
# Hàm tứ phân vị trong R
q1_price <- quantiles_price[2]
q3_price <- quantiles_price[4]
q1_total <- quantiles_total[2]
q3_total <- quantiles_total[4]
# Tính IQR cho order_price và order_total
IQR_price <- q3_price - q1_price
IQR_total <- q3_total - q1_total

# Hàm tính giá trị biên dưới (lower) và biên trên (upper) của IQR
calc_lower <- function(Q1, IQR) { return(Q1 - 1.5 * IQR) }
calc_upper <- function(Q3, IQR) { return(Q3 + 1.5 * IQR) }

# Tính giá trị cận dưới và cận trên cho order_price và order_total
lower_price <- calc_lower(q1_price, IQR_price)
upper_price <- calc_upper(q3_price, IQR_price)
lower_total <- calc_lower(q1_total, IQR_total)
upper_total <- calc_upper(q3_total, IQR_total)
# Điều chỉnh giá trị order_total ra ngoài phạm vi IQR
for (i in 1:length(merged_data$order_total)) {
  Tạo một vòng lặp cho cột order_total
  if (merged_data$order_total[i] > upper_total) {
    merged_data$order_total[i] = upper_total  # Thay giá trị cận trên của order_total
  } else if(merged_data$order_total[i] < lower_total ){
    merged_data$order_total[i] = lower_total  # Thay giá trị cận dưới của order_total
  }
}
# Điều chỉnh giá trị order_price ra ngoài phạm vi IQR
## Tạo một vòng lặp cho cột order_price
for (i in 1:length(merged_data$order_price)) {
  if (merged_data$order_price[i] > upper_price) {
    merged_data$order_price[i] = upper_price  # Thay giá trị cận trên của order_price
```

```
  } else if(merged_data$order_price[i] < lower_price){
    merged_data$order_price[i] = lower_price  # Thay giá trị dưới của order_price
  }
}
```

Listing 10: Xử lý giá trị ngoại lai bằng phương pháp IQR



Hình 3: *Sau khi dùng tứ phân vị*

4.2 Thống kê dữ liệu

Mục tiêu của nhóm là hiểu rõ hơn về chi phí và doanh thu trong từng mùa, từ đó đưa ra các quyết định hợp lý như thay đổi chiến lược giao hàng hoặc cải thiện hiệu quả kinh doanh trong các mùa. Hướng đến mục tiêu chung là cung cấp dịch vụ tốt nhất cho khách hàng theo từng thời điểm trong năm, nhóm đã thực hiện các phân tích thống kê về những hạng mục quan trọng bao gồm:

- Kiểm tra sự tương quan giữa order_price và order_total
- Tổng số đơn hàng theo từng mùa.
- Tổng chi phí giao hàng.

```
season_summary <- merged_data %>%
  group_by(season) %>%
  summarise(
    total_orders = n(),
    avg_order_total = mean(order_total, na.rm = TRUE),
    total_delivery_charges = sum(delivery_charges, na.rm = TRUE)
  )
print(season_summary)
```

Listing 11: Thống kê

```
> print(season_summary)
  season total_orders avg_order_total total_delivery_charges
  <chr>         <int>           <dbl>                  <dbl>
```

```
4 1 autumn          247          12992.              17277.
5 2 spring          266          12320.              23526.
6 3 summer          249          12487.              19892.
7 4 winter          238          12009.              16475.
```

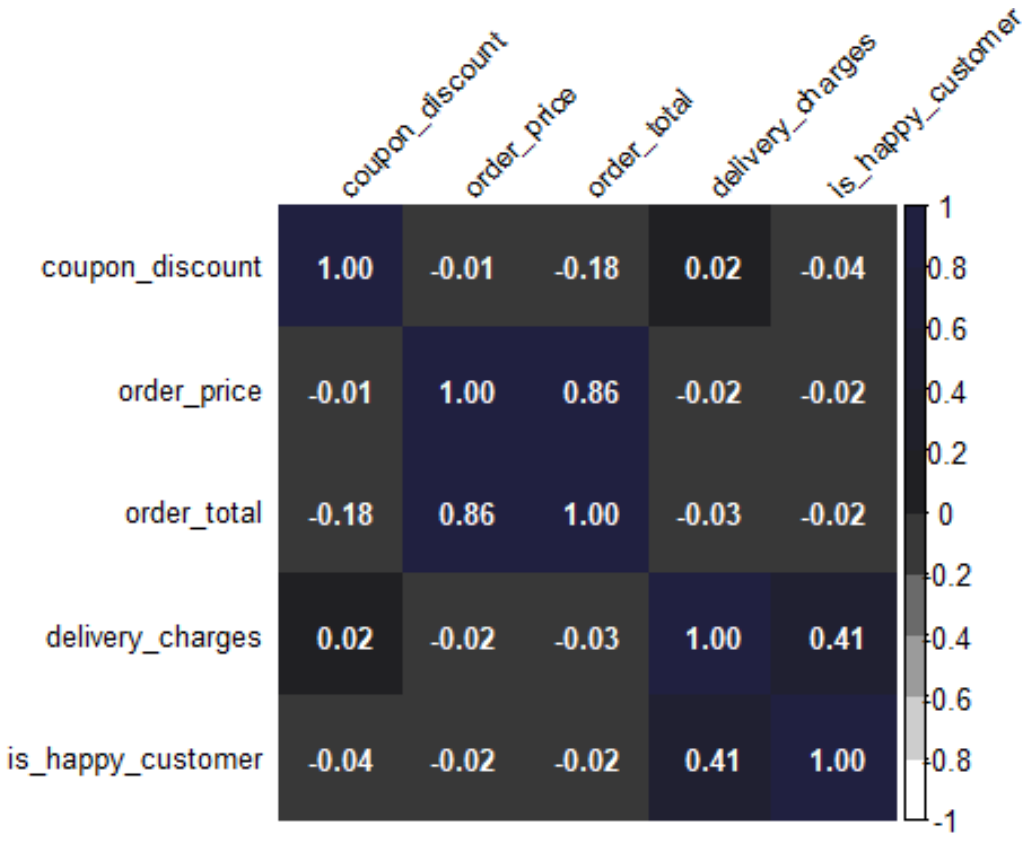
Listing 12: Thông số từng hạng mục theo mùa

4.2.1 Ma trận tương quan giữa order_price và total_order

Trước khi vẽ đồ thị, nhóm muốn kiểm tra liệu có mối quan hệ giữa order_total và order_price hay không? Nhóm chọn phương pháp ma trận **cor** để thể hiện độ tương quan

