BỘ CÔNG THƯƠNG

TRƯỜNG ĐẠI HỌC KINH TẾ KỸ THUẬT CÔNG NGHIỆP

KHOA KHOA HỌC ỨNG DỤNG

BÁO CÁO ĐỒ ÁN 1

(NHÓM 1_2)

SỬ DỤNG NGÔN NGỮ R ĐỂ THĂM DÒ VÀ TRỰC QUAN DỮ LIỆU VỀ GIÁ CẢ HÀNG HÓA VÀ XU HƯỚNG TIÊU DÙNG CỦA CÁC LOẠI HÀNG HÓA

HÀ NỘI – 2024

BỘ CÔNG THƯƠNG

TRƯỜNG ĐẠI HỌC KINH TẾ KỸ THUẬT CÔNG NGHIỆP

KHOA KHOA HỌC ỨNG DỤNG

BÁO CÁO ĐỒ ÁN 1

SỬ DỤNG NGÔN NGỮ R ĐỂ THĂM DÒ VÀ TRỰC QUAN DỮ LIỆU VỀ GIÁ CẢ HÀNG HÓA VÀ XU HƯỚNG TIÊU DÙNG CỦA CÁC LOẠI HÀNG HÓA

Họ và tên nhóm sinh viên:

Nhóm đăng ký đồ án: Nhóm số: 1_2 - DHKL16A2HN		
Họ và tên các thành viên nhóm	Mã Sinh viên	Điện thoại
Hồ Thị Minh Hằng	22174600024	0385641416
Trần Đức Lương	22174600051	0384014983
Nguyễn Ngọc Bắc	22174600045	0988967119
Đoàn Thị Kim Ánh	22174600098	0338527349
Nguyễn Văn Hà	22174600018	0705729569

HÀ NỘI – 2024

LỜI CẢM ƠN

Để hoàn thành báo cáo Đồ án 1 (Trực quan hóa dữ liệu bằng R) này trước hết chúng em xin gửi đến quý thầy, cô giáo trong khoa Khoa Học Ứng Dụng trường Đại học Kinh tế - Kỹ thuật Công nghiệp lời cảm ơn chân thành.

Chúng em xin gửi lời cảm ơn tới Công ty TNHH phân phối lân, đạm VADFO đã hỗ trợ và cung cấp nguồn dữ liệu hàng hóa để chúng em có thể dựa vào nguồn dữ liệu này và thực hành báo cáo.

Đặc biệt, em xin gửi lời cảm ơn sâu sắc nhất đến thầy Trần Chí Lê, người đã tận tình hướng dẫn, giúp đỡ chúng em hoàn thành bài báo cáo đồ án này. Đồ án 1 (Trực quan hóa dữ liệu bằng R) đã trang bị cho chúng em những kiến thức, kỹ năng cơ bản cần có để hoàn thành đề tài này.

Tuy nhiên, trong quá trình hoàn thiện báo cáo, do kiến thức chuyên sâu còn hạn chế nên chúng em vẫn còn nhiều thiếu sót khi tìm hiểu, đánh giá và trình bày về đề tài. Rất mong nhận được sự quan tâm, góp ý của thầy để bài báo cáo của chúng em được đầy đủ và hoàn chỉnh hơn.

Em xin chân thành cảm ơn!

TÊN ĐỒ ÁN: SỬ DỤNG NGÔN NGỮ R ĐỂ THĂM DÒ VÀ TRỰC QUAN DỮ LIỆU VỀ GIÁ CẢ HÀNG HÓA VÀ XU HƯỚNG TIÊU DÙNG CỦA CÁC LOẠI HÀNG HÓA

Yêu cầu chi tiết:

1.Giới thiệu:

Mục tiêu và tầm quan trọng của việc phân tích dữ liệu giá cả hàng hóa và xu hướng tiêu dùng của các loại hàng hóa:

- Việc phân tích dữ liệu về giá cả hàng hóa và xu hướng tiêu dùng của các loại hàng hóa đóng vai trò vô cùng quan trọng trong việc hiểu và dự đoán hành vi của thị trường. Mục tiêu của việc này là cung cấp cái nhìn sâu sắc và chi tiết về cách mà giá cả và nhu cầu tiêu dùng thay đổi theo thời gian và các yếu tố khác nhau như mùa vụ, tình trạng kinh tế và các chiến lược tiếp thị. Thông qua việc phân tích này, chúng ta có thể phát hiện ra các xu hướng mới, dự đoán biến động trong thị trường, đưa ra các quyết định kinh doanh thông minh và tối ưu hóa chiến lược bán hàng. Đồng thời, việc nắm bắt được các xu hướng tiêu dùng cũng giúp các doanh nghiệp tăng cường sự cạnh tranh, đáp ứng nhu cầu của khách hàng một cách hiệu quả và tăng cường sự thịnh vượng trong thị trường cạnh tranh ngày càng khốc liệt hiện nay.
- Ngoài ra, việc phân tích dữ liệu về giá cả hàng hóa và xu hướng tiêu dùng cũng giúp các doanh nghiệp xác định được những cơ hội mới và nguy cơ tiềm ẩn trong thị trường. Bằng cách này, họ có thể định hình chiến lược kinh doanh để tận dụng những cơ hội và đối phó với những thách thức, từ đó tạo ra lợi ích cạnh tranh và bền vững. Hơn nữa, thông qua việc hiểu rõ về nguồn gốc của sự biến động trong giá cả và nhu cầu tiêu dùng, các doanh nghiệp cũng có thể tối ưu hóa quy trình sản xuất và quản lý nguồn lực một cách hiệu quả hơn. Tóm lại, phân tích dữ liệu này không chỉ là công cụ hỗ trợ quan trọng cho quản lý kinh doanh mà còn là yếu tố quyết định trong việc định hình chiến lược và thành công của các doanh nghiệp trong môi trường kinh doanh ngày nay.

Giới thiệu tóm tắt về phương pháp phân tích thăm dò và trực quan dữ liệu:

Phương pháp phân tích thăm dò và trực quan dữ liệu về giá cả hàng hóa và xu hướng tiêu dùng của các loại hàng hóa đóng vai trò quan trọng trong việc hiểu sâu hơn về thị trường và hành vi của người tiêu dùng. Dưới đây là một tóm tắt về các phương pháp phân tích thăm dò và trực quan dữ liệu được áp dụng trong việc này:

- Thống kê mô tả: Phương pháp này bao gồm tính toán các thống kê cơ bản như trung bình, trung vị, độ lệch chuẩn, phân vị và phân phối của dữ liệu giá cả và nhu cầu tiêu dùng. Thống kê mô tả giúp tóm tắt và hiểu rõ hơn về đặc điểm cơ bản của dữ liệu.
- Biểu đồ đường: Sử dụng để theo dõi và minh họa sự biến đổi của giá cả hàng hóa và xu hướng tiêu dùng theo thời gian. Biểu đồ đường thường được sử dụng để nhận diện các xu hướng dài hạn, biến động ngắn hạn và mùa vụ.
- Biểu đồ cột: Được sử dụng để so sánh giá cả và nhu cầu tiêu dùng giữa các loại hàng hóa hoặc giữa các thời điểm khác nhau. Biểu đồ cột giúp trực quan hóa sự khác biệt và tương quan giữa các biến.
- Biểu đồ hộp: Thường được sử dụng để minh họa phân phối của dữ liệu giá cả và nhu cầu tiêu dùng, bao gồm giá trị trung bình, phân vị và các điểm dữ liệu ngoại lệ. Biểu đồ hộp giúp phát hiện và hiểu rõ hơn về sự biến động và phân bố của dữ liệu.
- Biểu đồ tương quan: Được sử dụng để phân tích mối quan hệ tương quan giữa các biến, chẳng hạn như mối quan hệ giữa giá cả và lượng tiêu thụ, giữa giá cả của các mặt hàng khác nhau, hoặc giữa giá cả và các yếu tố khác như tình trạng kinh tế.

2. Thu thập dữ liệu:

- Sử dụng nguồn dữ liệu phổ biến: Lấy từ dữ liệu Công ty TNHH phân phối lân, đạm.
- Kỹ thuật thu thập dữ liệu là: Phỏng vấn và khảo sát, lấy số liệu từ bộ phận kế toán.
- Làm sạch dữ liệu và xử lý các giá trị thiếu: sử dụng gói library(tidyr) bằng cách chuyển đổi dữ liệu thành dạng mà mỗi hàng là một quan sát và mỗi cột là một biến. Các kỹ thuật thuộc gói library(tidyr) để làm sạch dữ liệu: Gộp(Pivoting), Làm sạch dữ liệu(Data Cleaning), số lượng quan sát còn lại phụ thuộc vào Nguồn dữ liệu ban đầu, Tiêu chuẩn làm sạch, Phương pháp làm sạch dữ liệu , mục tiêu cuối cùng của phân tích. Chuyển đổi dữ liệu phù hợp cho phân tích.

3. Thống kê mô tả:

Các đại lượng thống kê cơ bản: Tần suất (Frequency), Trung bình(Mean), Trung vị(Median), Phương sai(Variance) và Độ lệch chuẩn(Standard Deviation), Phân phối(Distribution).

Biểu đồ tóm tắt dữ liệu:

- Sử dụng hàm boxplot() để vẽ boxplot biểu diễn phân phối dữ liệu.
- Sử dụng biểu đồ cột (bar plot) hoặc biểu đồ phân phối (histogram) để phân biệt các mã hàng phân phối lớn.
- Sử dụng biểu đồ dạng "scatter plot" (biểu đồ phân tán) hoặc "line plot" (biểu đồ đường) để mô tả mối quan hệ giữa biến số lượng và giá tiền của mặt hàng xuất nhập và tồn kho.
- Sử dụng biểu đồ tròn (pie chart) để so sánh ra các mặt hàng bán chạy nhất, doanh thu cao nhất.
- Sử dụng biểu đồ tăng trưởng(Growth Chart) để phân tích giá cả hàng hóa.
- Sử dụng biểu đồ tương quan(Correlation Chart) để xác định mức độ tương quan giữa giá cả của các loại hàng hóa.
- Sử dụng biểu đồ Pareto dùng để xác định các yếu tố quan trọng nhất ảnh hưởng đến xu hướng tiêu dùng của người tiêu dùng.
- Sử dụng biểu đồ thời gian(Time Series Chart)) được sử dụng để theo dõi xu hướng và biến động của một biến (ví dụ: giá cả hàng hóa, doanh số bán hàng, lợi nhuận) theo thời gian.

Phân tích phân phối dữ liệu: Để phân tích mối liên hệ giữa giá cả hàng hóa và xu hướng tiêu dùng của các loại hàng hóa ta sử dụng:

- Sử dụng ước lượng tỷ lệ để so sánh được sản phẩm bán chạy nhất.
- Sử dụng ước lượng hồi quy tuyến tính để dự đoán giá bán dựa trên số lượng bán và loại sản phẩm.
- Kiểm định t để so sánh giá bán trung bình của hai loại sản phẩm khác nhau.
- Kiểm định ANOVA**:** So sánh giá bán trung bình của nhiều loại sản phẩm khác nhau. Xác định xem có sự khác biệt đáng kể về giá bán giữa các loại sản phẩm hay không.

4. Phân tích mối liên hệ giữa các biến:

Biến nhập - xuất - tồn:

- Biến nhập (Inventory In): Đây là lượng hàng hoá hoặc dịch vụ được nhập vào hệ thống hoặc tồn kho trong một khoảng thời gian cụ thể. Biến này đại diện cho sự nhập hàng hóa hoặc dịch vụ vào doanh nghiệp.

- Biến xuất (Inventory Out): Đây là lượng hàng hoá hoặc dịch vụ được xuất ra khỏi hệ thống hoặc tồn kho trong một khoảng thời gian cụ thể. Biến này thường đại diện cho doanh số bán hàng hoặc dịch vụ.
- Biến tồn (Inventory On-hand): Đây là lượng hàng hoá hoặc dịch vụ vẫn còn lại trong hệ thống hoặc tồn kho sau khi đã trừ đi lượng hàng hoá đã bán hoặc sử dụng. Biến này cho biết mức độ tồn kho hiện tại.
- Nếu có một mẫu hay mối quan hệ tuyến tính giữa hai biến, bạn sẽ thấy các điểm trên biểu đồ scatter plot sắp xếp thành một hình dạng gần như thẳng hàng.

Mối liên hệ giữa các biến xuất, nhập, tồn được phân tích như sau:

- Tương quan dương giữa biến nhập và biến tồn, tương quan dương giữa biến xuất và biến tồn, tương quan ngược giữa biến nhập và biến xuất, mối quan hệ không tuyến tính hoặc không đồng đều. Ta có thể phân tích kĩ hơn về mối liên hệ giữa các biến này cung cấp thông tin quan trọng để hiểu và cải thiện quản lý tồn kho, dự báo nhu cầu thị trường và tối ưu hóa hoạt động kinh doanh.

Ma trận tương quan:

- Ma trận tương quan là một bảng chứa tất cả các hệ số tương quan giữa các cặp biến. Trong trường hợp này, bạn có thể xây dựng ma trận tương quan cho các biến nhập, xuất và tồn.
- Mỗi ô trong ma trận sẽ chứa hệ số tương quan giữa hai biến. Hệ số tương quan có thể nằm trong khoảng từ -1 đến 1, với -1 cho biết mối quan hệ nghịch biến hoàn toàn, 1 cho biết mối quan hệ đồng biến hoàn toàn và 0 cho biết không có mối quan hệ tuyến tính nào.
- Tương quan giữa biến nhập và biến tồn: Nếu lượng hàng nhập vào tăng, có thể dẫn đến tăng lượng.

Hệ số tương quan và độ đồng biến:

- Hệ số tương quan giữa các biến cung cấp thông tin về mức độ mối quan hệ tuyến tính giữa chúng. Nếu hệ số tương quan gần 1 hoặc bằng -1, có nghĩa là mối quan hệ giữa các biến là mạnh mẽ
- Độ đồng biến cho biết hướng của mối quan hệ. Nếu độ đồng biến dương, khi một biến tăng, biến kia cũng tăng; nếu độ đồng biến âm, khi một biến tăng, biến kia giảm.

Biểu đồ Scatter Plot:

- Biểu đồ scatter plot là một biểu đồ hai chiều trong đó mỗi điểm biểu diễn một cặp giá trị của hai biến. Trong trường hợp này, chúng ta có thể sử dụng scatter plot để biểu diễn mối quan hệ giữa các cặp biến nhập - xuất.

5. Trực quan hóa dữ liệu:

Biểu đồ cột (Bar chart) hoặc biểu đồ cột ngang (Horizontal bar chart): Sử dụng để so sánh số lượng hoặc giá trị của các mặt hàng xuất, nhập hoặc tồn. Bạn có thể tạo biểu đồ cột cho từng loại hàng hóa hoặc cho mỗi thời điểm.Hiển thị dữ liệu phân cấp thông qua các hình chữ nhật, phù hợp để trực quan hóa cấu trúc tồn kho của các loại hàng hóa:

- Sử dụng để so sánh giữa các loại hàng hóa hoặc giữa các thời điểm nhập, xuất hoặc tồn kho.
- Đặc biệt hữu ích khi so sánh dữ liệu giữa các nhóm khác nhau.

Biểu đồ đường (Line chart): Sử dụng để theo dõi xu hướng thay đổi của số lượng hoặc giá trị hàng hóa theo thời gian. Điều này có thể giúp bạn nhận biết xu hướng nhập hàng, xuất hàng và tồn kho trong một khoảng thời gian nhất định.Biểu đồ đường thích hợp để theo dõi xu hướng biến động của giá trị hàng hóa nhập, xuất và tồn qua các thời điểm. Biểu đồ này sử dụng các biến như: Ngày hạch toán, Số lượng nhập, Số lượng xuất, Số lượng tồn hoặc Giá trị nhập, Giá trị xuất, Giá trị tồn:

- Dùng để đo giá trị phân phối trong 1 khoảng thời gian nhất định.
- Sử dụng để theo dõi xu hướng thay đổi của số lượng hoặc giá trị hàng hóa theo thời gian. Điều này có thể giúp bạn nhận biết xu hướng nhập hàng, xuất hàng và tồn kho trong một khoảng thời gian nhất định.

Biểu đồ pie chart (Biểu đồ tròn): Sử dụng để biểu diễn cấu trúc phân cấp của dữ liệu. Tuy nhiên, lưu ý rằng biểu đồ pie chart thường không được khuyến khích sử dụng nếu có quá nhiều loại dữ liệu, vì nó có thể trở nên khó hiểu và không rõ ràng. Biểu đồ hình tròn thích hợp để thể hiện tỷ lệ phần trăm giữa các kho theo số lượng nhập hoặc xuất. Biểu đồ này sử dụng các biến như Mã kho, Số lượng nhập hoặc Số lượng xuất để tính phần trăm:

- Phù hợp để trình bày phần trăm của mỗi loại hàng hóa trong tổng số hàng nhập, xuất hoặc tồn kho.

Bảng phân công công việc			
Nội dung công việc	**Kết quả đạt được**	**Thời gian bắt đầu, kết thúc**	**Người, đơn vị thực hiện**
Thu thập dữ liệu:	Sử dụng nguồn dữ liệu phổ biến: Lấy từ dữ liệu Công ty TNHH phân phối lân, đạm Kỹ thuật thu thập dữ liệu là: Phỏng vấn và khảo sát, lấy số liệu từ bộ phận kế toán Làm sạch dữ liệu và xử lý các giá trị thiếu: sử dụng gói library(tidyr). Các kỹ thuật thuộc gói library(tidyr) để làm sạch dữ liệu: Gộp(Pivoting), Làm sạch dữ liệu(Data Cleaning). Chuyển đổi dữ liệu phù hợp cho phân tích.	Từ 18/03 đến 24/03	Nguyễn Văn Hà
Thống kê mô tả:	Các đại lượng thống kê cơ bản: Tần suất, Trung bình, Trung vị, Phương sai và Độ lệch chuẩn, Phân phối. Biểu đồ tóm tắt dữ liệu: Sử dụng hàm boxplot() để vẽ boxplot biểu diễn phân phối dữ liệu. Sử dụng biểu đồ cột (bar plot) hoặc biểu đồ phân phối (histogram) để phân biệt các mã hàng phân phối lớn. Sử dụng biểu đồ dạng "scatter plot" (biểu đồ phân tán) hoặc "line plot" (biểu đồ đường) để mô tả mối quan hệ giữa biến số lượng và giá tiền của mặt hàng xuất nhập và tồn kho.	Từ 25/03 đến 05/04	Hồ Thị Minh Hằng

	Sử dụng biểu đồ tròn (pie chart) để so sánh ra các mặt hàng bán chạy nhất, doanh thu cao nhất. Sử dụng biểu đồ tăng trưởng(Growth Chart) để phân tích giá cả hàng hóa Sử dụng biểu đồ tương quan(Correlation Chart) để xác định mức độ tương quan giữa giá cả của các loại hàng hóa. Sử dụng biểu đồ Pareto dùng để xác định các yếu tố quan trọng nhất ảnh hưởng đến xu hướng tiêu dùng của người tiêu dùng. Sử dụng biểu đồ thời gian(Time Series Chart)) được sử dụng để theo dõi xu hướng và biến động của một biến (ví dụ: giá cả hàng hóa, doanh số bán hàng, lợi nhuận) theo thời gian. Phân tích phân phối dữ liệu. Để phân tích mối liên hệ giữa giá cả hàng hóa và xu hướng tiêu dùng của các loại hàng hóa ta sử dụng: Sử dụng ước lượng tỷ lệ để so sánh được sản phẩm bán chạy nhất. Sử dụng ước lượng hồi quy tuyến tính để dự đoán giá bán dựa trên số lượng bán và loại sản phẩm. Kiểm định t để so sánh giá bán trung bình của hai loại sản phẩm khác nhau. Kiểm định ANOVA**:** So sánh giá bán trung bình của nhiều loại sản phẩm khác nhau. Xác		