```
1 overview_data <- merged_data[c("order_price", "delivery_charges", "coupon_discount","order_
      total", "is_expedited_delivery", "is_happy_customer", "shopping_cart")]
2 numeric_data <- overview_data[sapply(overview_data, is.numeric)]
3 cor_matrix <- cor(numeric_data)
4 # Hiện thi ma trận
5 corrplot(
6   cor_matrix,
7   method = "color",
8   col= colorRampPalette(c("white","#202020", "#202040"))(10) ,
9   type = "full",
10  order = "hclust",
11  tl.col = "black",
12  tl.srt = 45,
13  addCoef.col = "white",
14  number.cex = 0.8,
15  tl.cex = 0.8,
16  diag = TRUE,
17  cl.pos = "r"
18 )
```

Listing 13: Code kiểm tra độ tương quan



Hình 4: *Biểu đồ colorgram*

Nhận xét: Có sự tương quan giữa order_total và order_price.

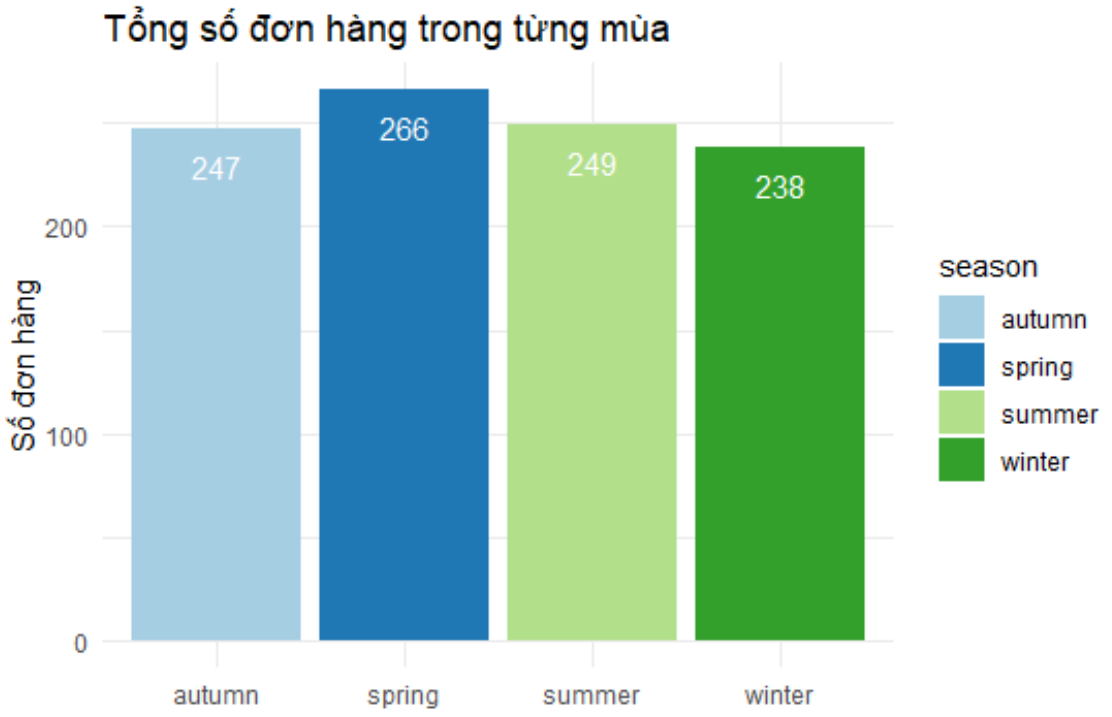
4.2.2 Tổng đơn hàng theo mùa

```
ggplot(data = season_summary, aes(x = season, y = total_orders, fill = season)) +
  geom_bar(stat = "identity") +
  geom_text(aes(label = total_orders), vjust = 2,color = "white", ) +
  theme_minimal() +
  labs(
    title = "Tổng số đơn hàng trong từng mùa",
    x = " ",
    y = " " # Để trống
  ) +
  scale_fill_brewer(palette = "Paired")
```

Listing 14: Đồ thị thể hiện tổng số đơn hàng theo từng mùa

Nhận xét: Vào mùa xuân (spring), khách hàng thường mua sắm nhiều hơn nên tổng số đơn hàng tập trung nhiều vào mùa xuân. Nhưng sự chênh lệch không đáng kể.

4.2.3 Tổng phí giao hàng theo từng mùa



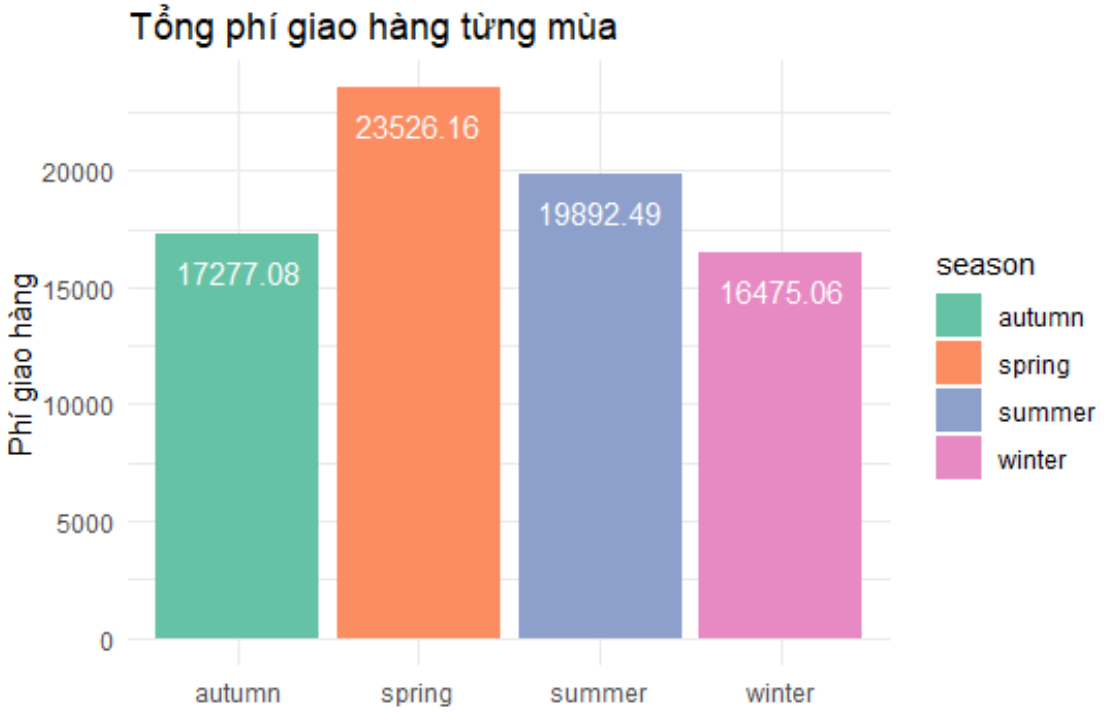
Hình 5: *Tổng số đơn hàng theo từng mùa*

```r
ggplot(data = season_summary, aes(x = season, y = total_delivery_charges, fill = season)) +
  geom_bar(stat = "identity") +
  geom_text(aes(label = total_delivery_charges), vjust = 2,color = "white", ) +
  theme_minimal() +
  labs(
    title = "Tổng phí giao hàng từng mùa",
    x = " ",
    y = " " #Để trống
  ) +
  scale_fill_brewer(palette = "Set2")
```

Listing 15: Đồ thị Tổng chi phí giao hàng theo từng mùa

Nhận xét: Lượng chi phí đơn hàng đạt giá trị cao nhất vào mùa xuân và chúng nó độ chênh lệch cao giữa các mùa.

Tổng kết: Việc phân tích các chỉ số trên sẽ giúp nhóm có cái nhìn tổng quan và đưa ra các chiến lược phù hợp nhằm tối ưu hóa các yếu tố kinh doanh theo từng mùa, đáp ứng nhu cầu của khách hàng một cách tối đa.



Hình 6: *Tổng phí đơn hàng theo từng mùa*

5 Thống Kê Suy Diễn

Kiểm tra phân phối chuẩn

Trước khi lựa chọn các phương pháp thống kê, nhóm phải kiểm tra dữ liệu **season**, price_total, order_total. có tuân theo bảng phân phối chuẩn hay không với phương pháp **shapiro**.