	định xem có sự khác biệt đáng kể về giá bán giữa các loại sản phẩm hay không.		
Phân tích mối liên hệ giữa các biến:	Biến nhập - xuất - tồn: Biến nhập (Inventory In): Đây là lượng hàng hoá hoặc dịch vụ được nhập vào hệ thống hoặc tồn kho trong một khoảng thời gian cụ thể. Biến xuất (Inventory Out): Đây là lượng hàng hoá hoặc dịch vụ được xuất ra khỏi hệ thống hoặc tồn kho trong một khoảng thời gian cụ thể. Biến tồn (Inventory On-hand): Đây là lượng hàng hoá hoặc dịch vụ vẫn còn lại trong hệ thống hoặc tồn kho sau khi đã trừ đi lượng hàng hoá đã bán hoặc sử dụng. Ma trận tương quan: Ma trận tương quan là một bảng chứa tất cả các hệ số tương quan giữa các cặp biến, xây dựng ma trận tương quan cho các biến nhập, xuất và tồn. Hệ số tương quan và độ đồng biến: Hệ số tương quan giữa các biến cung cấp thông tin về mức độ mối quan hệ tuyến tính giữa chúng. Độ đồng biến cho biết hướng của mối quan hệ. Nếu độ đồng biến dương, khi một biến tăng, biến kia cũng tăng; nếu độ đồng	Từ 25/03 đến 10/04	Đoàn Thị Kim Ánh

	biến âm, khi một biến tăng, biến kia giảm. Biểu đồ Scatter Plot: Biểu đồ scatter plot là một biểu đồ hai chiều trong đó mỗi điểm biểu diễn một cặp giá trị của hai biến, ở đây sử dụng scatter plot để biểu diễn mối quan hệ giữa các cặp biến nhập - xuất.		
Trực quan hóa dữ liệu:	Biểu đồ cột (Bar chart) hoặc biểu đồ cột ngang (Horizontal bar chart): Sử dụng để so sánh số lượng hoặc giá trị của các mặt hàng xuất, nhập hoặc tồn. Biểu đồ đường (Line chart): Sử dụng để theo dõi xu hướng thay đổi của số lượng hoặc giá trị hàng hóa theo thời gian. Biểu đồ pie chart (Biểu đồ tròn): Sử dụng để biểu diễn trình bày phần trăm của mỗi loại hàng hóa trong tổng số hàng nhập, xuất hoặc tồn kho. Biểu đồ hộp (Box plot): Sử dụng để mô tả phân phối của giá trị hàng hóa nhập, xuất hoặc tồn. Biểu đồ scatter plot (Biểu đồ phân tán): Sử dụng để xem mối quan hệ giữa hai biến: số lượng xuất và giá trị hàng hóa. Sử dụng để hiển thị mối quan hệ giữa hai hoặc nhiều biến: giữa giá trị hàng nhập và hàng xuất. Biểu đồ treemap (Biểu đồ cây): Để biểu diễn cấu trúc của dữ liệu hàng hóa theo từng nhóm	Từ 25/03 đến 15/04	Nguyễn Ngọc Bắc, Trần Đức Lương

	hoặc phân loại.Hiển thị dữ liệu phân cấp thông qua các hình chữ nhật, phù hợp để trực quan hóa cấu trúc tồn kho của các loại hàng hóa. Kết quả dự kiến: Cho biết số lượng hàng hóa nhập và xuất để đưa ra thống kê sử dụng loại hàng hóa theo các mùa.		
Tạo ra một số chương trình tinh chỉnh, đánh giá, trực quan dự liệu ban đầu để tiền xử lí dữ liệu trích xuất.	Viết chương trình trực quan hóa dữ liệu trích xuất bằng ngôn ngữ lập trình R.	Từ 10/04 đến 20/04	Nguyễn Ngọc Bắc, Trần Đức Lương
Một số chủ đề nâng cao:	Phân tích thành phần chính (PCA). Thành phần chính: PC1: Giải thích 40,4% biến thiên dữ liệu. Đây là thành phần chính quan trọng nhất, thể hiện mối liên hệ chặt chẽ giữa các biến: Doanh thu bán hàng Lợi nhuận gộp Tài sản lưu động Tổng tài sản PC2: Giải thích 24,6% biến thiên dữ liệu. Thành phần này thể hiện mối liên hệ giữa các biến: Vốn chủ sở hữu Nợ phải trả Lợi nhuận ròng Biến phụ: Số lượng nhân viên: Không có mối liên hệ rõ ràng với các thành phần chính. Lý do: PC1 tập trung vào các biến liên quan đến hiệu quả hoạt động (doanh thu, lợi	Từ 20/04 đến 09/05	Hằng, Lương, Bắc, Ánh, Hà

	nhuận, tài sản). PC2 tập trung vào các biến liên quan đến cấu trúc tài chính (vốn chủ sở hữu, nợ phải trả). Số lượng nhân viên không có mối liên hệ trực tiếp với hiệu quả hoạt động hay cấu trúc tài chính trong dữ liệu này. Lưu ý: Phân tích này chỉ dựa trên thông tin được cung cấp trong ảnh chụp màn hình. Việc xác định thành phần chính và biến phụ có thể thay đổi tùy thuộc vào phương pháp phân tích và dữ liệu cụ thể. Kết luận: PC1 và PC2 là hai thành phần chính trong phân tích CPA, giải thích tổng cộng 65% biến thiên dữ liệu. Doanh thu bán hàng, lợi nhuận gộp, tài sản lưu động, tổng tài sản, vốn chủ sở hữu, nợ phải trả và lợi nhuận ròng là các biến quan trọng góp phần hình thành các thành phần chính. Số lượng nhân viên không có mối liên hệ rõ ràng với các thành phần chính trong dữ liệu này.		
Viết báo cáo	Chương 1: Giới thiệu tổng quan về thư viện ggplot và phân tích thống kê	Từ 20/03 đến 30/03	Đoàn Thị Kim Ánh
	Chương 2: Phân tích thăm dò dữ liệu	Từ 30/04 đến 10/05	Hồ Thị Minh Hằng
	Chương 3: Trực quan dữ liệu	Từ 30/04 đến 10/05	Hồ Thị Minh Hằng, Đoàn Thị Kim Ánh

LỜI GIỚI THIỆU

Bài báo cáo này tập trung vào việc sử dụng ngôn ngữ lập trình R để thăm dò và trực quan hóa dữ liệu về giá cả hàng hóa và xu hướng tiêu dùng của các loại hàng hóa. Trong thời đại số hóa ngày nay, việc thu thập, phân tích và hiểu được dữ liệu là yếu tố then chốt để đưa ra các quyết định kinh doanh hiệu quả.

Khám phá cách R, một trong những ngôn ngữ lập trình phổ biến nhất trong lĩnh vực khoa học dữ liệu, có thể được áp dụng để thăm dò và phân tích dữ liệu thị trường. Bằng cách sử dụng các kỹ thuật thống kê và trực quan hóa dữ liệu, chúng ta sẽ đi sâu vào những thông tin quan trọng về giá cả và hành vi tiêu dùng của khách hàng.

Qua bài báo cáo này, ta sẽ có cơ hội:

- Hiểu rõ hơn về R và khả năng áp dụng của nó trong phân tích dữ liệu thị trường.
- Khám phá các phương pháp thăm dò dữ liệu và trực quan hóa thông qua các gói phần mềm phổ biến như ggplot2 và dplyr.
- Đánh giá và phân tích xu hướng giá cả và tiêu dùng của các loại hàng hóa thông qua dữ liệu thực tế.
- Rút ra những thông tin quan trọng để hỗ trợ quyết định kinh doanh và chiến lược tiếp thị.

Bằng cách kết hợp sức mạnh của R và sự sáng tạo trong phân tích dữ liệu, chúng ta sẽ có cơ hội tối ưu hóa hiệu suất kinh doanh và đáp ứng một cách linh hoạt với nhu cầu thị trường đang thay đổi liên tục.

NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

……

……

……

……

……

……

……

……

……

……

……

……

……

……

……

……

Điểm:……..…...

(Bằng chữ:……...……..)

Hà Nội, ngày 27 tháng 05 năm 2024

GIẢNG VIÊN HƯỚNG DẪN

Trần Chí Lê

MỤC LỤC

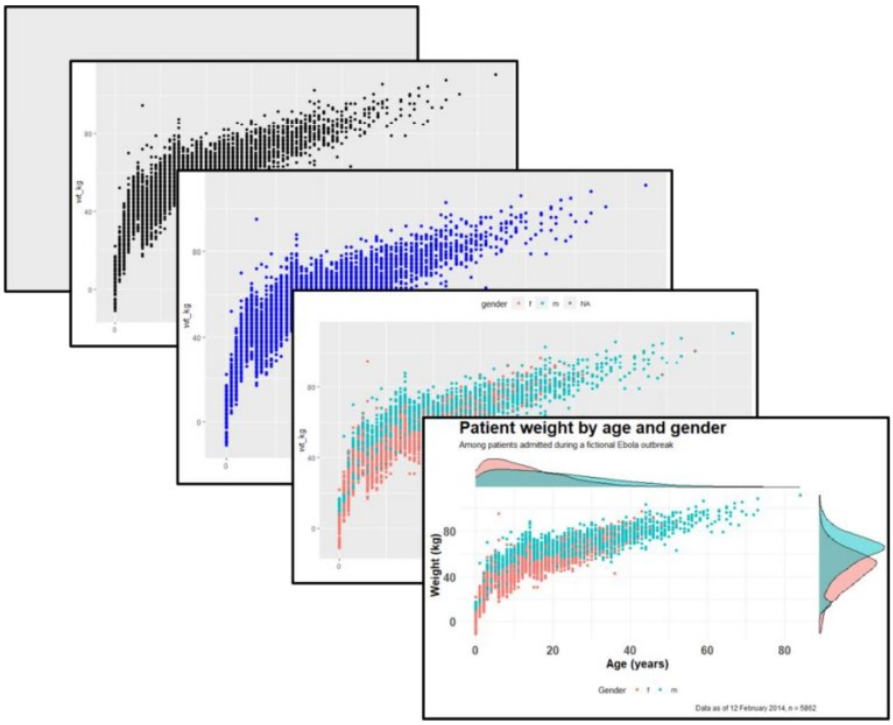
DANH MỤC HÌNH VẼ VÀ BẢNG BIỂU

CHƯƠNG 1

GIỚI THIỆU TỔNG QUAN VỀ THƯ VIỆN GGPLOT VÀ PHÂN TÍCH THỐNG KÊ

1. 1. Giới thiệu tổng quan về thư viện ggplot

1.1.1. Biểu đồ với package ggplot2



Hình 1.1. Minh họa quy trình thêm các lớp hàm phân tích trong ggplot.

```
mpg
#> # A tibble: 234 × 11
#>   manufacturer model displ  year   cyl trans      drv     cty   hwy fl    class
#>   <chr>        <chr> <dbl> <int> <int> <chr>      <chr> <int> <int> <chr> <chr>
#> 1 audi         a4      1.8  1999     4 auto(l5)   f        18    29 p     compa…
#> 2 audi         a4      1.8  1999     4 manual(m5) f        21    29 p     compa…
#> 3 audi         a4      2    2008     4 manual(m6) f        20    31 p     compa…
#> 4 audi         a4      2    2008     4 auto(av)   f        21    30 p     compa…
#> 5 audi         a4      2.8  1999     6 auto(l5)   f        16    26 p     compa…
#> 6 audi         a4      2.8  1999     6 manual(m5) f        18    26 p     compa…
#> # ℹ 228 more rows
```

Bảng 1.1. Dữ liệu quan sát về các mẫu ô tô (nguồn: ggplot2).

a. Cú pháp cơ bản

Chúng ta có thể minh họa cú pháp cơ bản như sau:

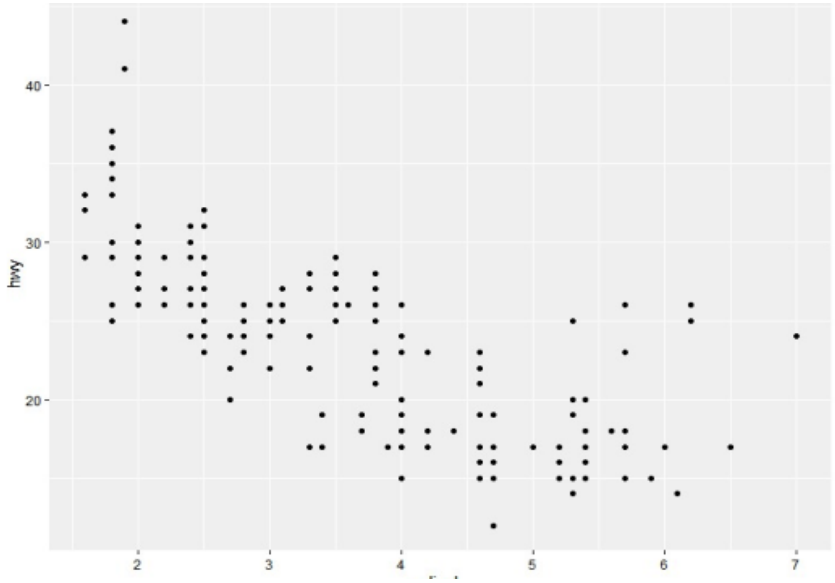
```
ggplot(data = my_data)+                    # sử dụng dữ liệu "my_data"
  geom_yyy(                               # thêm một lớp các hàm-hình biểu đồ
    mapping = aes(x = col1, y = col2),    # gán dữ liệu tới các trục
    color = "red")+                       # thêm một số đặc điểm khác (như màu sắc)
  labs()+                                 # thêm tiêu đề, nhãn, bảng số,..
  theme()                                 # điều chỉnh cỡ chữ, màu sắc, phông chữ
```

b. Gán các biến dữ liệu cho biểu đồ

Hầu hết các hàm-hình geom phải được cho biết cái gì được s dụng để vẽ biểu đồ, vì vậy chúng ta phải cung cấp cách map (gán) các biến số trong dữ liệu tới các thành phần của biểu đồ như là các trục, màu đối tượng, kích thước đối tượng, v.v. Đối với hầu hết các geoms, các thành phần thiết yêu phải được gán tới các cột trong dữ liệu là trục x, và (nếu cần) là trục y.

Ví dụ 1.1. Trong lệnh ggplot() dưới đây, dữ liệu được thiết lập là bộ dữ liệu mpg. Trong đó đối số mapping = aes(), cột displ được gán cho trục x, và cột hwy được gán cho trục y.

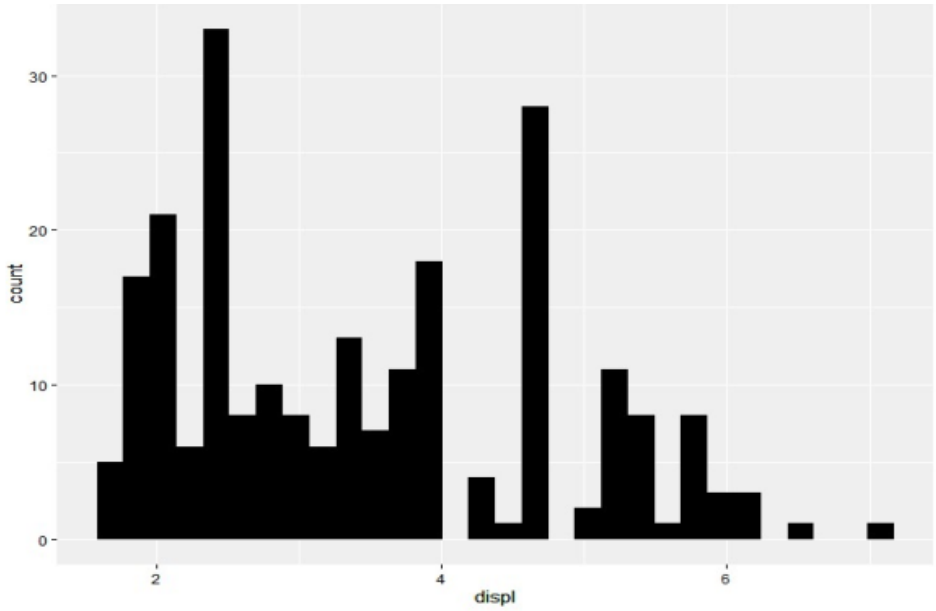
```
ggplot(data = my_data)+
geom_poit(mapping = aes(x = displ, y = hwy))
# kết quả hiển thị:
```



Hình 1.2. Cách gán biến số vào các trục.

Ví dụ 1.2. Lệnh sau tương tự Ví dụ 1.1, chỉ có một sự khác biệt nhỏ về cách mapping và hàm geom. Hàm geom_histogram() chỉ yêu cầu gán cột cho trục x, bởi vì trục số lượng y được tạo ra một cách tự động bằng số phần tử của mỗi biến x.

```
ggplot(data = mpg, mapping = aes(x = displ))+
  geom_histogram()
# kết quả hiển thị:
```



Hình 1.3. Cách gán biến vào lớp hàm geom_histogram.

c. Tính thẩm mỹ trong biểu đồ

Tính thẩm mỹ trong biểu đồ có thể là màu sắc, kích thước, độ trong suốt, vị trí, v.v. của dữ liệu được vẽ. Không phải tất cả các geoms sẽ có các tùy chọn về tính thẩm mỹ, trang trí giống nhau, nhưng một số tùy chọn được áp dụng với phần lớn các geoms. Dưới đây là một số trang trí hay ghp:

▪ shape = Hiển thị một điểm với hàm geom_point() dưới dạng dấu chấm, ngôi sao, hình tam giác hohc hình vuông,…

▪ fill = Màu sắc bên trong (vd: của cột hoặc boxplot).

▪ color = Đường bên ngoài của cột, boxplot, v.v., hoặc màu của điểm nếu sử dụng hàm geom_point().

▪ size = Kích thước (vd: độ dày của đường, kích thước của điểm).

▪ alpha = Độ trong suốt (1 = bình thường, 0 = vô hình).

▪ binwidth = Độ rộng các bins trong biểu đồ histogram.

▪ width = Độ rộng của các cột trong "biểu đồ cột".

▪ linetype = Kiểu của đường (vd: liền, nét đứt, chấm chấm).

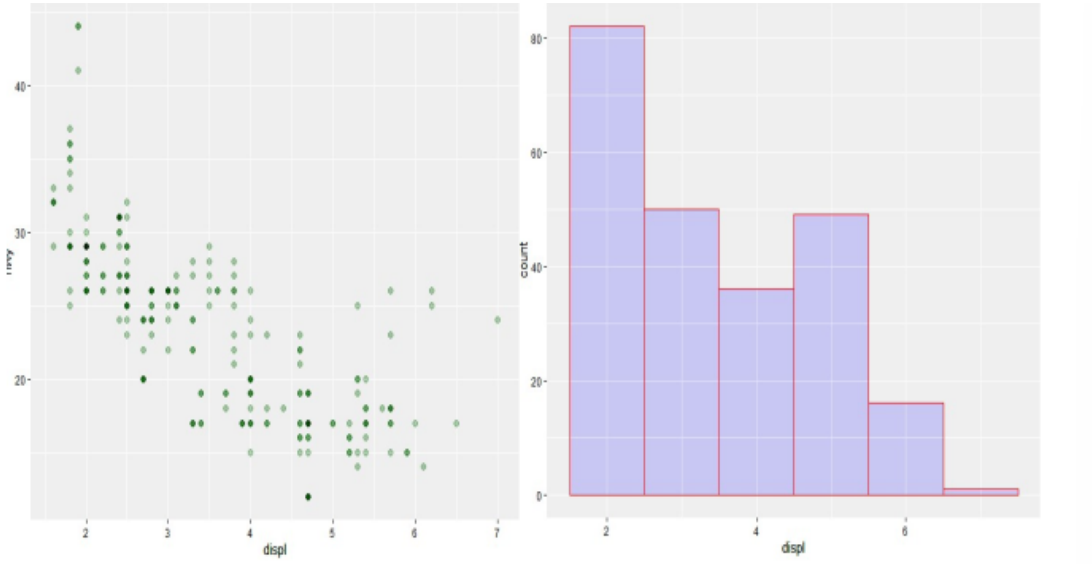
Trang trí của đối tượng biểu đồ có thể được gán giá trị theo hai cách: Gán một giá trị tĩnh
(vd: color = "blue") để áp dong cho tất cả các quan sát được vẽ biểu đồ hoặc gán cho từng biến của dữ liệu (vd: color = hospital) để hiển thị từng quan sát phụ thuộc vào giá trị của nó trong biến đó.

• **Trang trí với một giá trị tĩnh**

Nếu muốn yếu tố trang trí cho đối tượng biểu đồ là tĩnh, nghĩa là - giống nhau đối với mọi quan sát trong dữ liệu, chúng ta gán nó bên trong geom nhưng ở bên ngoài đối số mapping = aes() . Các phép gán này có thể ví dụ như: size = 1 hoặc color = "blue".

Ví dụ 1.3. Xét bộ dữ liệu mpg, với phép gán giá trị tĩnh về màu sắc

```r
# biểu đồ vô hướng
ggplot(data = mpg, mapping = aes(x = displ, y = hwy))+    # gán dữ liệu và các trục
  geom_point(color = "darkgreen", size = 2, alpha = 0.2)  # hàm tạo điểm
# biểu đồ phân bố
ggplot(data = mpg, mapping = aes(x = displ))+             # gán dữ liệu và các trục
  geom_histogram(                                         # hàm phân bố
    binwidth = 1,                                         # độ rộng cột
    color = "red",                                        # đường màu
    fill = "blue",                                        # màu tô bên trong
    alpha = 0.1)                                          # độ trong
# kết quả hiện thị
```



Hình 1.4. Phép gán tĩnh.

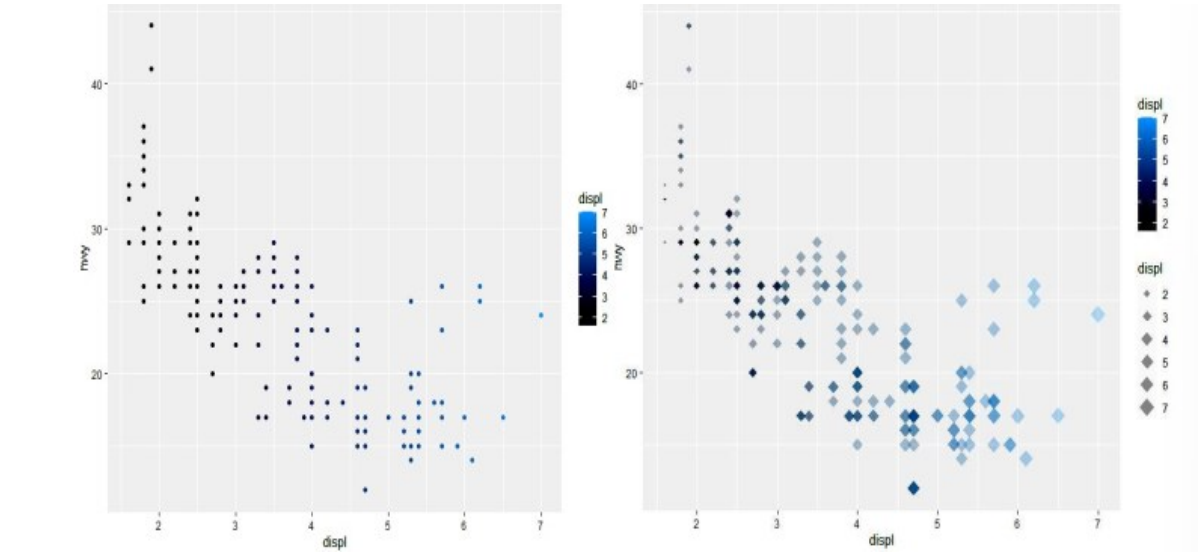
• **Trang trí theo giá trị của từng biến**

Để thực hiện được điều này, chúng ta gán yɮu tố trang trí của biểu đồ với một biɮn (không trong dấu ngoặc kép). Điều này phải được thực hiện bên trong một hàm mapping = aes()

Ví dụ 1.4. Xét bộ dữ liệu mpg, với phép trang trí theo biến x = displ bởi màu sắc-color hoặc kích cỡ-size.

```
# biểu đồ vô hướng 1
ggplot(data = mpg, mapping = aes( x = displ, y = hwy, color = displ))+
  geom_point()
```

```
# biểu đồ vô hướng 2
ggplot(data = mpg, mapping = aes( x = displ, y = hwy,color = displ,size = displ))+
  geom_point(shape = "diamond", alpha = 0.3)
# kết quả hiển thị:
```



Hình 1.5. Trang trí theo biến

Nhận xét: Việc gán các yếu tố trang trí bên trong đối số mapping = aes() có thể được viết ở một số chỗ trong các lệnh vẽ biểu đồ và thậm chí có thể được viết nhiều lần. Nó có thể được viết trong lệnh ggplot() trên cùng, hoặc cho từng geom riêng lẻ bên dưới. Các kiểu viết bao gồm:

▪ Các phép gán được thực hiện ở lệnh ggplot() trên cùng sẽ được mặc định kế thừa ở bất kỳ các geom bên dưới, giống như cách mà x = và y = được kế thừa.

▪ Các phép gán được thực hiện trong một geom chỉ áp dong cho geom đó.

▪ Tương tự, data = được chỉ định cho lệnh ggplot() ở trên đầu sẽ áp dụng mặc định cho tất cả các geom bên dưới.