```r
# Trích xuất các cột order_price và order_total từ merged_data
get_order_price <- merged_data$order_price
get_order_total <- merged_data$order_total

# Kiểm tra phân phối chuẩn với Shapiro-Wilk cho cột order_total
shapiro_order_total <- shapiro.test(get_order_total)
# Kiểm tra phân phối chuẩn với Shapiro-Wilk cho cột order_price
shapiro_order_price <- shapiro.test(get_order_price)

# In kết quả kiểm tra Shapiro-Wilk cho toàn bộ dữ liệu
print(shapiro_order_price)  # Kết quả kiểm tra cho order_price
print(shapiro_order_total)  # Kết quả kiểm tra cho order_total
# Kiểm tra phân phối chuẩn theo từng mùa
seasons <- c("spring", "summer", "fall", "winter")
# Tạo danh sách để lưu giá trị order_price theo từng mùa
order_price_by_season <- list()
# Lặp qua từng mùa, trích xuất dữ liệu và lưu vào danh sách
for (season in seasons) {
  # Trích xuất dữ liệu của từng mùa bằng subset()
  season_data <- subset(merged_data, season == season)
  # Lưu giá trị order_price của từng mùa vào danh sách
  order_price_by_season[[season]] <- season_data$order_price
}
# Tạo bảng tóm tắt kết quả kiểm tra Shapiro-Wilk cho từng mùa
shapiro_summary <- data.frame(
  Season = names(order_price_by_season),  # Tên các mùa
  P_value = sapply(order_price_by_season, function(order_price) {
    # Tính p-value từ kiểm tra Shapiro-Wilk cho từng mùa
    shapiro.test(order_price)$p.value
  })
)
# Đánh giá phân phối chuẩn: nếu p-value > 0.05 thì phân phối chuẩn
shapiro_summary$Normal_Distribution <- ifelse(shapiro_summary$P_value > 0.05, "True",  # Phân phối chuẩn
"False") # Không phải phân phối chuẩn
# In bảng tóm tắt kết quả kiểm tra Shapiro-Wilk theo mùa
print(shapiro_summary)
```

Listing 16: Shapiro Test

Mùa	P-value	Có phải phân phối chuẩn?
spring	7.348016e-18	Không
summer	7.348016e-18	Không
fall	7.348016e-18	Không
winter	7.348016e-18	Không

Bảng 1: Kiểm tra phân phối chuẩn của mùa

```
> print(shapiro_order_price)  # Kết quả kiểm tra cho order_price
  Shapiro-Wilk normality test
data:  get_order_price
W = 0.95034, p-value < 2.2e-16
> print(shapiro_order_total)  # Kết quả kiểm tra cho order_total
  Shapiro-Wilk normality test
data:  get_order_total
W = 0.94557, p-value < 2.2e-16

% Giới thiệu
```

Listing 17: Hai cột còn lại

Nhận xét: Từ những dữ kiện trên, ta có thể kết luận chúng không tuân theo phân phối chuẩn.

5.1 Kiểm định trung bình một mẫu

5.1.1 Bài toán thực tế

Sử dụng t-test , nhóm muốn kiểm tra giả thuyết rằng doanh thu bán hàng của cửa hàng trong mùa Xuân có bằng với giá trị kì vọng ($\mu = 50$) không? Nhóm đưa ra hai giả thuyết như sau: **Giả thuyết kiểm định:**

- H_0: $\mu = 50$ (Doanh thu trung bình mùa xuân bằng kỳ vọng)
- H_1: $\mu > 50$ (Doanh thu trung bình mùa xuân lớn hơn kỳ vọng)

Giải thích: Lí do nhóm sử dụng **t-test** vì mẫu lớn hơn 30 mẫu, chúng không tuân theo phân phối chuẩn.

5.1.2 Tính giá trị kiểm định

5.1.3 Mô tả dữ liệu và công thức tính

```
spring_data <- merged_data %>%
  filter(season == "spring")
## Xuất ra trung bình mẫu, sd , muy
```

```r
spring_stats <- spring_data %>%
  summarise(
    mean = mean(order_total),
    sd = sd(order_total),
    n = n()
  )
```

Listing 18: Tính các thông số

Thông số	**Giá trị**
Trung bình mẫu $\bar{X}$	12,320
Giá trị kỳ vọng μ_0	50
Độ lệch chuẩn mẫu s	7,757
Số quan sát n	266 (vì df = 265)

Bảng 2: Các thông số thống kê dùng trong bài toán kiểm định một mẫu

Áp dụng lý thuyết t-test cho một mẫu, ta có công thức tính giá trị kiểm định như sau:

$$Z_{qs} = \frac{\bar{X} - \mu_0}{\frac{s}{\sqrt{n}}}$$

Thay số vào:

$$Z = \frac{12320 - 50}{\frac{7757}{\sqrt{266}}} = \frac{12270}{474.89} \approx 25.83$$

5.1.4 Xử lý trên phần mềm RStudio

Xử lý trên phần mềm RStudio, nhóm sử dụng hàm `z.test` để tính giá trị kiểm định và p-value. Cụ thể, nhóm sử dụng hàm này với các tham số như sau:

```r
z.test(x = spring_data$order_total,
       mu = 50,
       sigma.x = sd(spring_data$order_total),
       alternative = "greater")
```

Listing 19: Kiểm định một mẫu

5.1.5 Kết luận

```
    One-sample z-Test
data:  spring_data$order_price
z = 28.704, p-value < 2.2e-16
alternative hypothesis: true mean is greater than 50
```

```
95 percent confidence interval:
 12920.34       NA
sample estimates:
mean of x
 13702.69
```

Listing 20: Kết quả sau khi dùng t-test

Kết luận: p-value < 0.05, như vậy ta có thể khẳng định , không đủ điều kiện để bác bỏ H_0

5.2 Kiểm định trung bình hai mẫu

5.2.1 Bài toán thực tế

Sau khi đã xác định được doanh thu của cửa hàng tại mùa xuân không vượt qua giá trị kì vọng, nhóm tiếp tục đem so sánh dữ liệu **mùa xuân** so với **mùa hè** với mức ý nghĩa $\alpha = 0.05$

Nhóm đặt giả thuyết như sau

- **Giả thuyết không (H_0):** $\mu_1 = \mu_2$ (Doanh thu trung bình mùa xuân bằng mùa hè)
- **Giả thuyết đối (H_1):** $\mu_1 \neq \mu_2$ (Có sự khác biệt giữa 2 mùa)

5.2.2 Tính giá trị kiểm định

Để tính giá trị kiểm định, nhóm sử dụng phương pháp tách 2 mẫu **xuân** và **hè** ra làm 2 và lọc ra thông số độ lệch chuẩn, n (số mẫu) và mean (trung bình mẫu)

```

\begin{lstlisting}[language=R, caption= z-test]
group_stats <- merged_data %>%
  group_by(season) %>%
  summarise(
    mean = mean(order_total),
    sd = sd(order_total),
    n = n(),
    .groups = "drop"  # tránh cảnh báo nhóm
  ) %>%
  filter(season %in% c("spring", "summer"))

print(group_stats)

# Độ lệch chuẩn 2 mùa xuân và hè
sd_spring <- group_stats$sd[group_stats$season == "spring"]
sd_summer <- group_stats$sd[group_stats$season == "summer"]

## Tổng giá trị đơn hàng trong 2 mùa
```

```r
spring_data <- merged_data$order_total[merged_data$season == "spring"]
summer_data <- merged_data$order_total[merged_data$season == "summer"]
### Z -test
z_test_result <- z.test(x = spring_data, y = summer_data,
       mu = 0,
       sigma.x = sd_spring,
       sigma.y = sd_summer,
       alternative = "two.sided")

print(z_test_result)
```

Listing 21: Các thông số thống kê dùng trong kiểm định trung bình 2 mẫu

5.2.3 Mô tả dữ liệu và công thức tính

Bảng 3: Chỉ số bài toán `order_total` theo mùa Xuân và Hè

Mùa	**Trung bình (μ)**	**Độ lệch chuẩn (s)**	**Số lượng (n)**
Xuân (Spring)	12,320	7,757	266
Hè (Summer)	12,487	7,697	249

Trong phần lý thuyết, nhóm đã đề cập phương pháp t-test cho hai mẫu. Do đó nhóm sẽ sử dụng công thức tính giá trị kiểm định và thông số trong bảng mô tả dữ liệu như sau:

$$Z = \frac{12319.55 - 12486.75}{\sqrt{\frac{7757.41^2}{266} + \frac{7697.42^2}{249}}} \approx -0.24542$$

5.2.4 Xử lý trên phần mềm RStudio

Xử lý trên phần mềm RStudio, nhóm sử dụng hàm `t.test` để tính giá trị kiểm định và p-value. Cụ thể, nhóm sử dụng hàm này với các tham số như sau:

```r
  t_test_result_two_sample<-t.test(order_total ~ season,
       data = merged_data %>% filter(season %in% c("spring", "summer")),
       var.equal = FALSE) # giả định phương sai không bằng nhau
print(t_test_result_two_sample)
```

Listing 22: Các thông số thống kê dùng trong kiểm định trung bình 2 mẫu

```
    Welch Two Sample t-test

data:  order_total by season
t = -0.24542, df = 511.26, p-value = 0.8062
alternative hypothesis: true difference in means between group spring and group summer is not
    equal to 0
95 percent confidence interval:
```