Ví dụ 1.6. Mỗi lệnh sau sẽ tạo ra cùng một biểu đồ giống nhau:

```
# Mẫu thứ nhất
ggplot(data = mpg, mapping = aes(x = displ))+
  geom_histogram()
# Mẫu thứ hai
```

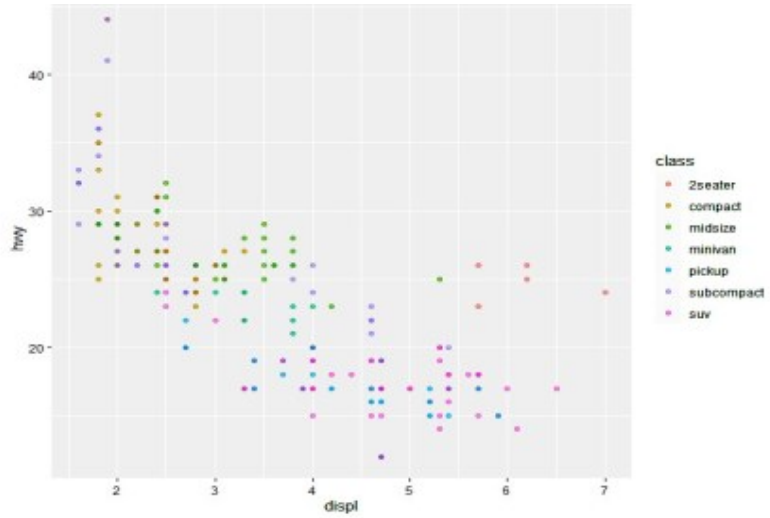
```
ggplot(data = mpg)+
  geom_histogram(mapping = aes(x = displ))
# Mẫu thứ ba
ggplot()+
  geom_histogram(data = mpg, mapping = aes(x = displ))
```

• Trang trí theo nhóm đối tượng

Ví dụ 1.7. Xét lại dữ liệu mpg, để có thông tin chi tiết hơn về dữ liệu, chúng ta có thể sử dụng màu sắc để phân biệt tới biến class hiển thị kiểu dáng của từng chiếc xe. Chúng ta sẽ đặt mapping = aes(color = "class") khi đó một chữ giải tự động xuất hiện . Phép gán này có thể được thực hiện bên trong mapping = aes() ở lệnh ggplot() đầu tiên (và được thừa kế bởi các geom), hoặc nó có thể được đặt trong một mapping = aes() riêng biệt bên trong geom.

Cả hai cách tiếp cận được trình bày dưới đây:

```
# cách tiếp cận 1
ggplot(data = mpg,mapping = aes(x = displ, y = hwy, color = class))+
  geom_point(alpha = 0.5)
# cách tiếp cận 2
ggplot(data = mpg, mapping = aes(x = displ, y = hwy))+
  geom_point(mapping = aes(color = class),alpha = 0.5)
# kết quả hiển thị:
```



Hình 1.6. Trang trí màu sắc theo nhóm

d. Gán nhãn cho biểu đồ

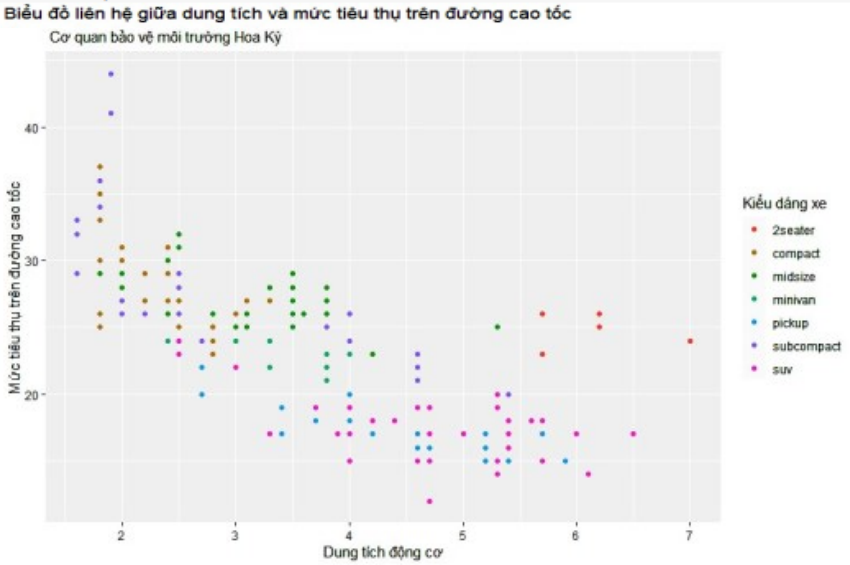
Việc đặt tên cho tiêu đề biểu đồ, tên các biến trên trục, các chú thích là công việc không thể thiếu khi vẽ biểu đồ, và việc này được thực hiện với hàm labs() bằng cách them dấu + như cách chúng ta thêm các geoms.

Bên trong hàm labs(), cung cấp các chuỗi ký tự cho các đối số sau:

- x = và y = Tiêu đề trục x và trục y (nhãn).
- title = Tiêu đề chính của biểu đồ.
- subtitle = Tiêu đề phhụ của biểu đồ, nhỏ hơn và đặt bên dưới tiêu đề chính.
- caption = Chú thích của biểu đồ, mặc định ở góc phải dưới.

Ví dụ 1.8. Ví dụ dưới đây là biểu đồ chúng ta đã tạo ở Ví dụ 1.7 nhưng có thêm các nhãn:

```
Bieu_do1<-ggplot(data = mpg) +
  geom_point(mapping = aes(x = displ, y = hwy,color=class))+
  labs(title = "Biểu đồ liên hệ giữa dung tích và mức tiêu thụ trên đường cao tốc",
     subtitle = " Cơ quan bảo vệ môi trường Hoa Kỳ",x = "Dung tích động cơ",y = "Mức
tiêu thụ trên đường cao tốc", color = "Kiểu dáng xe")
# in biểu đồ
Bieu_do1
# Kết quả hiển thị:
```



Hình 1.7. Gán nhãn cho biểu đồ

e. Căn chỉnh trong biểu đồ

Thực hiện theo hai cách: Căn chỉnh theo mặc định sẵn có và căn chỉnh cá nhân đơn lẻ

• Căn chỉnh theo mặc định

Chúng ta có thể sử dụng một số hàm chủ đề hoàn chỉnh bên dưới đây.

+ theme_gray(): Chủ đề ggplot2 đặc trưng với nền màu xám và đường lưới màu trắng, được thiết kế để đưa dữ liệu về phía trước nhưng vẫn giúp việc so sánh trở nên dễ dàng.

+ theme_bw(): Chủ đề ggplot2 tối trên ánh sáng cổ điển. Có thể hoạt động tốt hơn cho bài thuyết trình trình chiếu bằng máy chiếu.

+ theme_linedraw(): Một chủ đề chỉ có các đường màu đen có chiều rộng khác nhau trên nền trạng, gợi nhớ đến một bản vẽ đường. Phục vụ mục đích tương tự như theme_bw(). Lưu ý rằng chủ đề này có một số dòng rất mỏng (<< 1 pt) khi in ấn rất dễ mất hình ảnh.

+ theme_light(): Một chủ đề tương tự như theme_linedraw() nhưng có các đường và trục màu xám nhạt, để hướng sự chú ý nhiều hơn tới dữ liệu.

+ theme_dark(): Tương tự màu tối của theme_light(), với kích thước dòng tương tự nhưng nền tối, hữu ích để làm nổi bật những đường màu mảnh.

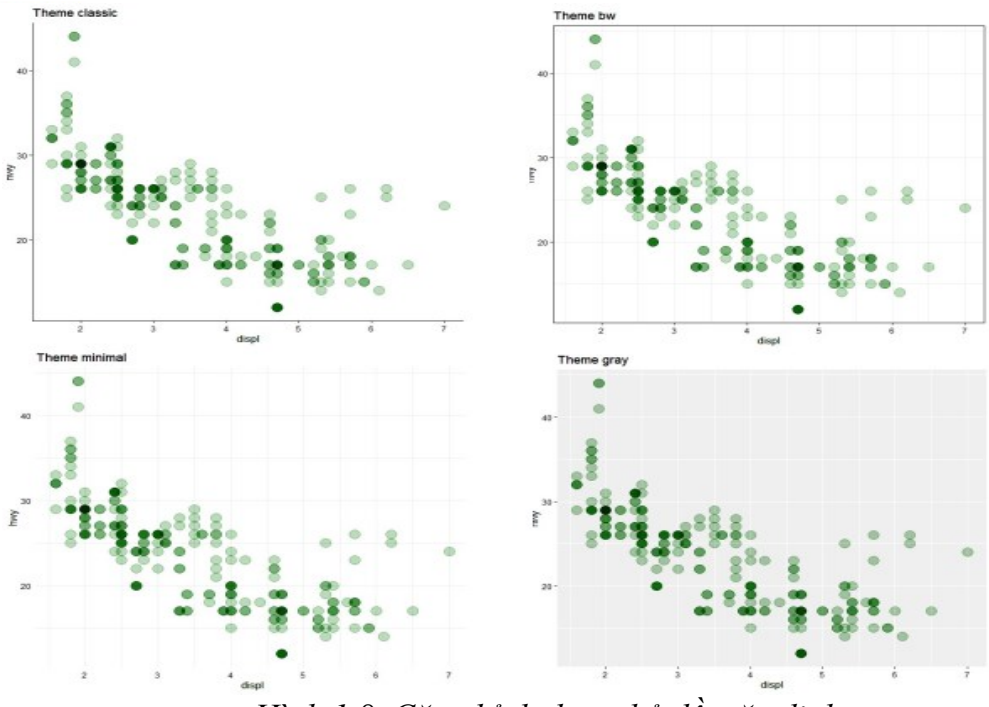
+ theme_minimal(): Một chủ đề tối giản không có chú thích nền.

+ theme_classic(): Một chủ đề có giao diện cổ điển với các đường trục x và y và

không có đường lưới.
+ theme_void(): Một chủ đề hoàn toàn trống rỗng.
+ theme_test(): Một chủ đề cho bài kiểm tra đơn vị trực quan. Lý tưởng nhất là nó không bao giờ thay đổi ngoại trừ cho các tính năng mới.

Ví dụ 1.9. Ví dụ dưới đây minh họa một vài căn chỉnh theo chủ đề mặc định:

```
# căn chỉnh theo chủ đề Theme classic
ggplot(data = mpg, mapping = aes(x = displ, y = hwy))+
  geom_point(color = "darkgreen", size = 0.5, alpha = 0.2)+
  labs(title = "Theme classic")+
  theme_classic()
# căn chỉnh theo chủ đề Theme bw
ggplot(data = mpg, mapping = aes(x = displ, y = hwy))+
  geom_point(color = "darkgreen", size = 0.5, alpha = 0.2)+
  labs(title = "Theme bw")+
  theme_bw()
# căn chỉnh theo chủ đề Theme minimal
ggplot(data = mpg, mapping = aes(x = displ, y = hwy))+
  geom_point(color = "darkgreen", size = 0.5, alpha = 0.2)+
  labs(title = "Theme minimal")+
  theme_minimal()
# căn chỉnh theo chủ đề Theme gray
ggplot(data = mpg, mapping = aes(x = displ, y = hwy))+
  geom_point(color = "darkgreen", size = 0.5, alpha = 0.2)+
  labs(title = "Theme gray")+
  theme_gray()
# kết quả hiển thị:
```



Hình 1.8. Căn chỉnh theo chủ đề mặc định.

Căn chỉnh cá nhân đơn lẻ

Cú pháp cơ bản là:

+ Bên trong hàm theme(), hãy viɮt tên đối số cho phần tỹ biểu đồ mà ta muốn chỉnh sửa, chẳng hạn như plot.title =.
+ Cung cấp một hàm element_() tới đối số.
+ Thường sử dụng nhất là element_text(), một số khác bao gồm element_rect() chọn màu nền cho canvas, hoặc element_blank() để xóa các phần tử biểu đồ.
+ Bên trong hàm element_(), xác định giá trị đối số cần gán để điều chỉnh theo ý bạn mong muốn.

Sau đây là một số đối số phổ biến của hàm theme():

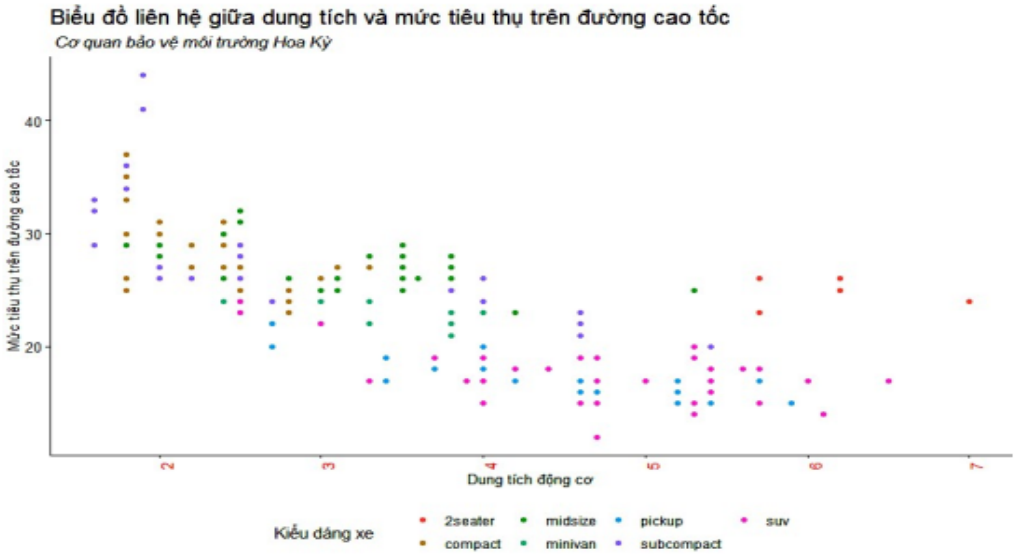
Đối số	Những điều chỉnh
plot.title = element_text()	Tiêu đề chính
plot.subtitle = element_text()	Tiêu đề phụ
plot.caption = element_text()	Liên quan tới caption (kiểu font, màu sắc, kích cỡ, góc độ, vjust, hjust…)
axis.title = element_text()	Tiêu đề trục (cả trục x và y) (kích cỡ, góc độ, màu sắc…)
axis.title.x = element_text()	Chỉ tiêu đề trục x (sử dụng .y để chỉ áp dụng với trục y)
axis.text = element_text()	Văn bản trên trục (cả trục x và y)
axis.text.x = element_text()	Chỉ văn bản trục x (sử dụng .y để chỉ áp dụng với trục y)
axis.ticks = element_blank()	Loại bỏ ticks của trục
axis.line = element_line()	Đường trục (màu sắc, kích thước, kiểu đường: nét đứt, nét liền mảnh, v.v.)
strip.text = element_text()	Văn bản trong Facet strip (màu sắc, kích thước, góc độ…)
strip.background = element_rect()	facet strip (tô màu, màu sắc, kích thước…)

Bảng 1.2. Các đối số hay sử dụng cho việc căn chỉnh trong hàm theme.

Ví dụ 1.10. Ví dɒ dưới đây sỹ dɒng biểu đồ từ file đã lưu Bieu_do1 và thêm vào một vài
căn chỉnh theo tùy chọn cá nhân

```
Bieu_do1+
  theme_classic()+
  theme(legend.position = "bottom",
    plot.title = element_text(size = 15),
    plot.caption = element_text(hjust = 0),
```

```
    plot.subtitle = element_text(face = "italic"),
    axis.text.x = element_text(color = "red", size = 10, angle = 90),
    axis.text.y = element_text(size = 10),
    axis.title = element_text(size = 10))
```



Hình 1.9. Căn chỉnh biểu đồ theo tùy chọn.

Giải thích:

▪ legend.position = là đặc biệt nhất vì nó chỉ chấp nhận các giá trị đơn giản như "bottom", "top", "left", và "right". các đối số liên quan đến văn bản yêu cầu bạn đặt các chi tiết bên trong hàm element_text().

▪ Cỡ chư tiêu đề với element_text(size = 30).

▪ Căn lề caption với element_text(hjust = 0) (từ trái qua phải).

▪ Tiêu đề phụ được in nghiêng với element_text(face = "italic").

f) Phối màu sắc, tô màu, thang đo

• Phối màu

Để phối màu sắc của các đối tượng biểu đồ (geoms/shapes) như điểm, cột, đường, ô, v.v., chúng ta sử dụng các tham số `color` (màu bên ngoài) hoặc `fill` (màu bên trong). Riêng đối với `geom_point()`, chúng ta chỉ có thể điều chỉnh `color =` để xác định màu của điểm. Khi thiết lập màu hoặc tô màu, chúng ta có thể sử dụng tên màu được R nhận dạng như "red" (để xem danh sách các màu đầy đủ, gõ `?colors` trong cửa sổ soạn thảo hoặc ấn F1).

1.2. Phân tích thống kê

Phân tích thống kê trong trực quan hóa dữ liệu là quá trình sử dụng các phương pháp thống kê để hiểu và phân tích các dữ liệu được biểu diễn dưới dạng đồ thị, biểu đồ hoặc

hình ảnh khác. Mục tiêu của phân tích thống kê trong trực quan hóa dữ liệu là trích xuất thông tin ý nghĩa từ dữ liệu và giúp người sử dụng hiểu rõ hơn về các mối quan hệ, xu hướng và biến đổi trong dữ liệu.

Mô tả dữ liệu: Đây là quá trình khám phá và mô tả dữ liệu bằng các đặc điểm thống kê như trung bình, phương sai, phân phối, tần số, và các đặc tính khác. Trực quan hóa dữ liệu thông qua biểu đồ cung cấp cái nhìn tổng quan về dữ liệu và giúp nhận biết các đặc điểm chính.

Tổng quan về dữ liệu: Đầu tiên, ta cần xem xét tổng quan về dữ liệu, bao gồm số lượng quan sát, số lượng biến, và kiểu dữ liệu của mỗi biến. Điều này giúp chúng ta hiểu cơ bản về phạm vi và tính chất của dữ liệu.

Thống kê mô tả: Thống kê mô tả là việc sử dụng các chỉ số thống kê như trung bình, trung vị, phương sai, độ lệch chuẩn, tần suất, và phân phối để mô tả các đặc điểm của dữ liệu. Thông qua thống kê mô tả, chúng ta có thể hiểu về trung bình và biến thiên của các biến, sự phân bố của chúng, và các giá trị cực đại và cực tiểu.

Biểu đồ và biểu đồ: Sử dụng biểu đồ và biểu đồ để trực quan hóa dữ liệu là một phần quan trọng của mô tả dữ liệu. Các biểu đồ phổ biến bao gồm biểu đồ cột, biểu đồ đường, biểu đồ phân tán, biểu đồ hộp và râu, và biểu đồ vòng. Các biểu đồ này giúp chúng ta thấy các mối quan hệ, xu hướng và biến đổi trong dữ liệu một cách trực quan và dễ hiểu.

Phân tích tương quan: Phân tích tương quan giữa các biến là một phần quan trọng của mô tả dữ liệu. Bằng cách xem xét mối quan hệ giữa các biến, chúng ta có thể hiểu được cách mà chúng tương tác và ảnh hưởng lẫn nhau.

Kiểm tra phân phối: Kiểm tra phân phối của dữ liệu giúp chúng ta hiểu về cách mà dữ liệu được phân bố trong mỗi biến. Các phân phối phổ biến bao gồm phân phối chuẩn, phân phối Poisson, và phân phối nhị phân.

Xử lý dữ liệu bị thiếu và ngoại lai: Trong quá trình mô tả dữ liệu, chúng ta cần xem xét xem có dữ liệu bị thiếu hoặc ngoại lai không và cần xử lý chúng như thế nào. Điều này có thể bao gồm điền giá trị bị thiếu, hoặc loại bỏ các quan sát ngoại lai.

Quá trình mô tả dữ liệu là bước quan trọng trong quy trình phân tích dữ liệu và giúp chúng ta có cái nhìn tổng quan về dữ liệu và hiểu rõ hơn về tính chất và cấu trúc của nó.

Phân tích so sánh: Phân tích thống kê trong trực quan hóa dữ liệu cho phép so sánh các nhóm dữ liệu khác nhau và xác định sự khác biệt giữa chúng. Các phương pháp thống kê như kiểm định giả thuyết, phân tích phương sai (ANOVA) và kiểm tra tương quan được sử dụng để xác định sự khác biệt có ý nghĩa giữa các nhóm.

Chọn phương pháp phù hợp: Trước tiên, cần xác định phương pháp phù hợp để phân tích sự khác biệt giữa các nhóm. Phương pháp này phụ thuộc vào loại dữ liệu và mục tiêu của nghiên cứu. Các phương pháp phổ biến bao gồm kiểm định t, kiểm định ANOVA, kiểm định Mann-Whitney U, và kiểm định Kruskal-Wallis.

Xác định các nhóm so sánh: Tiếp theo, cần xác định các nhóm mà bạn muốn so sánh. Điều này có thể là các nhóm thuộc cùng một biến hoặc nhóm khác nhau trong dữ liệu.

Thực hiện phân tích: Sau khi xác định các nhóm, tiến hành phân tích để đánh giá sự khác biệt giữa chúng. Sử dụng phương pháp thống kê phù hợp để tính toán giá trị thống kê và xác định xem liệu sự khác biệt giữa các nhóm có ý nghĩa thống kê không.

Đánh giá kết quả: Dựa trên kết quả của phân tích, đánh giá xem liệu sự khác biệt giữa các nhóm có ý nghĩa thống kê hay không. Thường thì nếu giá trị p (p-value) nhỏ hơn một ngưỡng ý nghĩa nhất định (thường là 0.05), thì chúng ta kết luận rằng có sự khác biệt đáng kể giữa các nhóm.

Trực quan hóa kết quả: Trực quan hóa kết quả của phân tích so sánh có thể giúp hiểu rõ hơn về sự khác biệt giữa các nhóm. Sử dụng biểu đồ như biểu đồ cột, biểu đồ hộp và râu, hoặc biểu đồ phân tán để minh họa sự khác biệt giữa các nhóm.

Kiểm tra giả thuyết: Cuối cùng, đánh giá kết quả của phân tích so sánh và kiểm tra giả thuyết. Xác định liệu kết quả có hỗ trợ hay phủ định giả thuyết ban đầu và đưa ra kết luận về sự khác biệt giữa các nhóm dữ liệu.

Phân tích so sánh là một công cụ mạnh mẽ trong phân tích thống kê, giúp chúng ta hiểu rõ hơn về các mối quan hệ và sự khác biệt trong dữ liệu.

Khám phá mối quan hệ: Trực quan hóa dữ liệu cùng với phân tích thống kê giúp phát hiện ra các mối quan hệ, tương tác và xu hướng trong dữ liệu. Các biểu đồ như biểu đồ phân tán, biểu đồ dòng, biểu đồ cột, biểu đồ đường được sử dụng để thể hiện mối quan hệ giữa các biến.

Biểu đồ phân tán (Scatter plots): Sử dụng biểu đồ phân tán để minh họa mối quan hệ giữa hai biến liên tục. Biểu đồ này cho phép bạn xem xét mức độ tương quan và hình dạng của mối quan hệ, bao gồm cả hình dạng tuyến tính và phi tuyến tính.