```
 -1505.705  1171.298
sample estimates:
mean in group spring mean in group summer
             12319.55             12486.75
```

Listing 23: Kết quả t-test

5.2.5 Kết luận

Với $p_value = 0.8602 > 0.05$, nhóm không có đủ bằng chứng để bác bỏ giả thuyết H_0 tức là không có sự khác biệt trong doanh thu giữa hai mùa.

5.3 Annova một yếu tố

5.3.1 Bài toán thực tế

Một cửa hàng điện tử muốn kiểm tra xem **giá trị đơn hàng trung bình** (`order_total`), với cỡ mẫu của mỗi nhóm là 30, có bị ảnh hưởng bởi **mùa vụ** (`season`) hay không. Dữ liệu được thu thập từ các đơn hàng, phân loại thành 4 nhóm theo mùa: **Winter** (mùa đông), **Summer** (mùa hè), **Autumn** (mùa thu) và **Spring** (mùa xuân).

Với mức ý nghĩa $\alpha = 5\%$, sử dụng phương ph **mô hình ANOVA một yếu tố** để kiểm tra xem giá trị đơn hàng trung bình giữa các mùa có sự khác biệt đáng kể hay không.

5.3.2 Giả thuyết kiểm định

- Các tổng thể phải tuân theo phân phối chuẩn $N(\mu, \sigma^2)$
- Số nhóm $k = 4$
- Phương sai bằng nhau $\sigma_1^2 = \sigma_2^2 = \sigma_3^2 = \sigma_4^2$
- Các mẫu quan sát (từ k tổng thể) được lấy độc lập, với $n = 30$ cho mỗi nhóm, tổng số quan sát $N = 120$.

5.3.3 Hiện thực phép toán

Gọi μ_i là giá trị đơn hàng trung bình của mùa thứ i.

$$H_0 : \mu_1 = \mu_2 = \mu_3 = \mu_4,$$
$$H_1 : \exists\ i \neq j \text{ sao cho } \mu_i \neq \mu_j.$$

5.3.4 Tính các tham số mẫu

Bảng 4: Trung bình và phương sai mẫu theo mùa

Mùa	Trung bình mẫu $\overline{x}_i$	Phương sai mẫu s_i^2
Winter	12499.31	56, 686, 589.91
Summer	12493.37	61, 622, 987.35
Autumn	12858.15	48, 321, 908.37
Spring	14594.52	106, 019, 827.40

5.3.5 Giá trị trong và giữa các nhóm

Tổng bình phương giữa nhóm (SSB):

$$SSB = 14,303,293.20 + 14,550,442.35 + 3,299,751.675 + 55,465,154.35 = 87,618,641.575$$

Bình phương trung bình giữa nhóm (MSB):

$$MSB = \frac{SSB}{k-1} = \frac{87,618,641.575}{3} \approx 29,206,213.86$$

Tổng bình phương trong nhóm (SSW):

$$SSW = 1,643,911,107 + 1,787,066,635 + 1,401,335,343 + 3,074,574,995 = 7,906,888,080$$

Bình phương trung bình trong nhóm (MSW):

$$MSW = \frac{SSW}{N-k} = \frac{7,906,888,080}{116} \approx 68,162,828.28$$

5.3.6 Giá trị thống kê kiểm định

$$F = \frac{MSB}{MSW} = \frac{29,206,213.86}{68,162,828.28} \approx 0.4285$$

5.3.7 Miền bác bỏ

Với $\alpha = 5\%$, $df_1 = k - 1 = 3$, $df_2 = N - k = 116$:

$$F_{0.05;3;116} \approx 2.6802$$

Miền bác bỏ:

$$RR = (2.6802, +\infty)$$

5.3.8 Kết luận

Kết luận: Vì $F \approx 0.4285 < 2.6802$, không bác bỏ H_0 nên không có bằng chứng thống kê cho thấy giá trị đơn hàng trung bình giữa các mùa khác biệt đáng kể.

5.4 Xử lý trên phần mềm R

Sử dụng phương pháp leveneTest trong R để thực hiện

```
##################################################
# 9. ANOVA (4 mùa)
##################################################
df_cleaned <- merged_data %>%
  group_by(season) %>%
  mutate(order_total_cleaned = ifelse(order_total > quantile(order_total, 0.75) + 1.5 * IQR(order_total) |
                                      order_total < quantile(order_total, 0.25) - 1.5 * IQR(order_total),
                                      NA, order_total)) %>%
  ungroup()

leveneTest(order_total_cleaned ~ season, data = df_cleaned)

anova_model <- aov(order_total_cleaned ~ season, data = df_cleaned)
summary(anova_model)

# Boxplot ANOVA
ggplot(df_cleaned, aes(x = season, y = order_total_cleaned)) +
  geom_boxplot() +
  theme_minimal() +
  labs(title = "So sánh Order Total giữa các mùa (ANOVA)")
```

Listing 24: Phép kiểm ANOVA một yếu tố

```
Df    Sum Sq   Mean Sq F value Pr(>F)
season        3 3.576e+08 119214928   2.299 0.0759 .
Residuals   981 5.086e+10  51848700
```

Listing 25: Kết quả ANOVA một yếu tố

Kết luận: Không đủ điều kiện để bác bỏ giả thuyết H_0 với p-value $= 0.0759 > 0.05$, tức là không có sự khác biệt đáng kể giữa giá trị đơn hàng trung bình của các mùa.

5.5 Hồi quy tuyến tính

5.5.1 Bài toán thực tế

Xây dựng mô hình hồi quy tuyến tính với biến phụ thuộc:

$$Y = \texttt{order\_price}$$

và hai biến độc lập:

$$X_1 = \texttt{coupon\_discounts}, \quad X_2 = \texttt{delivery\_charges}.$$

Mô hình hồi quy bội được viết dưới dạng:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \varepsilon_i, \quad i = 1, 2, \ldots, n$$

trong đó:

- Y: Giá trị đơn hàng (`order_price`)
- X_1: Giảm giá bằng coupon (`coupon_discounts`)
- X_2: Phí giao hàng (`delivery_charges`)
- ε_i: Sai số ngẫu nhiên, có kỳ vọng $E(\varepsilon) = 0$, phương sai σ^2

5.5.2 Cơ sở lý thuyết

Mô hình hồi quy bội mở rộng mô hình hồi quy đơn bằng cách đưa vào nhiều biến độc lập:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p + \varepsilon$$

- β_0: Hệ số chặn
- β_j: Hệ số hồi quy của biến độc lập X_j
- ε: Sai số ngẫu nhiên

5.5.3 Phương pháp ước lượng

Sử dụng phương pháp **Bình phương tối thiểu (OLS)**:

$$\hat{\beta} = (X^\mathsf{T} X)^{-1} X^\mathsf{T} Y$$

trong đó:

- X: Ma trận giá trị các biến độc lập $(n \times (p+1))$
- Y: Vector giá trị của biến phụ thuộc $(n \times 1)$
- $\hat{\beta}$: Vector hệ số hồi quy $(p+1) \times 1$

Hệ số xác định R^2:

$$R^2 = 1 - \frac{SS_{\text{res}}}{SS_{\text{tot}}}$$

với:

$$SS_{\text{res}} = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2, \quad SS_{\text{tot}} = \sum_{i=1}^{n}(y_i - \overline{y})^2$$

5.5.4 Kết quả mô hình OLS

Mô hình trong R:

```
model <- lm(order_price ~ coupon_discount + delivery_charges, data = df)
summary(model)
```

Kiểm tra đa cộng tuyến:

```
vif(model)
# coupon_discount: 1.000546
# delivery_charges: 1.000546
```

$\Rightarrow$ VIF xấp xỉ 1 cho cả hai biến, không xảy ra đa cộng tuyến.

Hệ số ước lượng:

$$\hat{\beta}_0 = 13297.85, \quad \hat{\beta}_1 = 143.66, \quad \hat{\beta}_2 = 59.37$$

Ý nghĩa:

- Khi X_1 tăng 1 đơn vị, Y tăng trung bình 143.66
- Khi X_2 tăng 1 đơn vị, Y tăng trung bình 59.37

Kiểm định t:

- X_1: $p = 0.527 > 0.05 \Rightarrow$ không đủ bằng chứng biến có ý nghĩa
- X_2: $p = 0.662 > 0.05 \Rightarrow$ không đủ bằng chứng biến có ý nghĩa

Kiểm định F: p-value lớn hơn 0.05, $R^2 \approx 0 \Rightarrow$ mô hình không có ý nghĩa thống kê, biến phụ thuộc quan hệ yếu với các biến độc lập.

6 Thảo luận và Mở rộng

6.1 Nhận xét dữ liệu

- Trong mô hình tổng quan, có những cột dữ liệu không tuân theo phân phối chuẩn như **order_total** và **order_price**. Điều này có thể do sự hiện diện của các giá trị ngoại lai hoặc phân phối không đồng nhất.
- Phương sai không đồng nhất , có thể do sự biến động lớn trong giá trị đơn hàng giữa các mùa.
- Có những cột phụ thuộc với nhau, ví dụ **order_price** ảnh hưởng bởi **coupon_discount**.

6.2 Tính bao quát

Nhóm đã phân tích các yếu tố ảnh những ảnh hưởng đến sự thay đổi của dữ liệu, cũng như phân tích và kiểm định trong từng trường hợp như sau: Các kiểm định được chọn phù hợp với mục tiêu và đặc điểm dữ liệu (không chuẩn, không đồng nhất phương sai, có cặp). Ngoài ra, nhóm cũng mở rộng bằng kiểm định hậu nghiệm (Dunn Test) để phân tích sâu hơn sau Kruskal-Wallis. Do đó, phân tích được đánh giá là có tính bao quát tốt và hợp lý về mặt phương pháp. Cụ thể các phương pháp đã được nhóm sử dụng như sau:

- Phân tích 1 mẫu với **z-test**
- Phân tích 2 mẫu với **t-test**
- Phân tích Annova **b**ằng Krustal-Wallis
- Phân tích hồi quy tuyến tính

6.3 Điểm mạnh và điểm yếu của các phương pháp đã sử dụng

Bảng 5: Tổng hợp ưu điểm và nhược điểm của các phương pháp kiểm định

Phương pháp	Ưu điểm	Nhược điểm
Z-test một mẫu	• Dễ hiểu và triển khai. • Hiệu quả với cỡ mẫu lớn ($n > 30$). • Không yêu cầu phân phối chuẩn nhờ định lý giới hạn trung tâm.	• Giả định dữ liệu độc lập, mẫu đại diện. • Không phù hợp với cỡ mẫu nhỏ.
T-test hai mẫu (Welch)	• So sánh hai nhóm độc lập. • Không yêu cầu phương sai bằng nhau.	• Nhạy cảm với ngoại lai nếu dữ liệu không chuẩn. • Cần số lượng mẫu đủ lớn nếu phân phối không chuẩn.
Anova 1 yếu tố	• Dễ hiểu và thực hiện • Không yêu cầu phương sai đồng nhất.	• Yêu cầu tổng thể có phân phối chuẩn. • Không chỉ ra cặp nhóm nào khác biệt. • Cần kiểm định hậu nghiệm (như Dunn Test).
Phương pháp OLS trong hồi quy tuyến tính	• Dễ hiểu, tính toán nhanh, nhiều công cụ kiểm định, nền tảng lý thuyết vững	• Nhạy cảm với ngoại lai, cần giả định chặt chẽ, không xử lý tốt dữ liệu phi tuyến, đa cộng tuyến làm mất ổn định

6.4 Mối quan hệ giữa mục tiêu và phương pháp được chọn

Mục tiêu chính của nghiên cứu không phải là dự đoán giá trị, mà là kiểm tra xem các yếu tố như mùa vụ, đặc điểm sản phẩm hay hành vi mua hàng có tạo ra sự khác biệt có ý nghĩa thống kê về giá trị đơn hàng hay không.

Nhóm lựa chọn các phương pháp sau:

1. **Z-test một mẫu**: kiểm tra sự khác biệt giữa trung bình mẫu và giá trị giả thuyết.
2. **T-test hai mẫu (Welch)**: so sánh trung bình của hai nhóm độc lập khi phương sai không đồng nhất.
3. **ANOVA một yếu tố**: kiểm tra sự khác biệt trung bình giữa nhiều nhóm; nếu có ý nghĩa thống kê, mở rộng thêm **Dunn test** để xác định cặp nhóm khác biệt.
4. **Hồi quy tuyến tính**: phân tích tác động của biến độc lập lên biến phụ thuộc và đánh giá mức độ giải thích của mô hình.

Việc lựa chọn trên đảm bảo phù hợp với đặc điểm dữ liệu, đáp ứng mục tiêu so sánh sự khác biệt và phân tích mối quan hệ, thay vì thuần túy dự đoán.

6.5 Mở rộng: Dunn Test

Sau khi đưa ra những điểm mạnh hay giới hạn những phương pháp nhóm đã sử dụng ở trên. Nhóm quyết định sẽ mở rộng phương pháp **phân tích Annova 1 yếu tố**. Như đề cập ở trên, phương pháp kiểm định KW chỉ đưa ra được "bề nổi"hay chỉ kiểm tra sự khác biệt tổng thể giữa các nhóm.

Nhóm sẽ áp dụng phương pháp hậu nghiệm (pos-hoc) để tìm ra cụ thể nhóm nào có sự khác biệt. Nhóm đề xuất sử dụng **Dunn Test**.