Ma trận tương quan (Correlation matrix): Tạo ma trận tương quan để đánh giá mối quan hệ tuyến tính giữa tất cả các cặp biến liên tục trong dữ liệu. Giá trị của hệ số tương quan thể hiện độ mạnh và hướng của mối quan hệ. Một ma trận tương quan có thể được trực quan hóa bằng cách sử dụng biểu đồ heatmap.

Biểu đồ hộp và râu (Box plots): Biểu đồ hộp và râu là công cụ hữu ích để so sánh phân phối của một biến dọc theo các nhóm của một biến khác. Chúng cho phép bạn nhìn thấy sự phân tán và median của mỗi nhóm, cũng như các giá trị ngoại lai.

Biểu đồ cột (Bar plots) và biểu đồ dòng (Line plots): Sử dụng biểu đồ cột và biểu đồ dòng để so sánh giá trị trung bình hoặc tỷ lệ của một biến phụ thuộc theo các nhóm của biến độc lập. Điều này giúp hiểu rõ hơn về sự khác biệt giữa các nhóm.

Phân tích phương sai (Variance analysis): Phân tích phương sai là một phương pháp để kiểm tra xem có sự khác biệt đáng kể giữa các nhóm dữ liệu không. Nó đánh giá sự biến động của dữ liệu trong mỗi nhóm và xem xét xem có sự khác biệt đáng kể giữa các nhóm hay không.

Phân tích hồi quy (Regression analysis): Phân tích hồi quy giúp xác định mối quan hệ giữa một biến phụ thuộc và một hoặc nhiều biến độc lập. Điều này cho phép dự đoán giá trị của biến phụ thuộc dựa trên các biến độc lập.

Tóm lại, khám phá mối quan hệ trong phân tích thống kê chi tiết giúp hiểu rõ hơn về cấu trúc và đặc điểm của dữ liệu, đồng thời cung cấp thông tin cần thiết để đưa ra những kết luận có ý nghĩa.

Dự đoán và mô hình hóa: Phân tích thống kê trong trực quan hóa dữ liệu cũng có thể được sử dụng để dự đoán xu hướng tương lai và xây dựng các mô hình dự đoán. Các phương pháp như hồi quy tuyến tính, hồi quy logistic và cây quyết định thường được sử dụng để dự đoán và mô hình hóa dữ liệu.

Hồi quy tuyến tính: Hồi quy tuyến tính là một phương pháp phổ biến để dự đoán giá trị của biến phụ thuộc dựa trên các biến độc lập. Mô hình hồi quy tuyến tính tìm ra một đường thẳng hoặc một mặt phẳng tối ưu sao cho tổng bình phương của sai số giữa giá trị dự đoán và giá trị thực tế là nhỏ nhất.

Hồi quy logistic: Hồi quy logistic được sử dụng khi biến phụ thuộc là một biến nhị phân hoặc một biến phân loại. Mô hình hồi quy logistic ước lượng xác suất của biến phụ thuộc bằng cách sử dụng một hàm logistic để chuyển đổi các giá trị liên tục thành xác suất.

Mô hình tuyến tính tổng quát (GLM): Mô hình tuyến tính tổng quát là một lớp mô hình linh hoạt hơn so với hồi quy tuyến tính thông thường, cho phép ước lượng và dự đoán giá trị của biến phụ thuộc dựa trên các biến độc lập với phân phối không nhất thiết là Gaussian.

Mạng nơ-ron: Mạng nơ-ron là một phương pháp mô hình hóa linh hoạt và mạnh mẽ trong phân tích dự đoán. Mạng nơ-ron có khả năng học và tự điều chỉnh từ dữ liệu để tạo ra các dự đoán chính xác với độ phức tạp khác nhau.

Mô hình cây quyết định và rừng ngẫu nhiên: Mô hình cây quyết định và rừng ngẫu nhiên là các phương pháp dựa trên cấu trúc cây quyết định để dự đoán giá trị của biến phụ thuộc dựa trên các biến độc lập. Rừng ngẫu nhiên kết hợp nhiều cây quyết định để tạo ra một dự đoán tốt hơn.

Mô hình chuỗi thời gian: Trong trường hợp dữ liệu được thu thập theo thời gian, mô hình chuỗi thời gian được sử dụng để dự đoán xu hướng và biến động của dữ liệu trong tương lai dựa trên các quan sát trước đó.

Quá trình dự đoán và mô hình hóa trong phân tích thống kê chi tiết giúp hiểu rõ hơn về mối quan hệ giữa các biến và dự đoán giá trị của biến phụ thuộc dựa trên các biến độc lập. Điều này cung cấp thông tin quan trọng để đưa ra các quyết định dựa trên dữ liệu.

Kiểm tra giả thuyết: Phân tích thống kê giúp kiểm tra các giả thuyết khoa học và đưa ra kết luận về tính chính xác của chúng. Các phương pháp như kiểm định t, kiểm định ANOVA, và kiểm định tương quan được sử dụng để kiểm tra các giả thuyết và đưa ra kết luận.

Dưới đây là các bước chính để kiểm tra giả thuyết trong phân tích thống kê chi tiết:

Xác định giả thuyết: Bước đầu tiên là xác định giả thuyết không và giả thuyết thay thế. Giả thuyết không (H0) là giả định ban đầu hoặc giả định mà không có sự can thiệp. Giả thuyết thay thế (H1) là giả định mà chúng ta muốn chứng minh.

Chọn phương pháp kiểm tra: Dựa vào loại dữ liệu và mục tiêu của nghiên cứu, chọn phương pháp kiểm tra thích hợp. Các phương pháp phổ biến bao gồm kiểm tra t, ANOVA, kiểm định chi-square, kiểm tra tương quan, vv.

Xác định mức ý nghĩa (alpha): Xác định mức ý nghĩa (alpha) cho kiểm tra. Mức ý nghĩa thường được đặt ở mức 0.05, tức là nếu giá trị p nhỏ hơn 0.05, chúng ta sẽ bác bỏ giả thuyết không.

Thu thập và phân tích dữ liệu: Thu thập dữ liệu và thực hiện phân tích thống kê thích hợp dựa trên phương pháp đã chọn.

Tính toán thống kê kiểm tra: Tính toán giá trị thống kê kiểm tra dựa trên dữ liệu và phương pháp đã chọn.

So sánh giá trị thống kê với ngưỡng alpha: So sánh giá trị thống kê tính được với mức ý nghĩa đã chọn. Nếu giá trị p (hoặc giá trị thống kê khác) nhỏ hơn mức ý nghĩa, chúng ta sẽ bác bỏ giả thuyết không và chấp nhận giả thuyết thay thế.

Rút ra kết luận: Dựa trên kết quả của kiểm tra, rút ra kết luận về giả thuyết được kiểm tra. Nếu có đủ bằng chứng thống kê để bác bỏ giả thuyết không, chúng ta có thể chấp nhận giả thuyết thay thế và kết luận dựa trên nó.

Báo cáo kết quả: Cuối cùng, báo cáo kết quả của kiểm tra giả thuyết trong bài báo cáo hoặc bài thuyết trình.

Quá trình kiểm tra giả thuyết trong phân tích thống kê chi tiết giúp cung cấp những chứng cứ thống kê để hỗ trợ quyết định và kết luận về mối quan hệ hoặc sự khác biệt giữa các nhóm trong dữ liệu.

Trong tổng quan, phân tích thống kê trong trực quan hóa dữ liệu cung cấp một cách tiếp cận toàn diện để hiểu và khám phá dữ liệu, từ việc mô tả dữ liệu đến việc phát triển và kiểm tra các mô hình dự đoán. Quá trình này cung cấp cái nhìn sâu sắc và ý nghĩa về dữ liệu, giúp người sử dụng đưa ra quyết định thông minh và định hình chiến lược.

Thống kê mô tả: Cho một biến số x1, x2, x3,…,xn chúng ta có thể tính toán một chỉ số thống kê mô tả như sau:

Lý thuyết	Hàm R
Số trung bình: $\overline{X} = \frac{1}{n}\sum_{i=1}^{n} x_i$	`mean(x)`
Phương sai: $s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{X})^2$	`var(x)`
Độ lệch chuẩn: $s = \sqrt{s^2}$	`sd(x)`
Trị số thấp nhất	`min(x)`
Trị số cao nhất	`max(x)`
Toàn cự (range)	`range(x)`

Bảng 1.3: Các hàm tính thống kê mô tả cơ bản R.

Ví dụ 1.19: Xét file dữ liệu Diem_TN, để tìm giá trị trung bình, phương sai của điểm toán (T) chúng ta dùng lệnh đơn lẻ hoặc lệnh tổng quan:

```
# lệnh đơn lẻ
mean(Diem_TN$T)
sd(Diem_TN$T)
# lệnh tổng quan
Sumary(Diem_TN$T)
# kết quả hiển thị
> mean
[1] 7.22
>Sd
[2] 0.8230345
> summary(T)
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
4.80	6.65	7.20	7.22	8.15	9.00

Trong kết quả trên, có hai chỉ số "1st Qu" và "3rd Qu" có nghĩa là first quartile (tương đương vị trí 25%) và third quartile (tương đương vị trí 75%) của một biến số . First quartile = 6.65 có nghĩa là 25% học sinh có điểm toán bằng hoặc thấp hơn 6.65. Tương tự Third quartile = 8.15 có nghĩa là 75% học sinh có điểm toán bằng hoặc thấp hơn 8.15. Trung vị (Median) = 7.2 có nghĩa là 50% học sinh có điểm toán 7,2 trở xuống (hay 7.2 trở lên).

CHƯƠNG 2

PHÂN TÍCH THĂM DÒ DỮ LIỆU

2.1. Thu thập dữ liệu

2.1.1. Nguồn dữ liệu

Dữ liệu được sử dụng trong đồ án này được lấy từ Công ty TNHH phân phối lân, đạm. Công ty cung cấp thông tin về giá cả hàng hóa và các xu hướng tiêu dùng của các loại sản phẩm mà họ phân phối.

2.1.2. Kỹ thuật thu thập dữ liệu

Phỏng vấn và khảo sát: Để thu thập thông tin chi tiết về giá cả và xu hướng tiêu dùng của các loại hàng hóa, chúng tôi đã tiến hành các cuộc phỏng vấn và khảo sát với các nhân viên kinh doanh, quản lý cửa hàng và khách hàng của Công ty TNHH.

Lấy số liệu từ bộ phận kế toán: Liên hệ với bộ phận kế toán của công ty để lấy dữ liệu cụ thể về giá cả và thông tin tài chính liên quan.

2.1.3. Làm sạch dữ liệu và xử lý các giá trị thiếu

Sử dụng gói library(tidyr): Áp dụng gói thư viện `tidyr` trong ngôn ngữ lập trình R để làm sạch và xử lý dữ liệu.

Các kỹ thuật thuộc gói library(tidyr) để làm sạch dữ liệu:

Gộp (Pivoting): Sử dụng các kỹ thuật gộp dữ liệu để biến đổi cấu trúc của dữ liệu sao cho phù hợp với mục tiêu phân tích.

Làm sạch dữ liệu (Data Cleaning): Tiến hành các bước làm sạch dữ liệu như loại bỏ giá trị trùng lặp, xử lý giá trị ngoại lai và giá trị thiếu.

2.1.4. Chuyển đổi dữ liệu phù hợp cho phân tích

Sau khi làm sạch dữ liệu, tiến hành đổi dữ liệu thành định dạng phù hợp cho quá trình phân tích. Các biến được định dạng lại và chuẩn hóa để đảm bảo tính nhất quán và đồng nhất trong quá trình phân tích dữ liệu.

Việc áp dụng những kỹ thuật này giúp tạo ra một tập dữ liệu chất lượng và có thể tin cậy để tiến hành phân tích và trực quan hóa dữ liệu về giá cả hàng hóa và xu hướng tiêu dùng của các loại hàng hóa.

2.2. Thăm dò dữ liệu

2.2.1. Đại lượng thống kê cơ bản

a. Summary(Tổng quan):

- Là một phương pháp nhanh chóng và thuận tiện để kiểm tra tổng quan về dữ liệu, giúp phát hiện và hiểu rõ hơn về đặc điểm cơ bản của các biến trong bảng dữ liệu.
    - `summary(hanghoa$soluongnhap)`

```
> summary(hanghoa$soluongnhap)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   0.00    0.00    0.00   17.76    0.50 5000.00
```

    - `summary(hanghoa$soluongxuat)`

```
> summary(hanghoa$soluongxuat)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   0.00    0.00    1.00   11.51    2.00 2000.00
```

    - `summary(hanghoa$soluongton)`

```
> summary(hanghoa$soluongton)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   0.00    2.00    6.00   45.76   17.00 5000.00
```

Nhận xét :

- Cách tiếp cận này giúp ta hiểu rõ hơn về phân phối và đặc điểm của các biến quan trọng trong dữ liệu.
- Bằng cách sử dụng summary(), ta có thể nhanh chóng kiểm tra các thông tin tổng quan và phát hiện ra các giá trị ngoại lệ hoặc các vấn đề khác trong dữ liệu.

b. Trung bình (Mean):

- Trung bình là một phép đo trung tâm phổ biến, cho biết giá trị trung bình của một biến trong tập dữ liệu.Trong thống kê, việc tính toán trung bình giúp đại diện cho dữ liệu. Nó cho biết giá trị trung bình dự kiến của biến và cung cấp một ý nghĩa về trung tâm của phân phối dữ liệu.

```
mean(hanghoa$soluongxuat)
mean(hanghoa$soluongnhap)
mean(hanghoa$soluongton)
```

Kết quả:

```
> mean(hanghoa$soluongnhap)
[1] 17.76327
> mean(hanghoa$soluongxuat)
[1] 11.50979
> mean(hanghoa$soluongton)
[1] 45.7555
```

Nhận xét:

- Các giá trị trung bình này cung cấp một cái nhìn tổng quan về mức độ trung bình của số lượng xuất, số lượng nhập và số lượng tồn trong dữ liệu. Điều này có thể hữu ích để hiểu rõ hơn về khối lượng công việc hoặc lưu lượng hàng hóa trong một khoản thời gian nhất định.

c. Phương sai (Variance):

- Phương sai là một đại lượng thống kê dùng để đo độ biến đổi của dữ liệu từ trung bình. Phương sai thể hiện mức độ phân tán của dữ liệu từ trung bình. Nó càng lớn thì dữ liệu càng phân tán và ngược lại. Phương sai cho biết mức độ biến động của dữ liệu và có thể được sử dụng để so sánh sự biến động giữa các tập dữ liệu khác nhau.

```
var(hanghoa$soluongxuat)
var(hanghoa$soluongnhap)
var(hanghoa$soluongton)
```

Kết quả:

```
> var(hanghoa$soluongnhap)
[1] 80872.89
> var(hanghoa$soluongxuat)
[1] 16233.9
> var(hanghoa$soluongton)
[1] 144217
```

Nhận xét:

- Phương sai là một đại lượng thống kê đo lường mức độ biến đổi của dữ liệu từ giá trị trung bình. Nó được tính bằng cách lấy trung bình của bình phương khoảng cách giữa mỗi giá trị và giá trị trung bình của dữ liệu. Phương sai cung cấp một cách để đánh giá mức độ phân tán của dữ liệu, trong đó giá trị càng cao thì dữ liệu càng phân tán rộng.

d. Độ lệch chuẩn (Standard Deviation):

- Độ lệch chuẩn là căn bậc hai của phương sai. Nó đo lường mức độ biến động hoặc sự lan truyền của các giá trị trong tập dữ liệu so với trung bình.Độ lệch chuẩn cung cấp một đại diện cho mức độ biến động của dữ liệu một cách dễ hiểu hơn so với phương sai, vì nó có cùng đơn vị với dữ liệu gốc. Độ lệch chuẩn được sử dụng rộng rãi trong thống kê để đo lường sự biến động của dữ liệu và đánh giá rủi ro.

```
sd(hanghoa$soluongxuat)
sd(hanghoa$soluongnhap)
sd(hanghoa$soluongton)
```

Kết quả:

```
> sd(hanghoa$soluongnhap)
[1] 284.3816
> sd(hanghoa$soluongxuat)
[1] 127.4123
> sd(hanghoa$soluongton)
[1] 379.7592
```

Nhận xét :

- Độ lệch chuẩn là một phép đo về mức độ biến động của dữ liệu, nó cho biết mức độ mà các giá trị trong biến phân tán xung quanh giá trị trung bình. Điều này giúp hiểu rõ hơn về biến động của dữ liệu và sự ổn định của quá trình sản xuất hoặc giao dịch. Độ lệch chuẩn càng lớn thì dữ liệu càng phân tán rộng và ngược lại.

2.2.2. Thăm dò dữ liệu

a. Chia nhóm dữ liệu theo từng quý

```
# Tạo cột 'Quarter' để xác định quý của mỗi ngày hạch toán
library(dplyr)
hanghoa$Quarter <- quarters(hanghoa$ngayhachtoan)
# Tách dữ liệu thành 4 quý
quy1 <- hanghoa %>% filter(Quarter == "Q1")
#%>% được sử dụng để chuyển đối tượng hanghoa (dataframe) từ một hàm filter sang hàm khác mà không cần phải tham chiếu đến hanghoa nhiều lần
quy2 <- hanghoa %>% filter(Quarter == "Q2")
quy3 <- hanghoa %>% filter(Quarter == "Q3")
quy4 <- hanghoa %>% filter(Quarter == "Q4")
```

Đoạn code trên sử dụng gói dplyr trong R để phân tích dữ liệu"hanghoa" thành 4 quý dựa trên thông tin về ngày hạch toán của hàng hóa. Sau đó, nó in ra dữ liệu của từng quý để phân tích và đưa ra nhận xét.

- *Dữ liệu của Quý 1 (quy1):*
- *Lọc ra tất cả các hàng hóa thuộc quý 1 và lưu vào biến quy1.*

Sau đó, nó in ra dữ liệu của quý 1 bằng lệnh print(quy1).

	mahang	tenhang	ngayhachtoan	sochungtu	dongia	soluongnhap	giatrinhap	soluongxuat
	<chr>	<chr>	<dttm>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>
1	BB 331	BB 331- T…	2023-01-11 00:00:00	XK.00916	1.34e7	0	0	0
2	BB 331	BB 331- T…	2023-01-11 00:00:00	XK.00916	1.34e7	0	0	1.5
3	BB 331 SUPER	BB 331 SU…	2023-03-10 00:00:00	NK.00348	1.38e7	3	41496000	0
4	BB 331 SUPER	BB 331 SU…	2023-03-31 00:00:00	XK.01070	1.38e7	0	0	1.5
5	BB 332	BB-332 Th…	2023-03-10 00:00:00	NK.00348	1.12e7	3	33486000	0
6	BB 332	BB-332 Th…	2023-03-23 00:00:00	NK.00366	1.12e7	1.68	18696350	0
7	BB 333	BB 333 TH…	2023-01-11 00:00:00	XK.00917	1.04e7	0	0	1
8	BB 631	BB 631 TH…	2023-03-10 00:00:00	NK.00348	1.21e7	5	60440000	0
9	BB 632	BB 632 TH…	2023-03-10 00:00:00	NK.00348	1.15e7	4.15	47563150	0
10	BB 686	BB 686 Th…	2023-01-11 00:00:00	XK.00916	1.05e7	0	0	3

i 31 more rows
i 5 more variables: giatrixuat <dbl>, soluongton <dbl>, giatriton <dbl>, dongiaban <dbl>,

- *Dữ liệu của Quý 2 (quy2):*

Tương tự, lọc ra tất cả các hàng hóa thuộc quý 2 và lưu vào biến quy2.
In ra dữ liệu của quý 2 bằng lệnh print(quy2).

	mahang	tenhang	ngayhachtoan	sochungtu	dongia	soluongnhap	giatrinhap	soluongxuat
	<chr>	<chr>	<dttm>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>
1	BB 331 SUPER	BB 331 SU…	2023-04-01 00:00:00	XK.01075	1.38e7	0	0	1.5
2	BB 332	BB-332 Th…	2023-04-05 00:00:00	XK.01081	1.12e7	0	0	1.5
3	BB 332	BB-332 Th…	2023-04-12 00:00:00	XK.01089	1.12e7	0	0	1.5
4	BB 332	BB-332 Th…	2023-04-25 00:00:00	XK.01107	1.12e7	0	0	1.68
5	BB 333	BB 333 TH…	2023-04-25 00:00:00	XK.01107	1.04e7	0	0	0
6	BB 631	BB 631 TH…	2023-04-01 00:00:00	XK.01075	1.21e7	0	0	1.5
7	BB 631	BB 631 TH…	2023-04-05 00:00:00	XK.01081	1.21e7	0	0	1.5
8	BB 631	BB 631 TH…	2023-04-12 00:00:00	XK.01089	1.21e7	0	0	2
9	BB 632	BB 632 TH…	2023-04-05 00:00:00	XK.01081	1.15e7	0	0	1
10	BB 632	BB 632 TH…	2023-04-18 00:00:00	XK.01097	1.15e7	0	0	3.15

i 25 more rows
i 5 more variables: giatrixuat <dbl>, soluongton <dbl>, giatriton <dbl>, dongiaban <dbl>,

- *Dữ liệu của Quý 3 (quy3)*

Lọc và lưu dữ liệu của quý 3 vào biến quy3.
In ra dữ liệu của quý 3.

	mahang	tenhang	ngayhachtoan	sochungtu	dongia	soluongnhap	giatrinhap	soluongxuat
	<chr>	<chr>	<dttm>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>
1	BM 12.7.20	Phân bón N…	2023-09-29 00:00:00	NK.00475	1.4 e7	3	42000000	0
2	BM 15.15.15	Phân bón h…	2023-09-29 00:00:00	NK.00475	2.15e7	1	21500000	0
3	BM 15.9.15	Phân bón N…	2023-09-29 00:00:00	NK.00475	1.4 e7	3.28	45850000	0
4	BM 16.16.16	Phân bón h…	2023-09-29 00:00:00	NK.00475	1.53e7	4.2	64260000	0
5	BM 16.16.8	Phân bón h…	2023-09-29 00:00:00	NK.00475	1.35e7	3	40500000	0
6	BM 16-16-8	Phân NPK B…	2023-09-29 00:00:00	NK.00476	1.54e7	2	30700000	0
7	BM 20.10.10	Phân bón c…	2023-09-29 00:00:00	NK.00476	2.55e7	0.5	12750000	0
8	BM 20.10.8	Phân bón M…	2023-09-29 00:00:00	NK.00475	1.38e7	5	69000000	0
9	BM đạm	Phân Ammon…	2023-09-29 00:00:00	NK.00476	5 e6	1	5000000	0
10	BM kali	Phân Kali …	2023-09-29 00:00:00	NK.00476	1.20e7	2	24000000	0

i 77 more rows
i 5 more variables: giatrixuat <dbl>, soluongton <dbl>, giatriton <dbl>, dongiaban <dbl>,

- *Dữ liệu của Quý 4 (quy4):*

Lọc và lưu dữ liệu của quý 4 vào biến quy4.
In ra dữ liệu của quý 4.

Kết quả:

	mahang	tenhang	ngayhachtoan	sochungtu	dongia	soluongnhap	giatrinhap	soluongxuat
	<chr>	<chr>	<dttm>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>
1	BM 15.15.15	Phân bón h…	2023-12-27 00:00:00	XK.01712	2.15e7	0	0	0.2
2	BM 15.5.20	Phân bón N…	2023-12-20 00:00:00	NK.00521	2.39e7	2	47776000	0
3	BM 15.9.15	Phân bón N…	2023-11-30 00:00:00	XK.01629	1.4 e7	0	0	0.5
4	BM 15.9.15	Phân bón N…	2023-12-20 00:00:00	NK.00521	1.32e7	10	131680000	0
5	BM 16.16.16	Phân bón h…	2023-11-20 00:00:00	XK.01588	1.53e7	0	0	1
6	BM 16.16.16	Phân bón h…	2023-11-25 00:00:00	XK.01610	1.53e7	0	0	0.5
7	BM 16.16.16	Phân bón h…	2023-12-19 00:00:00	XK.01695	1.53e7	0	0	0.5
8	BM 16.16.16	Phân bón h…	2023-12-21 00:00:00	XK.01700	1.53e7	0	0	0.5
9	BM 16.16.8	Phân bón h…	2023-10-10 00:00:00	XK.01485	1.35e7	0	0	1
10	BM 16.16.8	Phân bón h…	2023-10-24 00:00:00	XK.01513	1.35e7	0	0	1.5