```r
dunn_result <- dunn.test(merged_data$order_price, merged_data$season, method = "bonferroni",
    list = TRUE)
# Tạo bảng kết quả
table <- cbind.data.frame(
  Comparison = dunn_result$comparisons,  # Các cặp so sánh (cụ thể mùa)
  Z_value = dunn_result$Z,               # Giá trị Z
  P_adjusted = dunn_result$P.adjusted    # P-value đã điều chỉnh
)
# Sắp xếp bảng theo p-value đã điều chỉnh
table <- table[order(table$P_adjusted), ]
# Tạo bảng với gt và thêm tiêu đề
table %>%
  gt() %>%
  tab_header(
    title = md("#### Kết quả phân tích pos-hoc (Dunn's Test)"),
    subtitle = "So sánh sự khác nhau giữa các nhóm mùa bằng Bonerroni"
  )
```

Listing 26: Thực hiện phân tích Pos Hoc trong R

6.6 Kết quả

Các mùa	Z-value	P.adjusted
autumn - winter	1.4580097	0.4345136
autumn - spring	0.8823710	1.0000000
autumn - summer	0.6008488	1.0000000
spring - summer	-0.2722941	1.0000000
spring - winter	0.6104089	1.0000000
summer - winter	0.8656661	1.0000000

Bảng 6: Post-hoc Analysis Results (Dunn's Test)

Kết luận: Sau khi xuất ra kết quả Dunn Test, tổng quan các mùa đều có khác biệt gì lớn.

7 Nguồn Dữ Liệu

- Dữ liệu mẫu: https://www.kaggle.com/datasets/muhammadshahrayar/transactional-retail-dataset-of-electronics-store
- Link R của bài tập lớn: https://github.com/bnhtho/btl_xtsk_241/blob/main/2333017_Assigment.R

8 Tài liệu Tham Khảo

Tài liệu

[1] Anderson, D. R., Sweeney, D. J., & Williams, T. A. (2016). *Statistics for Business and Economics* (11th ed.). Cengage Learning Vietnam Company Limited.

[2] Peter Dalgaard. *Introductory Statistics with R* (8th ed)

[3] Nguyễn Đình Huy. *Giáo trình xác suất và thống kê(2023)*, Nxb. Đại học Quốc Gia, Thành phố Hồ Chí Minh.