```
# ℹ 136 more rows
# ℹ 5 more variables: giatrixuat <dbl>, soluongton <dbl>, giatriton <dbl>, dongiaban <dbl>,
#   Quarter <chr>
```

Nhận xét:

- Quá trình phân tích dữ liệu thành từng quý giúp chúng ta hiểu rõ hơn về mô hình hoạt động kinh doanh theo mùa trong năm
- Việc phân chia dữ liệu thành các quý giúp đối chiếu và so sánh hiệu suất kinh doanh, lưu lượng giao dịch, hoặc các chỉ số khác giữa các quý.
- Điều này có thể giúp cho việc đưa ra các chiến lược kinh doanh hiệu quả dựa trên sự hiểu biết về cách mà hoạt động kinh doanh biến động theo từng mùa.

b. Số lượng xuất - nhập - tồn theo từng quý

```
# Tính tổng số lượng nhập, số lượng xuất và số lượng tồn cho mỗi quý
library(ggplot2)
summary_quy <- hanghoa %>%
  group_by(Quarter) %>%
  summarise(soluongnhap = sum(soluongnhap),
            soluongxuat = sum(soluongxuat),
            soluongton = sum(soluongton))
```

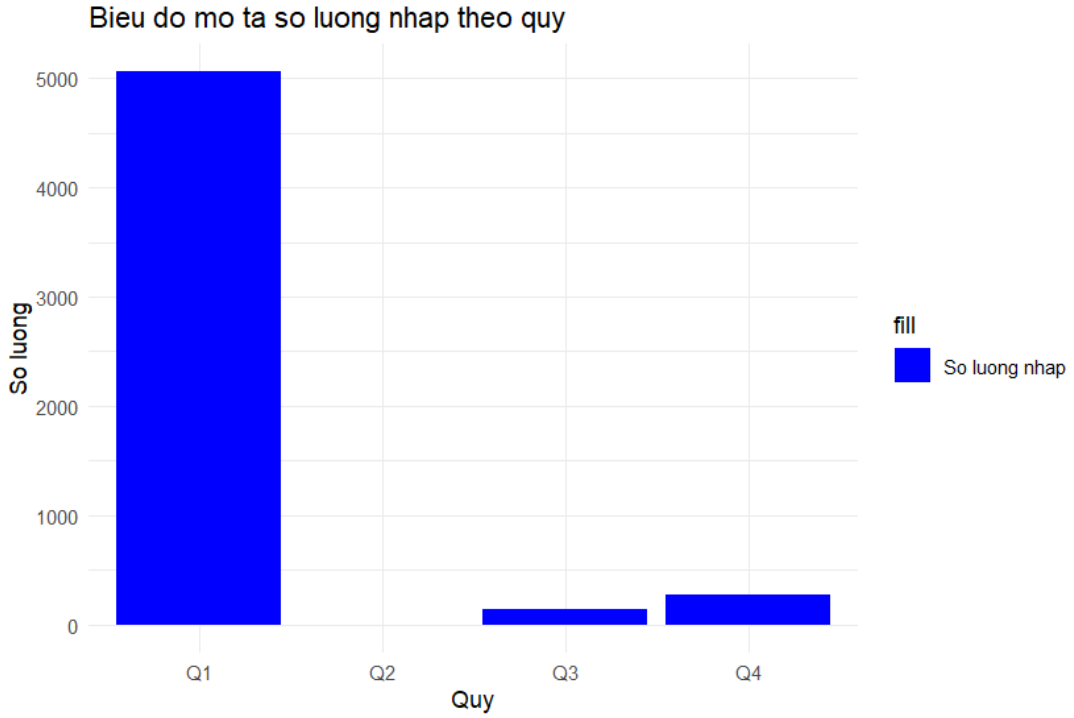
Sử dụng thư viện ggplot2 để tạo ra biểu đồ trực quan mô tả số lượng nhập, số lượng xuất và số lượng tồn theo từng quý. Trước tiên, nó tổng hợp dữ liệu theo từng quý bằng cách sử dụng hàm summarise() để tính tổng số lượng nhập, số lượng xuất và số lượng tồn trong mỗi quý.

Tạo biểu đồ

```
ggplot(summary_quy, aes(x = Quarter)) +
  geom_col(aes(y = soluongnhap, fill = "So luong nhap"), position = "dodge")
+
  labs(title = "Bieu do mo ta so luong xuat-nhap-ton theo quy",
       x = "Quy",
       y = "So luong") +
```

```
  scale_fill_manual(values = c("So luong nhap" = "blue"))
```

Kết quả:



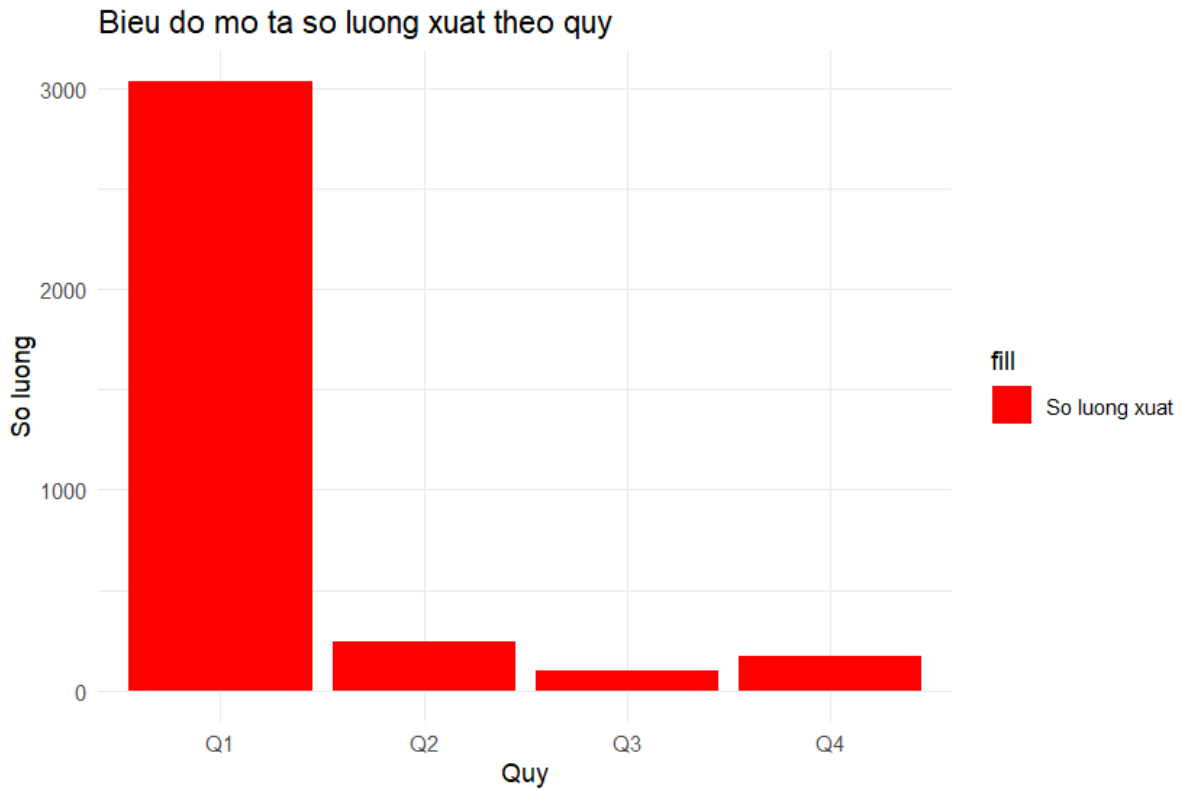
Hình 2.1. Biểu đồ mô tả số lượng nhập theo quý.

Nhận xét:

- Biểu đồ này cho thấy sự biến động của số lượng hàng hóa được nhập vào qua từng quý trong năm.
- Có thể thấy rằng có sự biến động đáng kể giữa các quý. Ví dụ, quý 1 có số lượng nhập cao nhất, sau đó giảm dần qua các quý tiếp theo, và có thể tăng trở lại vào cuối năm.
- Sự biến động này có thể phản ánh các yếu tố mùa vụ hoặc chiến lược kinh doanh cụ thể trong từng quý.

```
ggplot(summary_quy, aes(x = Quarter)) +
  geom_col(aes(y = soluongxuat, fill = "So luong xuat"), position = "dodge")
+
  labs(title = "Bieu do mo ta so luong xuat-nhap-ton theo quy",
       x = "Quy",
       y = "So luong") +
  scale_fill_manual(values = c("So luong xuat" = "red"))
```

***Kết quả*:**



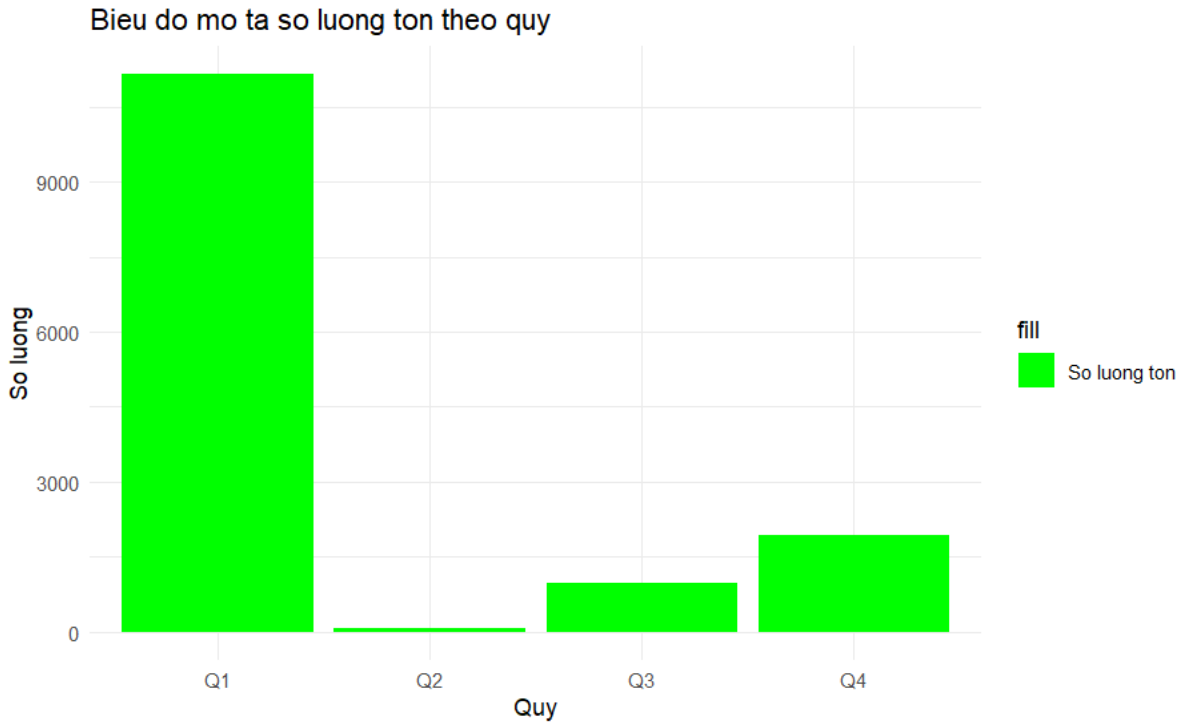
Hình 2.2. Biểu đồ mô tả số lượng xuất theo quý.

Nhận xét:

- Biểu đồ này cho thấy sự biến động của số lượng hàng hóa được xuất ra thị trường qua từng quý trong năm.
- Tương tự như biểu đồ số lượng nhập, có sự biến động đáng kể giữa các quý. Có thể thấy sự tăng giảm không đồng đều của số lượng xuất qua từng quý.
- Sự biến động này có thể phản ánh các yếu tố thị trường, cung cầu, hoặc chiến lược tiếp thị.
-

```
# Biểu đồ mô tả số lượng tồn theo quý
ggplot(summary_quy, aes(x = Quarter)) +
  geom_col(aes(y = soluongton, fill = "So luong ton"), position = "dodge") +
  labs(title = "Bieu do mo ta so luong xuat-nhap-ton theo quy",
       x = "Quy",
       y = "So luong") +
  scale_fill_manual(values = c( "So luong ton" = "green"))
```

Kết quả :



Hình 2.3. Biểu đồ mô tả số lượng tồn theo quý.

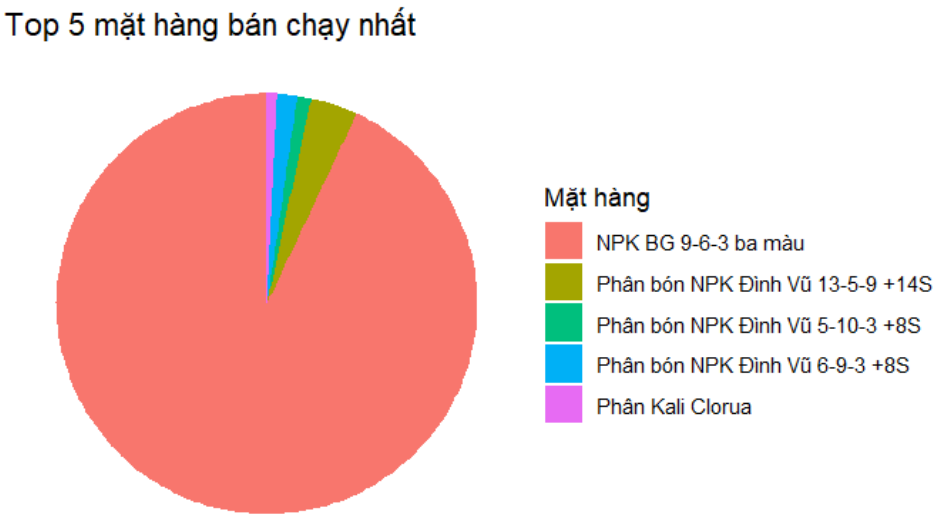
Nhận xét:

- Biểu đồ này biểu diễn sự biến động của số lượng hàng hóa tồn kho trong từng quý trong năm.
- Số lượng tồn thường ổn định hơn so với số lượng nhập và số lượng xuất, nhưng vẫn có sự biến động nhất định giữa các quý.
- Sự biến động này có thể phản ánh hiệu suất quản lý tồn kho, sự cân nhắc giữa cung và cầu, hoặc chiến lược lưu trữ hàng hóa.

2.2.3. Sử dụng biểu đồ tròn phân tích các mặt hàng bán chạy

```
# Sử dụng biểu đồ tròn (pie chart) để so sánh các mặt hàng
#bán chạy nhất và doanh thu cao nhất
summary_hanghoa <- hanghoa %>%
  group_by(mahang, tenhang) %>%
  summarise(tong_soluong_ban = sum(soluongxuat),
            tong_doanhthu = sum(dongiaban * soluongxuat))
# Chọn ra 5 mặt hàng bán chạy nhất
top5_banchay <- summary_hanghoa %>%
  arrange(desc(tong_soluong_ban)) %>%
  head(5)

# Biểu đồ tròn cho mặt hàng bán chạy nhất
```

```
ggplot(top5_banchay, aes(x = "", y = tong_soluong_ban, fill = tenhang)) +
  geom_bar(stat = "identity", width = 1) +
  coord_polar("y", start = 0) +
  labs(title = "Top 5 mặt hàng bán chạy nhất",
       fill = "Mặt hàng",
       y = "Tổng số lượng bán") +
  theme_void() +
  theme(legend.position = "right")
```



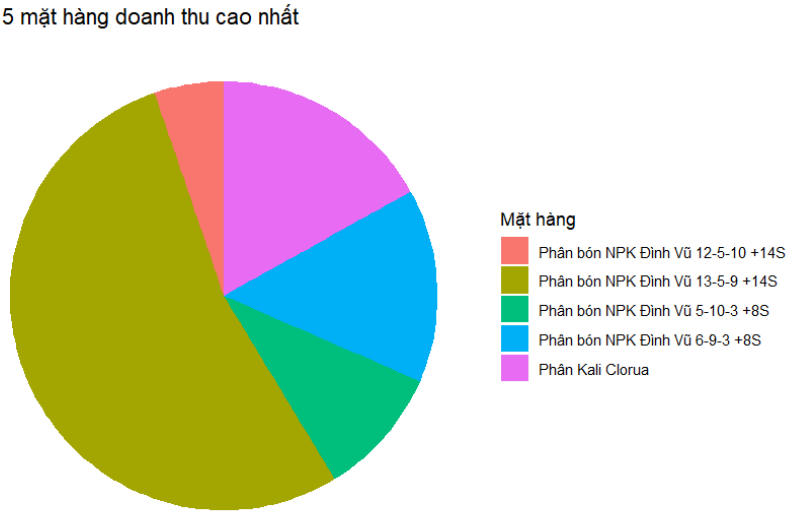
Hình 2.4. Biểu đồ mô tả top 5 mặt hàng bán chạy nhất.

Nhận xét:

- Biểu đồ tròn này hiển thị top 5 mặt hàng bán chạy nhất, với diện tích của mỗi cột tròn tương ứng với tổng số lượng bán của mặt hàng đó.
- Mỗi mặt hàng được phân biệt bằng màu sắc khác nhau, được hiển thị trong hình chú thích (legend) ở phía bên phải của biểu đồ.
- Tiêu đề và nhãn trục được đặt để giúp người đọc hiểu rõ hơn về nội dung của biểu đồ.

```
# Chọn ra 5 mặt hàng doanh thu cao nhất
top5_doanhthu <- summary_hanghoa %>%
  arrange(desc(tong_doanhthu)) %>%
  head(5)
# Biểu đồ tròn cho mặt hàng doanh thu cao nhất
ggplot(top5_doanhthu, aes(x = "", y = tong_doanhthu, fill = tenhang)) +
```

```
geom_bar(stat = "identity", width = 1) +
coord_polar("y", start = 0) +
labs(title = "Top 5 mặt hàng doanh thu cao nhất",
     fill = "Mặt hàng",
     y = "Tổng doanh thu") +
theme_void() +
theme(legend.position = "right")
```



Hình 2.5. Biểu đồ mô tả top 5 mặt hàng doanh thu cao nhất.

Nhận xét :

- Được sử dụng để chọn ra và thể hiện top 5 mặt hàng có doanh thu cao nhất từ một tập dữ liệu dự liệu hàng hóa (summary_hanghoa).
- Đoạn mã này giúp trả lời câu hỏi như "Các mặt hàng nào đang đóng góp nhiều nhất vào doanh thu của công ty?" hay "Những mặt hàng nào cần được chú ý để tối ưu hóa chiến lược kinh doanh?".
- Biểu đồ này giúp người dùng dễ dàng nhìn ra mức độ quan trọng của từng mặt hàng đối với doanh thu tổng cộng của công ty.
- Biểu đồ tròn được sử dụng để trực quan hóa top 5 mặt hàng có doanh thu cao nhất.

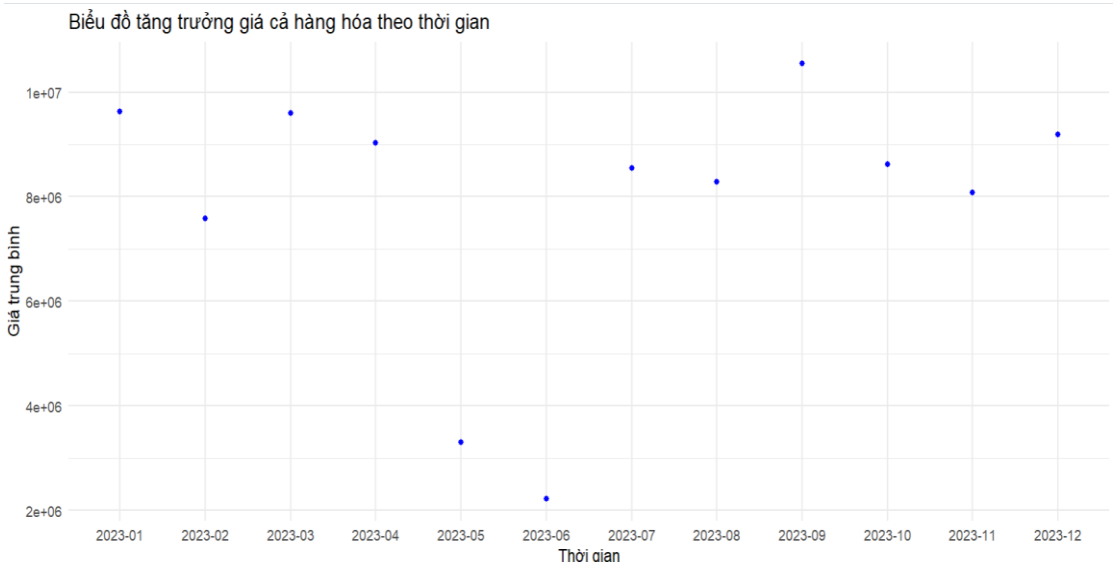
2.2.4. Sử dụng biểu đồ tăng trưởng (Growth chart) để phân tích giá cả hàng hóa

```
# Sử dụng biểu đồ tăng trưởng(Growth Chart) để phân tích giá cả hàng hóa
hanghoa$month_year <- format(hanghoa$ngayhachtoan, "%Y-%m")
# Tính giá trung bình hàng hóa theo tháng
mean_price <- hanghoa %>%
  group_by(month_year) %>%
  summarise(mean_dongia = mean(dongia))
```

Sử dụng hàm group by() để nhóm dữ liệu theo cột "month_year" và summarise để tính giá trung bình của cột "dongia" cho mỗi nhóm thời gian . Kết quả là tạo thêm 1 cột mới "month_year" và 1 dataframe mới là "mean_price".

```
Growth chart cho mean_price# Tạo biểu đồ tăng trưởng
ggplot(mean_price, aes(x = month_year, y = mean_dongia)) +
  geom_line(color = "blue") +
  geom_point(color = "blue") +
  labs(title = "Biểu đồ tăng trưởng giá cả hàng hóa theo thời gian",
       x = "Thời gian",
       y = "Giá trung bình") +
  theme_minimal()
```

Kết quả:



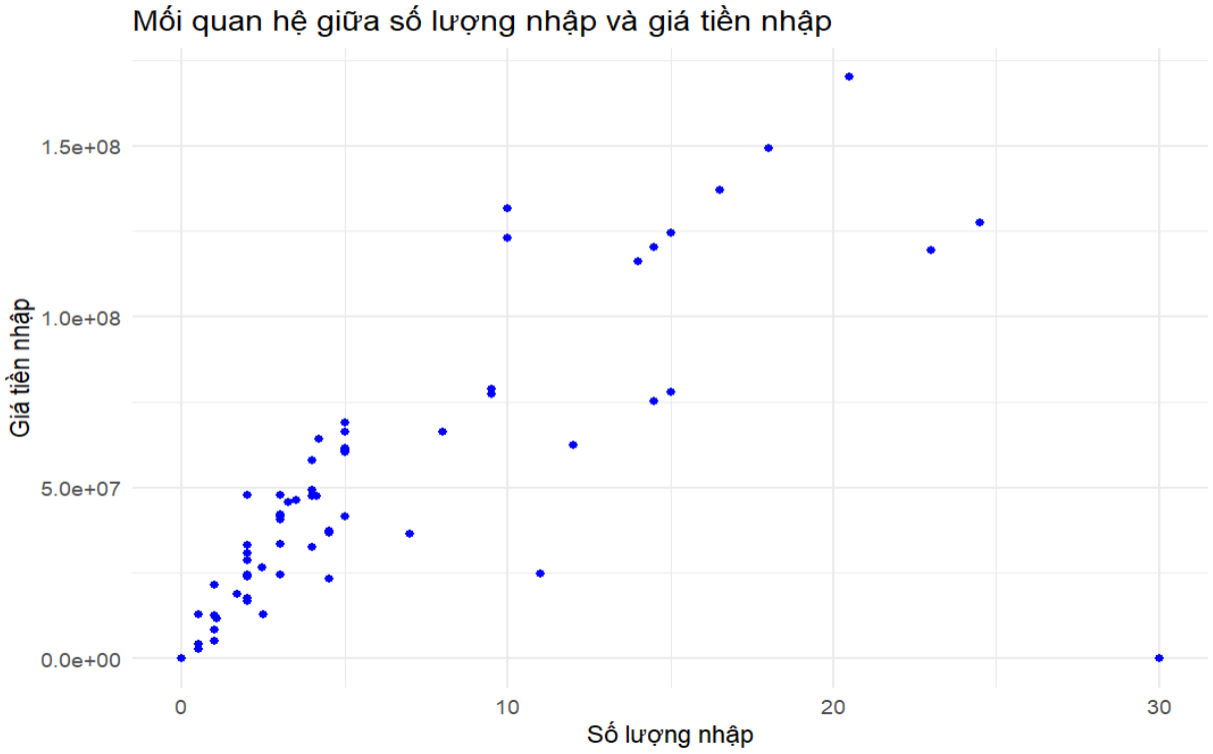
Hình 2.6. Biểu đồ tăng trưởng giá cả hàng hóa theo thời gian.

Nhận xét:

- Biểu đồ này giúp trực quan hóa xu hướng tăng giảm của giá trung bình hàng hóa theo thời gian.
- Nếu đường trendline (đường màu xanh) có xu hướng tăng, điều này cho thấy giá hàng hóa đang tăng dần theo thời gian. Ngược lại, nếu có xu hướng giảm, điều này cho thấy giá hàng hóa đang giảm dần.
- Các điểm dữ liệu được thêm vào biểu đồ giúp nhận biết giá trung bình cụ thể của từng tháng.
- Biểu đồ này có thể được sử dụng để phân tích xu hướng giá cả hàng hóa trong một khoảng thời gian nhất định và đưa ra dự đoán về xu hướng tương lai.

```
# Biểu đồ Scatter Plot mối quan hệ giữa số lượng nhập và giá tiền nhập
ggplot(hanghoa, aes(x = soluongnhap, y = giatrinhap)) +
  geom_point(color = "blue") +
  labs(title = "Mối quan hệ giữa số lượng nhập và giá tiền nhập",
       x = "Số lượng nhập",
       y = "Giá tiền nhập") +
  theme_minimal()
```

Kết quả:



Hình 2.7. Mối quan hệ giữa số lượng nhập và giá trị nhập.

Nhận xét:

- Nếu các điểm trên biểu đồ phân bố đều và không có một hình dạng hay xu hướng rõ ràng, điều này có thể cho thấy không có mối quan hệ tuyến tính giữa số lượng nhập và giá tiền nhập.
- Nếu có một xu hướng tăng dần hoặc giảm dần rõ ràng, bạn có thể kỳ vọng một mối quan hệ tương quan giữa hai biến.
- Biểu đồ này có thể được sử dụng để phát hiện các điểm ngoại lệ (outliers) hoặc để đưa ra dự đoán về giá tiền nhập dựa trên số lượng nhập.

2.2.5. Sử dụng ước lượng tỉ lệ để so sánh được sản phẩm bán chạy nhất

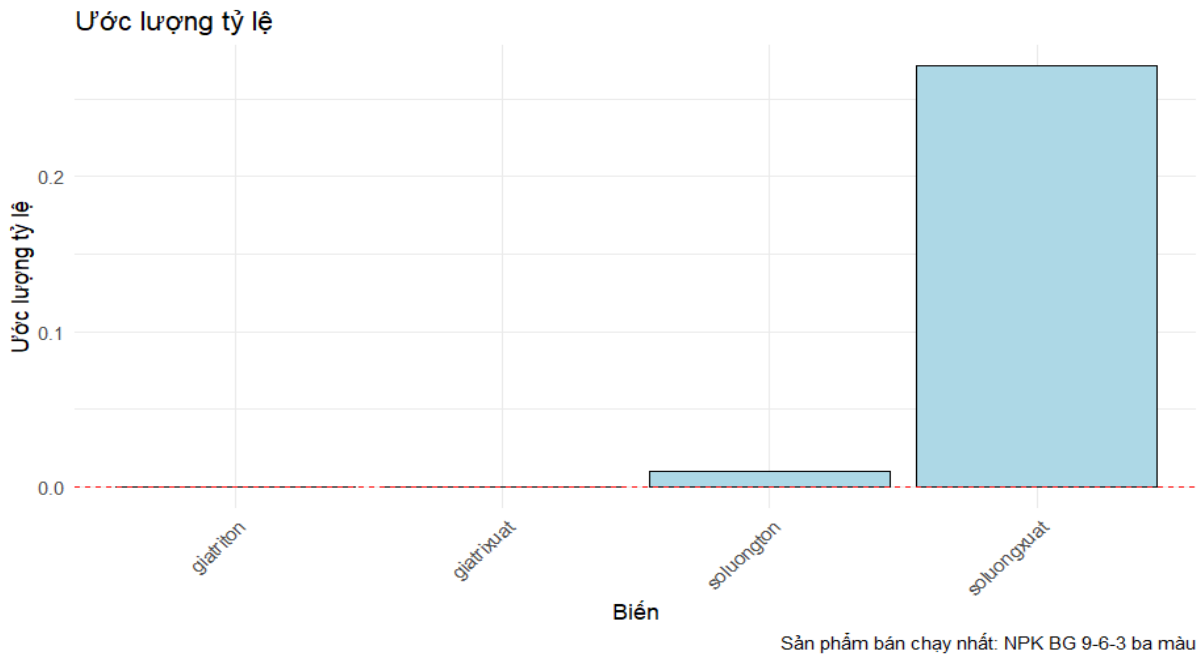
```
# Sắp xếp theo thứ tự giảm dần của số lượng bán
summary_hanghoa <- summary_hanghoa %>% arrange(desc(total_soluong))

# Chọn ra sản phẩm bán chạy nhất
top_selling_product <- summary_hanghoa$tenhang[1]

# Tạo biến nhị phân cho sản phẩm bán chạy nhất
hanghoa <- hanghoa %>%
  mutate(is_top_selling = as.numeric(tenhang == top_selling_product))

# Tính toán tỷ lệ bán chạy nhất
model <- glm(is_top_selling ~ soluongxuat + giatrixuat + soluongton + giatriton,
data = hanghoa, family = "binomial")

# Biểu đồ cột tỷ lệ ước lượng
coefficients_df <- as.data.frame(summary(model)$coefficients)
coefficients_df$variable <- rownames(coefficients_df)
```

- Đoạn mã sử dụng dữ liệu từ hanghoa và nhóm các mẫu theo tên sản phẩm (tenhang), sau đó tính tổng số lượng bán (soluongxuat) cho mỗi sản phẩm.
- Dữ liệu trong summary_hanghoa được sắp xếp theo thứ tự giảm dần của tổng số lượng bán.
- Sản phẩm bán chạy nhất được chọn dựa trên sản phẩm có tổng số lượng bán cao nhất từ summary_hanghoa.
- Một biến nhị phân is_top_selling được tạo ra để đánh dấu các mẫu dữ liệu của sản phẩm bán chạy nhất.
- Mô hình logistic regression được xây dựng để ước lượng tỷ lệ của biến phụ thuộc is_top_selling dựa trên các biến độc lập (soluongxuat, giatrixuat, soluongton, giatriton).
- Mô hình logistic regression được sử dụng để đánh giá mối quan hệ giữa các biến độc lập và xác suất sản phẩm được chọn là sản phẩm bán chạy nhất.

Biểu đồ cột hiển thị ước lượng tỉ lệ của các biến độc lập đối với biến độc lập đối với biến phụ thuộc trong mô hình logistic regression.

```
# Biểu đồ cột
ggplot(coefficients_df[-1, ], aes(x = variable, y = Estimate)) +
  geom_col(fill = "lightblue", color = "black") +
  geom_hline(yintercept = 0, color = "red", linetype = "dashed") +
  labs(title = "Ước lượng tỷ lệ",
       x = "Biến",
       y = "Ước lượng tỷ lệ",
       caption = paste("Sản phẩm bán chạy nhất:", top_selling_product),
       fill = "Biến") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

Kết quả:



Hình 2.8.Biểu đồ ước lượng tỷ lệ.

Nhận xét:

- Các cột có chiều cao dương (trên đường ngang màu đỏ) cho thấy mức độ tăng của xác suất của biến phụ thuộc khi giá trị của biến độc lập tăng.
- Các cột có chiều cao âm (dưới đường ngang màu đỏ) cho thấy mức độ giảm của xác suất của biến phụ thuộc khi giá trị của biến độc lập tăng.
- Biểu đồ này giúp hiểu rõ hơn về tác động của mỗi biến độc lập lên xác suất của biến phụ thuộc, và xác định được biến nào có ảnh hưởng lớn nhất đối với biến phụ thuộc.

2.3. Phân tích mối liên hệ tương quan

2.3.1. Biểu đồ tương quan

a. Phân tích mối liên hệ giữa các biến:

Biến nhập - xuất - tồn:

- Biến nhập (Inventory In): Đây là lượng hàng hoá hoặc dịch vụ được nhập vào hệ thống hoặc tồn kho trong một khoảng thời gian cụ thể. Biến này đại diện cho sự nhập hàng hóa hoặc dịch vụ vào doanh nghiệp.

- Biến xuất (Inventory Out): Đây là lượng hàng hoá hoặc dịch vụ được xuất ra khỏi hệ thống hoặc tồn kho trong một khoảng thời gian cụ thể. Biến này thường đại diện cho doanh số bán hàng hoặc dịch vụ.
- Biến tồn (Inventory On-hand): Đây là lượng hàng hoá hoặc dịch vụ vẫn còn lại trong hệ thống hoặc tồn kho sau khi đã trừ đi lượng hàng hoá đã bán hoặc sử dụng. Biến này cho biết mức độ tồn kho hiện tại.
- Nếu có một mẫu hay mối quan hệ tuyến tính giữa hai biến, các điểm trên biểu đồ scatter plot sắp xếp thành một hình dạng gần như thẳng hàng.
- Mối liên hệ giữa các biến xuất, nhập, tồn được phân tích như sau:
- Tương quan dương giữa biến nhập và biến tồn, tương quan dương giữa biến xuất và biến tồn, tương quan ngược giữa biến nhập và biến xuất, mối quan hệ không tuyến tính hoặc không đồng đều. Có thể phân tích kĩ hơn về mối liên hệ giữa các biến này cung cấp thông tin quan trọng để hiểu và cải thiện quản lý tồn kho, dự báo nhu cầu thị trường và tối ưu hóa hoạt động kinh doanh.

```
# Chọn các biến liên quan đến xuất, nhập và tồn từ dữ liệu hàng hóa
variables_to_analyze <- c("soluongxuat", "giatrixuat", "soluongnhap", "giatrinhap",
"soluongton", "giatriton")

# Tính toán ma trận tương quan giữa các biến
correlation_matrix <- cor(hanghoa[, variables_to_analyze])

# In ra ma trận tương quan
print("Ma trận tương quan:\n")
print(correlation_matrix)
```

Nhận xét:
Có ít nhất một cặp biến có mức độ tương quan cao (> 0.8). Điều này có thể chỉ ra sự phụ thuộc mạnh mẽ giữa các biến này, và cần được xem xét khi phân tích dữ liệu và đưa ra dự đoán.

Mối quan hệ giữa số lượng xuất và các biến khác:

- Số lượng xuất có mối tương quan dương với số lượng tồn, với hệ số tương quan là khoảng 0.39. Điều này có nghĩa là khi số lượng xuất tăng, số lượng tồn cũng có xu hướng tăng, và ngược lại.
- Số lượng xuất có mối tương quan tiêu cực với giá trị nhập và giá trị tồn. Điều này có thể được giải thích bởi việc khi số lượng xuất tăng, giá trị nhập và giá trị tồn có thể giảm do lượng hàng tồn kho giảm đi.

Mối quan hệ giữa giá trị xuất và các biến khác:

- Giá trị xuất không có mối tương quan đáng kể với số lượng nhập và số lượng tồn.

- Giá trị xuất có mối tương quan dương với giá trị nhập và giá trị tồn. Điều này có thể được hiểu là khi giá trị xuất tăng, giá trị nhập và giá trị tồn cũng có xu hướng tăng.

Mối quan hệ giữa số lượng nhập và các biến khác:

- Số lượng nhập có mối tương quan dương mạnh với giá trị nhập (khoảng 0.85) và giá trị tồn (khoảng 0.85). Điều này cho thấy mối quan hệ mạnh mẽ giữa số lượng hàng nhập và giá trị của chúng.
- Số lượng nhập không có mối tương quan đáng kể với số lượng xuất.

Mối quan hệ giữa giá trị nhập và các biến khác:

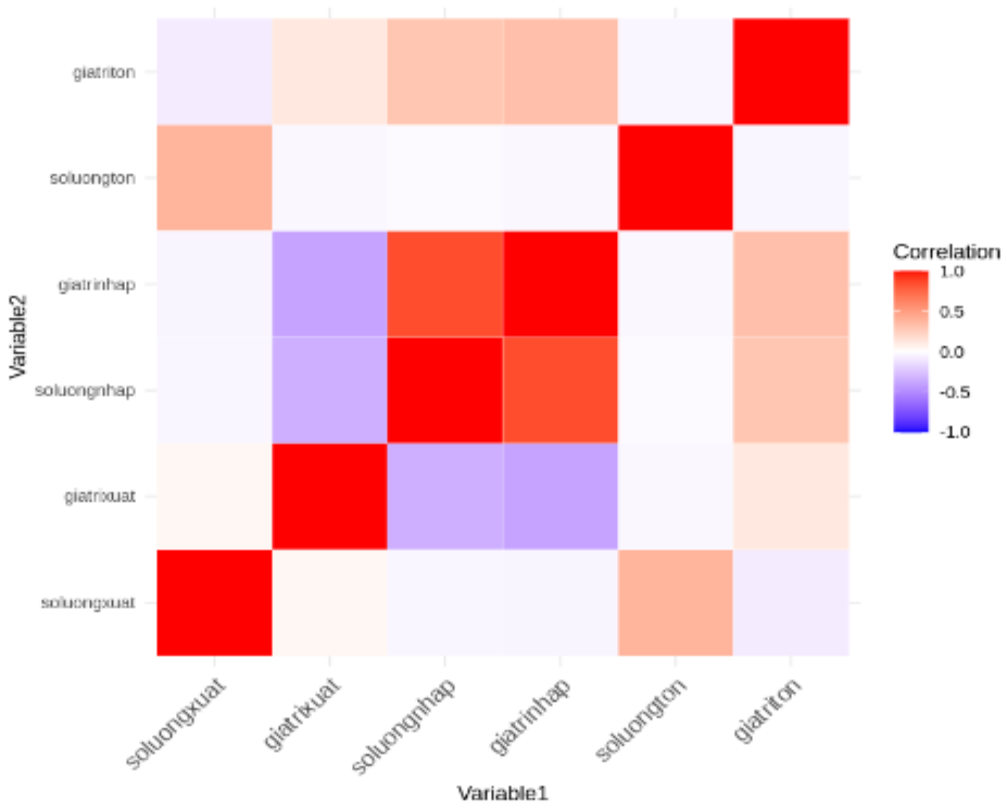
- Giá trị nhập có mối tương quan mạnh với số lượng nhập và giá trị tồn. Điều này chỉ ra mối quan hệ mạnh mẽ giữa giá trị của hàng nhập và số lượng hoặc giá trị tồn kho.

Mối quan hệ giữa số lượng tồn và giá trị tồn:

- Số lượng tồn và giá trị tồn có mối tương quan dương nhưng không mạnh (khoảng 0.39).

b. Từ file dữ liệu hanghoa.xlsx từ các mục trên, chúng ta lập ma trận biểu đồ tương quan.

```
library(ggplot2)
# Chọn các biến liên quan đến xuất, nhập và tồn từ dữ liệu hàng hóa
variables_to_analyze <- c("soluongxuat", "giatrixuat", "soluongnhap",
"giatrinhap", "soluongton", "giatriton")

# Tính toán ma trận tương quan giữa các biến
correlation_matrix <- cor(hanghoa[, variables_to_analyze])

# Chuyển đổi ma trận tương quan thành dataframe để sử dụng với ggplot2
correlation_df <- as.data.frame(as.table(correlation_matrix))
colnames(correlation_df) <- c("Variable1", "Variable2", "Correlation")

# Vẽ biểu đồ heatmap
ggplot(correlation_df, aes(Variable1, Variable2, fill = Correlation)) +
  geom_tile(color = "white") +
  scale_fill_gradient2(low = "blue", high = "red", mid =  "white",
                       midpoint = 0, limit = c(-1,1), space = "Lab",
                       name="Correlation") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, vjust = 1,
                                   size = 12, hjust = 1)) +
  coord_fixed()
```



Hình 2.9. Biểu đồ ma trận tương quan.

Kết luận:

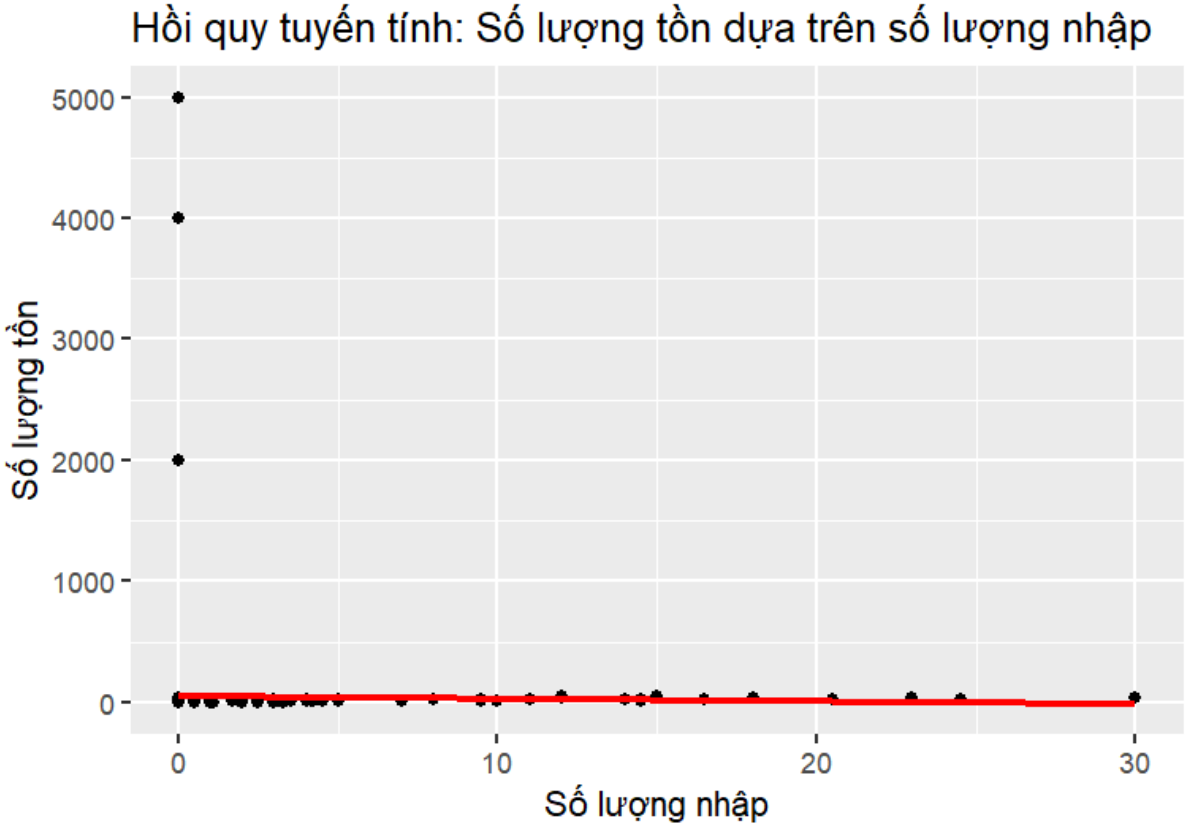
- Biểu đồ heatmap có thể giúp dễ dàng nhận biết mức độ tương quan giữa các biến thông qua sự biến đổi màu sắc trên lưới. Màu sắc càng sáng thể hiện mức độ tương quan càng cao, trong khi màu sắc càng tối thể hiện mức độ tương quan càng thấp.
- Các ô có màu đậm hơn cho thấy mối quan hệ tương quan lớn hơn giữa cặp biến tương ứng, trong khi các ô có màu nhạt hơn chỉ ra mối quan hệ tương quan yếu hơn.

2.3.2. Phân tích hồi quy

a. Hồi quy tuyến tính cho hàm nhập

```
#Phân tích hồi quy
# Tạo mô hình hồi quy tuyến tính
lm_model <- lm(soluongton ~ soluongnhap, data = hanghoa)
```

```
# Hiển thị kết quả của mô hình
summary(lm_model)
# Biểu đồ scatter plot cho biến nhập và tồn
ggplot(hanghoa, aes(x = soluongnhap, y = soluongton)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(x = "Số lượng nhập", y = "Số lượng tồn") +
  ggtitle("Hồi quy tuyến tính: Số lượng tồn dựa trên số lượng nhập")
```



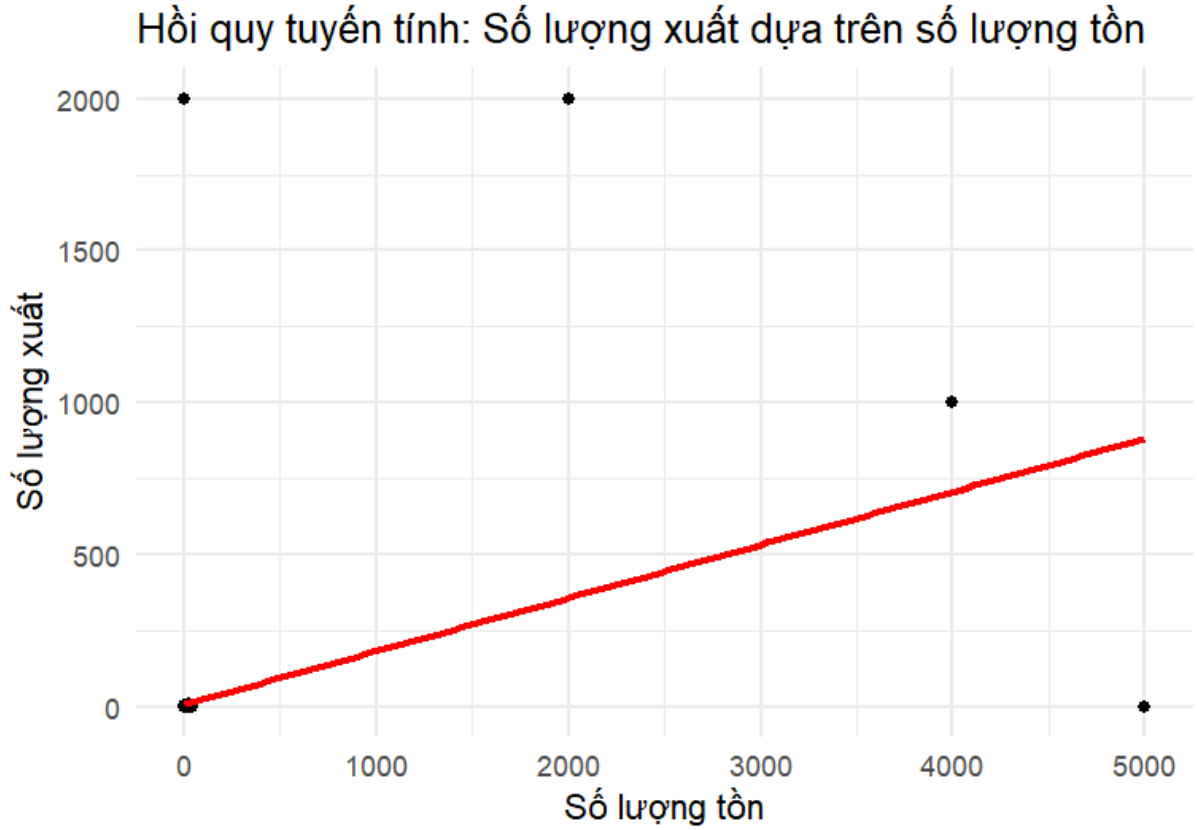
Hình 2.10. Biểu đồ hồi quy tuyến tính

Nhận xét:

- Mô hình có hiệu suất không tốt với R-squared nhỏ hơn hoặc bằng 0.5. Đường trendline có thể không phản ánh mối quan hệ mạnh mẽ giữa số lượng nhập và số lượng tồn. Biến Intercept có giá trị Estimate là 49.215 với t-value là 2.132 và p-value là 0.0338, trong khi biến soluongnhap có giá trị Estimate là -2.186 với t-value là -0.429 và p-value là 0.6680.
- Do giá trị p-value của Intercept (0.0338) nhỏ hơn mức ý nghĩa thống kê 0.05, chúng ta có bằng chứng thống kê để xác nhận mối quan hệ giữa Intercept và số lượng tồn. Vì vậy, trong trường hợp này, biến Intercept được coi là tốt hơn trong việc dự đoán số lượng tồn so với biến soluongnhap.

a. Tạo mô hình hồi quy tuyến tính cho hàm xuất:

```
# Tạo mô hình hồi quy tuyến tính cho hàm xuất
lm_model_xuat <- lm(soluongxuat ~ soluongton + giatriton, data = hanghoa)
# Hiển thị kết quả của mô hình
summary(lm_model_xuat)
# Biểu đồ scatter plot cho biến xuất và các biến đầu vào liên quan
ggplot(hanghoa, aes(x = soluongton, y = soluongxuat)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(x = "Số lượng tồn", y = "Số lượng xuất") +
  ggtitle("Hồi quy tuyến tính: Số lượng xuất dựa trên số lượng tồn") +
  theme_minimal()
```



Hình 2.11. Hồi quy tuyến tính số lượng xuất dựa trên số lượng tồn.

Nhận xét:

Mô hình đã được điều chỉnh với biến giá trị tồn (giatriton) và có vẻ có hiệu suất khá tốt với R-squared đáng chú ý. Tuy nhiên, cần kiểm tra các giả định của mô hình để đảm bảo tính đáng tin cậy của các ước lượng hồi quy. Biến Intercept: Giá trị Estimate là 22.44 với t-value là 1.707 và p-value là 0.0889. Biến soluongton: Giá trị Estimate là 0.1735 với t-value là 7.377 và p-value là 1.53e-12. Biến giatriton: Giá trị Estimate là -1.741e-07 với t-value là -1.363 và p-value là 0.1739. Với giá trị p-value, chúng ta xác định mức ý nghĩa thống kê của từng biến. Trong trường hợp này, biến soluongton có giá trị p-value rất nhỏ (1.53e-12), gần như bằng không, và giá trị t-value lớn

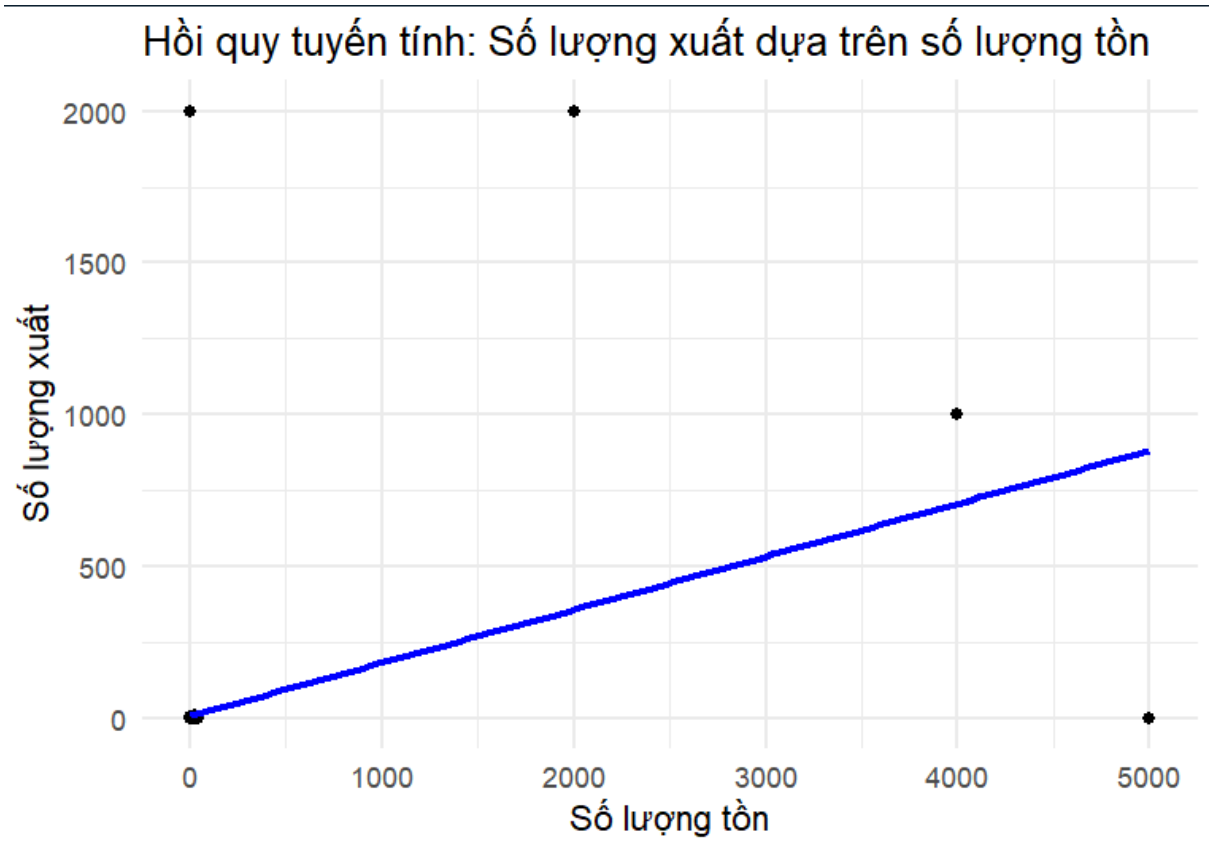
(7.377), cho thấy có mối quan hệ đáng kể với biến phụ thuộc. Do đó, biến soluongton được coi là có ảnh hưởng mạnh nhất trong mô hình hồi quy này.

b. Thêm đường dự đoán mô hình hồi quy lên biểu đồ scatter plot

```
# Thêm đường dự đoán từ mô hình hồi quy lên biểu đồ scatter plot
ggplot(hanghoa, aes(x = soluongton, y = soluongxuat)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  geom_smooth(method = "lm", se = FALSE, color = "blue", aes(y =
predict(lm_model_xuat))) +
  labs(x = "Số lượng tồn", y = "Số lượng xuất") +
  ggtitle("Hồi quy tuyến tính: Số lượng xuất dựa trên số lượng tồn") +
  theme_minimal()
```

Nhận xét:

Biểu đồ trên đã thêm đường dự đoán từ mô hình hồi quy mới (màu xanh). Đường này biểu diễn dự đoán số lượng xuất dựa trên số lượng tồn theo mô hình mới. Cần kiểm tra sự phù hợp của đường dự đoán này với dữ liệu thực tế.



Hình 2.12. Hồi quy tuyến tính thêm đường dự đoán.

c. Tính toán ma trận tương quan giữa các biến

```
# Tính toán ma trận tương quan giữa các biến
correlation_matrix <- cor(hanghoa[, variables_to_analyze])

# In ra ma trận tương quan
print(correlation_matrix)

# Lọc các cặp biến có hệ số tương quan gần 1 hoặc bằng -1
strong_correlations <- which(abs(correlation_matrix) >= 0.8 & correlation_matrix
!= 1, arr.ind = TRUE)

# Hiển thị các cặp biến có mối quan hệ tương quan mạnh
cat("Các cặp biến có mối quan hệ tương quan mạnh:\n")
for (i in 1:nrow(strong_correlations)) {
  row <- strong_correlations[i, 1]
  col <- strong_correlations[i, 2]
  cat(sprintf("%s và %s: %f\n", rownames(correlation_matrix)[row],
colnames(correlation_matrix)[col], correlation_matrix[row, col]))
}
```

Kết quả:

```
             soluongxuat  giatrixuat soluongnhap
soluongxuat  1.00000000  0.04408974 -0.03802536
giatrixuat   0.04408974  1.00000000 -0.34850380
soluongnhap -0.03802536 -0.34850380  1.00000000
giatrinhap  -0.04309972 -0.39501056  0.85253775
soluongton   0.39007661 -0.03504837 -0.02449542
giatriton   -0.08594912  0.12281087  0.29991449
             giatrinhap  soluongton   giatriton
soluongxuat -0.04309972  0.39007661 -0.08594912
giatrixuat  -0.39501056 -0.03504837  0.12281087
soluongnhap  0.85253775 -0.02449542  0.29991449
giatrinhap   1.00000000 -0.03216709  0.33234764
soluongton  -0.03216709  1.00000000 -0.03710198
giatriton    0.33234764 -0.03710198  1.00000000
```

Các cặp biến có mối quan hệ tương quan mạnh:

giatrinhap và soluongnhap: 0.852538
soluongnhap và giatrinhap: 0.852538

d. Phân tích độ đồng biến của mỗi cặp biến

```
# Phân tích độ đồng biến của mỗi cặp biến
cat("Phân tích độ đồng biến của mỗi cặp biến:\n")
for (i in 1:(length(variables_to_analyze) - 1)) {
  for (j in (i + 1):length(variables_to_analyze)) {
    variable1 <- variables_to_analyze[i]
    variable2 <- variables_to_analyze[j]
```

```
    correlation <- correlation_matrix[i, j]
    if (correlation > 0) {
      direction <- "dương (cùng hướng)"
    } else if (correlation < 0) {
      direction <- "âm (ngược hướng)"
    } else {
      direction <- "không có mối quan hệ tuyến tính"
    }
    cat(sprintf("%s và %s có độ đồng biến %s với hệ số tương quan là %f\n",
variable1, variable2, direction, correlation))
  }
}
```

Kết quả:

Phân tích độ đồng biến của mỗi cặp biến:
soluongxuat và giatrixuat có độ đồng biến dương (cùng hướng) với hệ số tương quan là 0.044090
soluongxuat và soluongnhap có độ đồng biến âm (ngược hướng) với hệ số tương quan là -0.038025
soluongxuat và giatrinhap có độ đồng biến âm (ngược hướng) với hệ số tương quan là -0.043100
soluongxuat và soluongton có độ đồng biến dương (cùng hướng) với hệ số tương quan là 0.390077
soluongxuat và giatriton có độ đồng biến âm (ngược hướng) với hệ số tương quan là -0.085949
giatrixuat và soluongnhap có độ đồng biến âm (ngược hướng) với hệ số tương quan là -0.348504
giatrixuat và giatrinhap có độ đồng biến âm (ngược hướng) với hệ số tương quan là -0.395011
giatrixuat và soluongton có độ đồng biến âm (ngược hướng) với hệ số tương quan là -0.035048
giatrixuat và giatriton có độ đồng biến dương (cùng hướng) với hệ số tương quan là 0.122811
soluongnhap và giatrinhap có độ đồng biến dương (cùng hướng) với hệ số tương quan là 0.852538
soluongnhap và soluongton có độ đồng biến âm (ngược hướng) với hệ số tương quan là -0.024495
soluongnhap và giatriton có độ đồng biến dương (cùng hướng) với hệ số tương quan là 0.299914
giatrinhap và soluongton có độ đồng biến âm (ngược hướng) với hệ số tương quan là -0.032167
giatrinhap và giatriton có độ đồng biến dương (cùng hướng) với hệ số tương quan là 0.332348
soluongton và giatriton có độ đồng biến âm (ngược hướng) với hệ số tương quan là -0.037102

CHƯƠNG 3

TRỰC QUAN DỮ LIỆU

3.1. Trực quan hóa dữ liệu

3.1.1. Tổng hợp dữ liệu số lượng hàng bán theo mã hàng và tháng

```
library(dplyr)
library(lubridate)

# Giả sử dữ liệu của bạn đã được lưu vào một dataframe có tên là data_df

# Chuyển đổi cột ngày thành định dạng ngày
data_df$ngayhachtoan <- dmy(data_df$ngayhachtoan)

# Tách dữ liệu theo tháng và hiển thị các mã hàng bán trong mỗi tháng
data_df %>%
  mutate(thang = month(ngayhachtoan),
         nam = year(ngayhachtoan)) %>%
  group_by(thang, nam, mahang) %>%
  summarize(so_luong_ban = sum(soluongxuat)) %>%
  filter(so_luong_ban > 0) %>%
  arrange(nam, thang)
```

Cài đặt và tải các gói thư viện cần thiết: Trước tiên, gói dplyr và lubridate được cài đặt và tải lên. Các gói này chứa các hàm và công cụ để thao tác dữ liệu và xử lý ngày tháng trong R.

Chuyển đổi cột ngày thành định dạng ngày: Hàm dmy() từ gói lubridate được sử dụng để chuyển đổi cột "ngayhachtoan" trong dataframe data_df thành định dạng ngày-tháng-năm.

Tách dữ liệu theo tháng và hiển thị các mã hàng bán trong mỗi tháng: Trong phần này, dữ liệu được biến đổi để tạo ra tổng số lượng hàng bán cho mỗi mã hàng trong mỗi tháng.

Cụ thể:

- mutate(thang = month(ngayhachtoan), nam = year(ngayhachtoan)): Tạo hai cột mới là "thang" (tháng) và "nam" (năm) từ cột ngày hạch toán.
- group_by(thang, nam, mahang): Nhóm dữ liệu theo tháng, năm và mã hàng.
- summarize(so_luong_ban = sum(soluongxuat)): Tính tổng số lượng xuất (số lượng bán) cho mỗi nhóm.
- filter(so_luong_ban > 0): Loại bỏ các hàng có số lượng bán bằng 0.
- arrange(nam, thang): Sắp xếp kết quả theo năm và tháng.

3.1.2. Trích xuất dữ liệu các quý và vẽ biểu đồ mã hàng theo đơn giá từng quý.

```
#dữ liệu mã hàng và số lượng quý 1
dongiaa1 <- quy1 %>%
```

```
  group_by(mahang) %>%
  summarise(dongia)
dongiaa1
ggplot(dongiaa1, aes(x = dongia,y = mahang)) +
  geom_col(stat="identiy",fill='lightblue', position = "dodge") +
  labs(title = "y",
       x = "don gia dữ liệu quý 1",
       y = "ma hang dữ liệu quý 1")
```

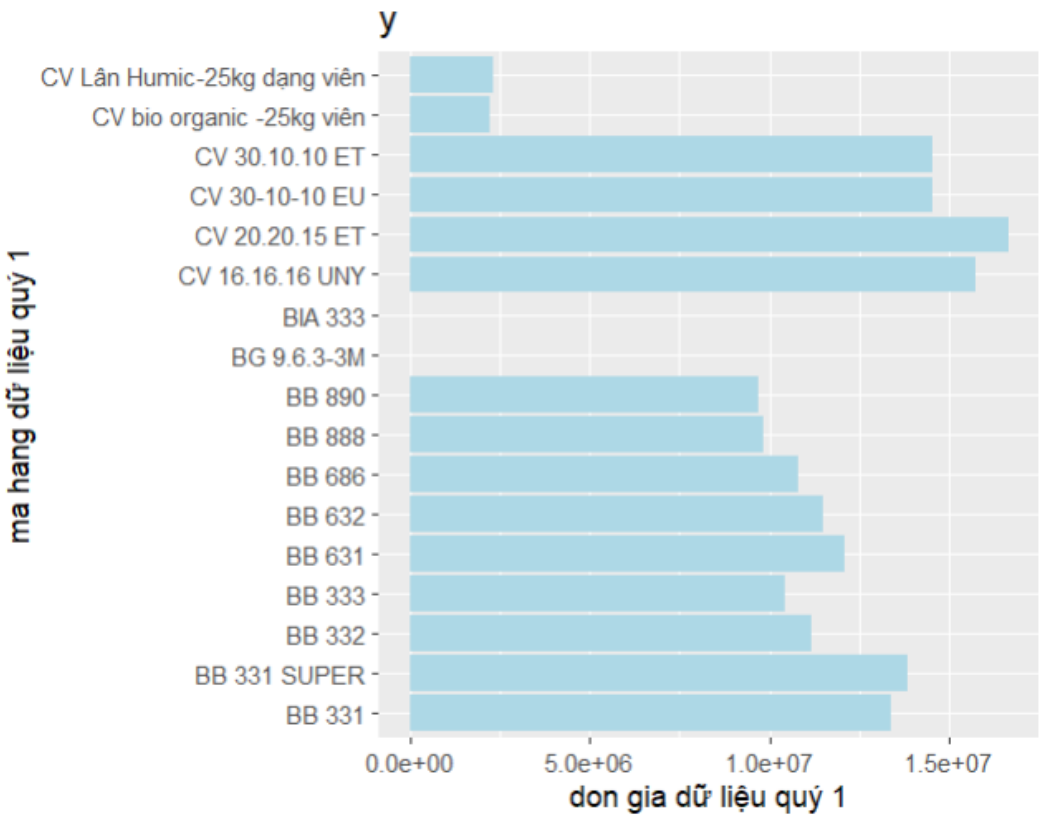
Tính toán giá trị trung bình của từng mặt hàng trong quý 1:

- dongiaa1 <- quy1 %>% group_by(mahang) %>% summarise(dongia): Đoạn code này tính giá trị trung bình (hoặc có thể là một phép toán khác, tùy thuộc vào ý định ban đầu của bạn) của cột "dongia" cho mỗi mã hàng trong dữ liệu của quý 1 (quy1). Dữ liệu được nhóm theo "mahang" (mã hàng) và sau đó được tổng hợp (summarise) để tính giá trị trung bình (hoặc phép toán khác nếu cần).

Tạo biểu đồ cột để biểu diễn giá trị trung bình của từng mặt hàng trong quý 1:

- ggplot(dongiaa1, aes(x = dongia,y = mahang)) + geom_col(stat="identity",fill='lightblue', position = "dodge"): Dòng này sử dụng gói ggplot2 để tạo biểu đồ cột. Biểu đồ này sử dụng dữ liệu trong dataframe "dongiaa1". Mỗi cột trong biểu đồ sẽ biểu diễn giá trị trung bình của một mặt hàng, với trục x là giá trị trung bình (dongia) và trục y là mã hàng (mahang).
- labs(title = "y", x = "don gia dữ liệu quý 1", y = "ma hang dữ liệu quý 1"): Dòng này đặt tiêu đề cho biểu đồ và các nhãn cho trục x và trục y.

Tóm lại: Mã này tính giá trị trung bình hoặc một phép toán khác (tùy thuộc vào cụ thể của cột "dongia") của từng mặt hàng trong quý 1 và sau đó tạo một biểu đồ cột để biểu diễn các giá trị này. Điều này giúp bạn hiểu được phân bố giá trị của các mặt hàng trong quý 1.



Hình 3.1. Biểu đồ mô tả mã hàng đơn giá quý 1.

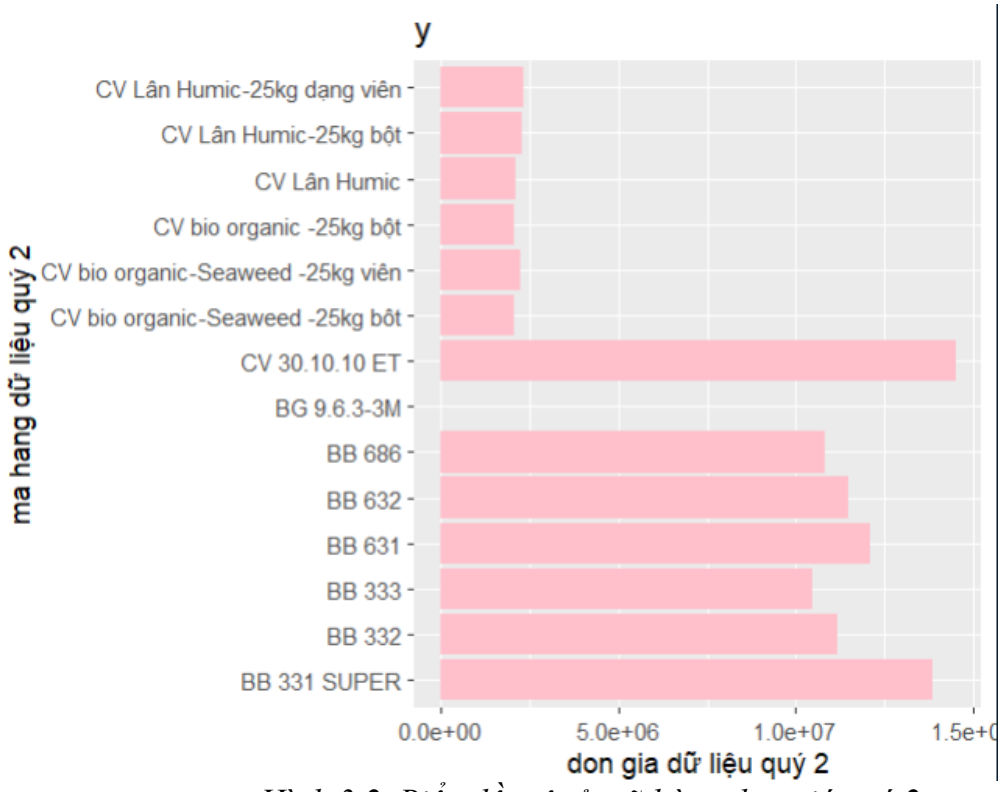
Nhận xét:

- Qua biểu đồ ta có thể thấy, mã hàng hóa CV30.10.10 ET, CV 30-10-10EU, CV 16.16.16 UNY có đơn giá trong quý 1 là cao nhất, sau đó đến mã BB331 SUPER, BB331 và còn mã BB còn lại. Mã CV Lân Humic-25kg dạng viên và CV Bio Organic-25kg viên là 2 mã có đơn giá thấp nhất.

Làm tương tự, ta sẽ có biểu đồ của từng quý 2,3,4.

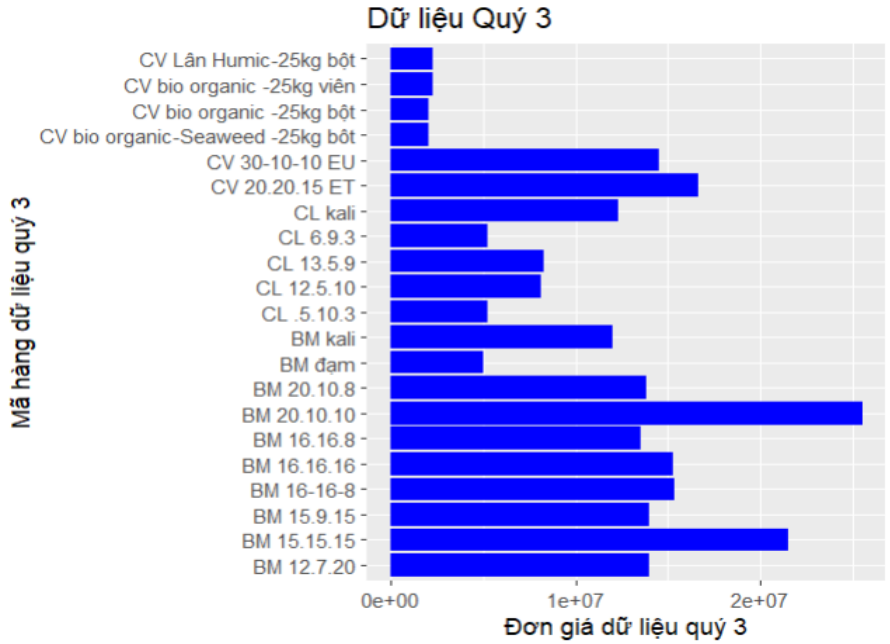
```
#Dữ liệu mà hàng và số lượng của quý 2
dongiaa2 <- quy2 %>%
  group_by(mahang) %>%
  summarise(dongia)
dongiaa2
ggplot(dongiaa2, aes(x = dongia,y = mahang)) +
  geom_col(stat="identiy",fill='pink', position = "dodge") +
  labs(title = "y",
       x = "don gia dữ liệu quý 2",
       y = "ma hang dữ liệu quý 2")


#Dữ liệu mã hàng và số lượng quý 3
dongiaa3 <- quy3 %>%
  group_by(mahang) %>%
  summarise(dongia)
ggplot(dongiaa3, aes(x = dongia, y = mahang)) +
```

```
  geom_col(stat = "identity", fill = 'blue', position = "dodge") +
  labs(title = "Dữ liệu Quý 3",
       x = "Đơn giá dữ liệu quý 3",
       y = "Mã hàng dữ liệu quý 3")


#Dữ liệu mã hàng và số lượng của quý 4
dongiaa4 <- quy4 %>%
  group_by(mahang) %>%
  summarise(dongia)
dongiaa4
ggplot(dongiaa4, aes(x = dongia, y = mahang)) +
  geom_col(stat = "identity", fill = 'red', position = "dodge") +
  labs(title = "Dữ liệu Quý 4",
       x = "Đơn giá dữ liệu quý 4",
       y = "Mã hàng dữ liệu quý 4")
```



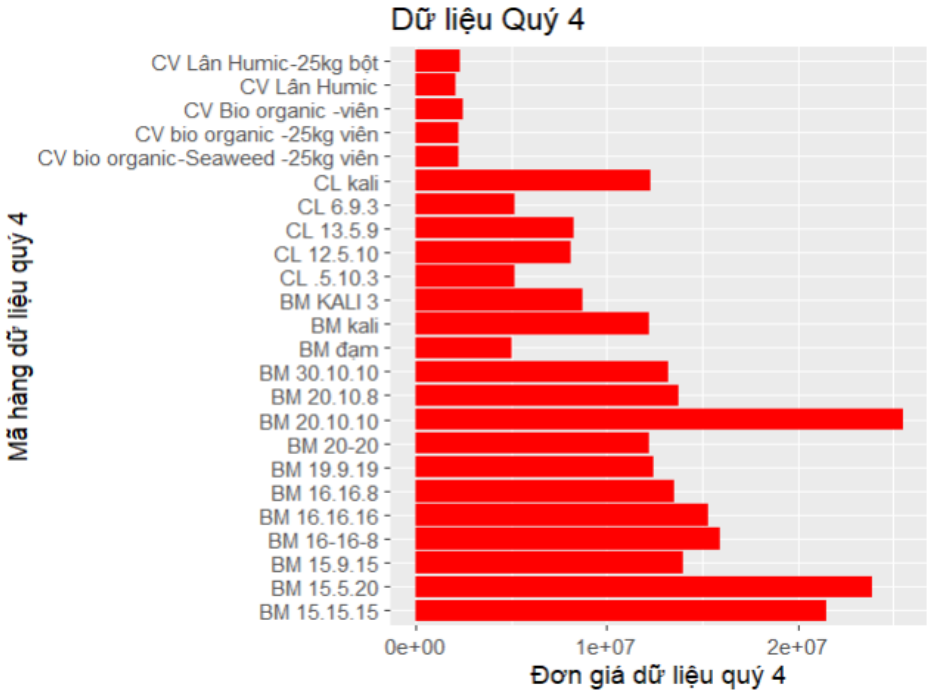
Hình 3.2. Biểu đồ mô tả mã hàng đơn giá quý 2.

Ở quý 2, các mã hàng hóa BB vẫn có đơn giá trung bình cao hơn các mã hàng khác.



Hình 3.3. Biểu đồ mô tả mã hàng đơn giá quý 3.

Ở quý 3, các mã hàng hóa BB vẫn có đơn giá trung bình cao hơn các mã hàng khác, sau đó đến cái mã hàng hóa CV, CL với mức đơn giá biến thiên không ổn định.



Hình 3.4. Biểu đồ mô tả mã hàng đơn giá quý 4.

Ở quý 4, các mã hàng hóa BM vẫn có đơn giá cao hơn các mã khác trong quý 4.

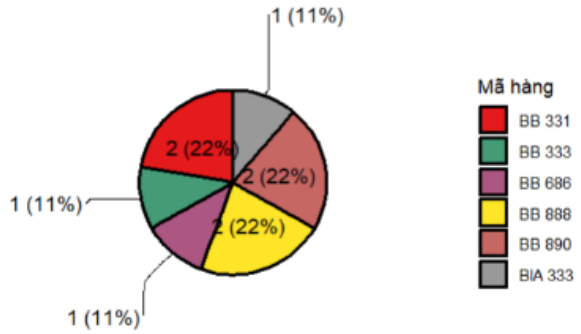
3.1.3. Biểu đồ Pie biểu đồ biểu diễn phân phối tỷ lệ của các mã hàng trong tháng tương ứng.

```
*biểu đồ pie 2D
hanghoa$Month = format(hanghoa$ngayhachtoan,'%m')

#dữ liệu chia theo tháng theo từng mã hàng
# Tạo danh sách chứa biểu đồ pie2D cho từng tháng
pie_list <- lapply(1:12, function(month) {
  thang_data <- extract_month_data(hanghoa, sprintf("%02d", month))
  ggpie(data = thang_data, group_key = "mahang", count_type = "full",
        label_info="all",label_type="horizon",label_split = NULL,
        label_size=4,label_pos="in",label_threshold = 15) +
    labs(title = paste("Biểu đồ 2D tháng", month), fill = "Mã hàng")
})

# Hiển thị danh sách biểu đồ pie
pie_list
```

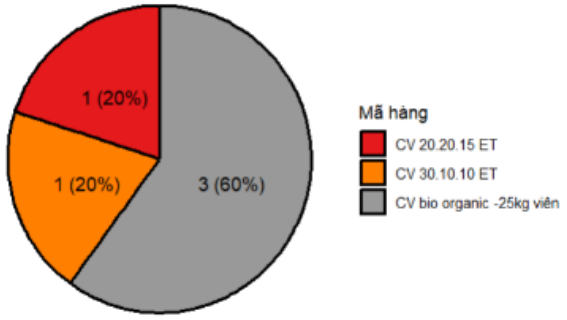
- hanghoa$Month = format(hanghoa$ngayhachtoan,'%m'): Dòng này tạo một cột mới trong dataframe hanghoa để lưu trữ thông tin về tháng dựa trên cột "ngayhachtoan".
- lapply(1:12, function(month) { ... }): Hàm lapply được sử dụng để lặp qua các tháng từ 1 đến 12 và tạo một danh sách các biểu đồ pie2D cho từng tháng.
- extract_month_data(hanghoa, sprintf("%02d", month)): Hàm này được giả định là một hàm tùy chỉnh để trích xuất dữ liệu cho một tháng cụ thể từ dataframe hanghoa.
- ggpie(...): Dùng để tạo biểu đồ pie2D từ dữ liệu của mỗi tháng, với các tham số được thiết lập để hiển thị mã hàng dưới dạng nhãn và phân phối tỷ lệ của mỗi mã hàng trong tháng.

Hiển thị danh sách biểu đồ pie: Sau khi tạo danh sách các biểu đồ pie cho từng tháng, danh sách này được hiển thị ra màn hình.
Xong khi chạy xong đoạn Code, ta sẽ được 12 biểu đồ tỉ lệ phân phối mã hàng tương ứng với 12 tháng tương ứng.

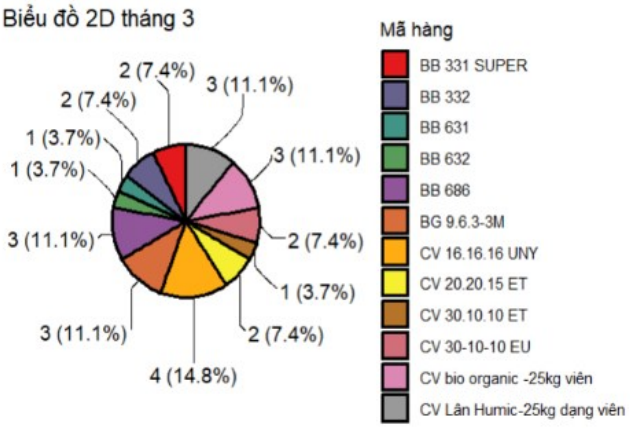
Biểu đồ 2D tháng 1



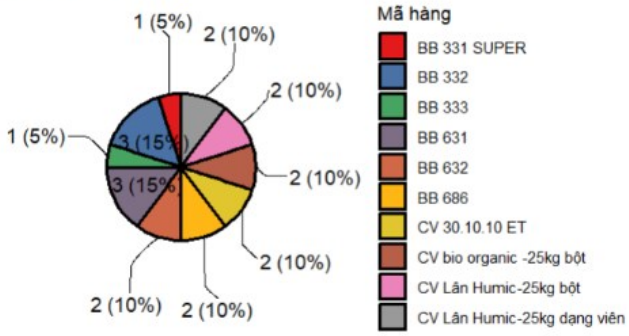
Biểu đồ 2D tháng 2



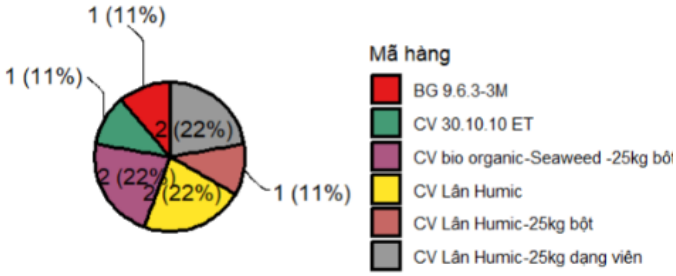
Biểu đồ 2D tháng 3



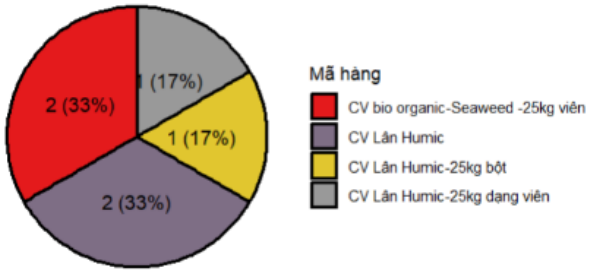
Biểu đồ 2D tháng 4

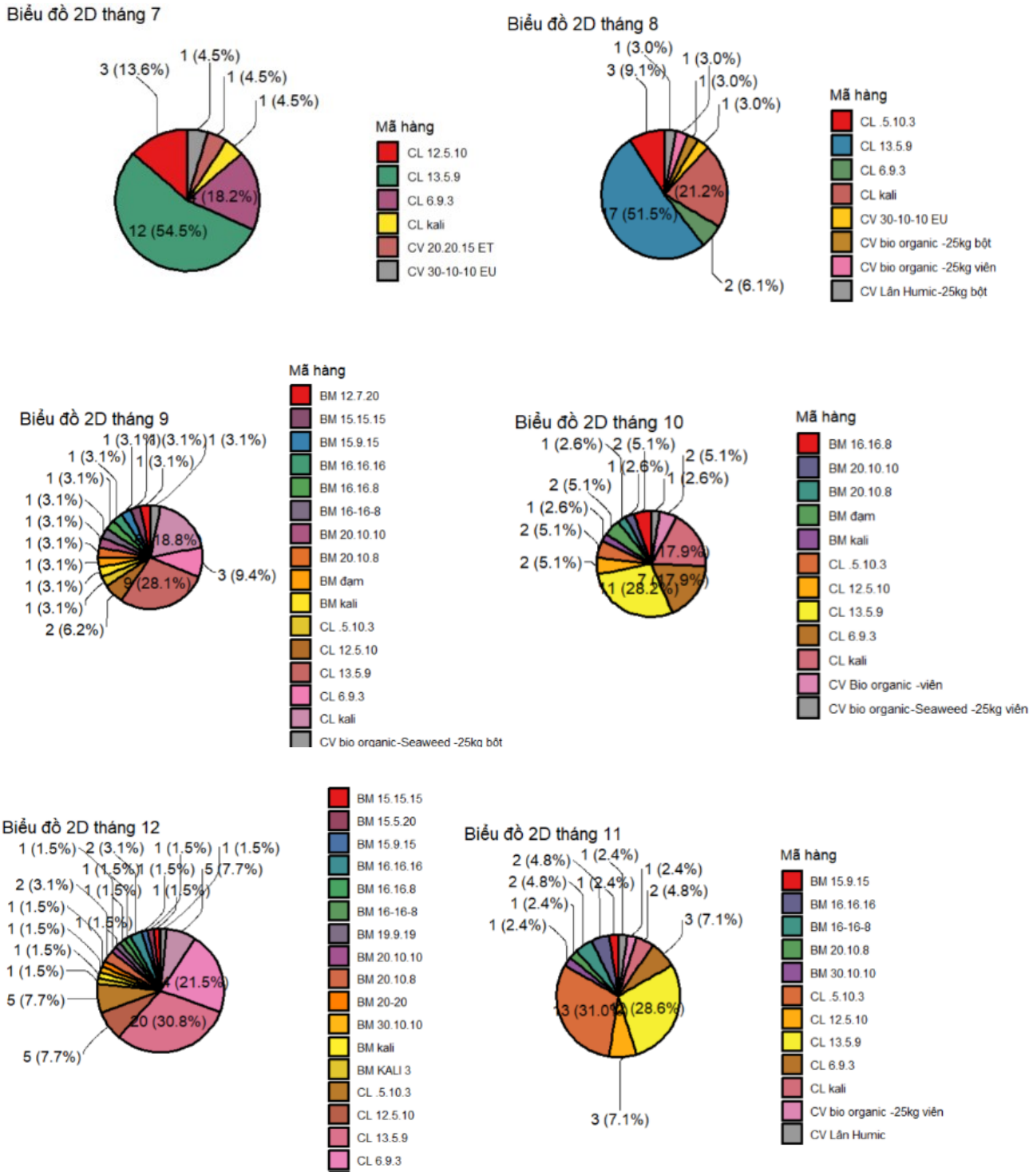


Biểu đồ 2D tháng 5



Biểu đồ 2D tháng 6

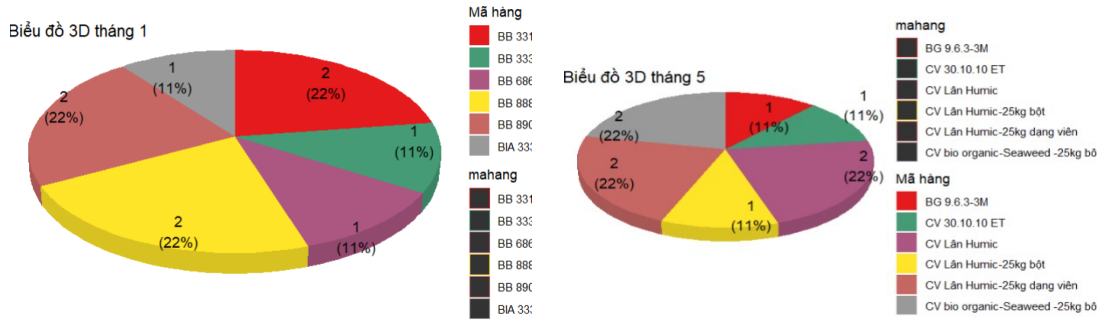




Hình 3.5. Biểu đồ tỷ lệ phân phối mã hàng trong 12 tháng.

Trên đây là 12 biểu đồ về tỉ lệ phân phối các mã hàng trong từng tháng. Nhìn qua ta có thể thấy mã nào được phân phối bán nhiều nhất.
Ngoài ra, ta có thể vẽ thêm biểu đồ pie 3D để nhìn biểu đồ đẹp hơn và có thể nhìn dễ dàng hơn. Dưới đây là một số tháng sử dụng biểu đồ pie 3D, các tháng còn lại tương tự.

```
#Dữ liệu 3D pie
pie3D_list <- lapply(1:12, function(month) {
  thang_data <- extract_month_data(hanghoa, sprintf("%02d", month))
  ggpie3D(data = thang_data, group_key = "mahang", count_type = "full",
tilt_degrees = -2) +
 labs(title = paste("Biểu đồ 3D tháng", month), fill = "Mã hàng")
})

# Hiển thị danh sách biểu đồ pie
pie3D_list
```



Hình 3.6. Biểu đồ pie 3D tỷ lệ phân phối mã hàng trong tháng 1 và tháng 5.

3.1.4. Trực quan hóa bằng biểu đồ line chart trực quan hóa dữ liệu về số lượng nhập của các nhóm mã hàng trong từng quý

- Chứa thông tin về tháng từ cột "ngayhachtoan":

```
# Tạo cột mới chứa thông tin về tháng từ cột "ngayhachtoan"
quy1$Thang <- format(as.Date(quy1$ngayhachtoan, format="%m/%d/%Y"), "%m")
quy2$Thang <- format(as.Date(quy2$ngayhachtoan, format="%m/%d/%Y"), "%m")
quy3$Thang <- format(as.Date(quy3$ngayhachtoan, format="%m/%d/%Y"), "%m")
quy4$Thang <- format(as.Date(quy4$ngayhachtoan, format="%m/%d/%Y"), "%m")
```

Sử dụng hàm format() và as.Date() để chuyển đổi cột "ngayhachtoan" thành định dạng ngày tháng, sau đó sử dụng hàm format() một lần nữa để chỉ lấy thông tin về tháng ("%m") từ ngày tháng đã chuyển đổi. Kết quả được gán vào cột "Thang" trong mỗi dataframe quy1, quy2, quy3, và quy4.

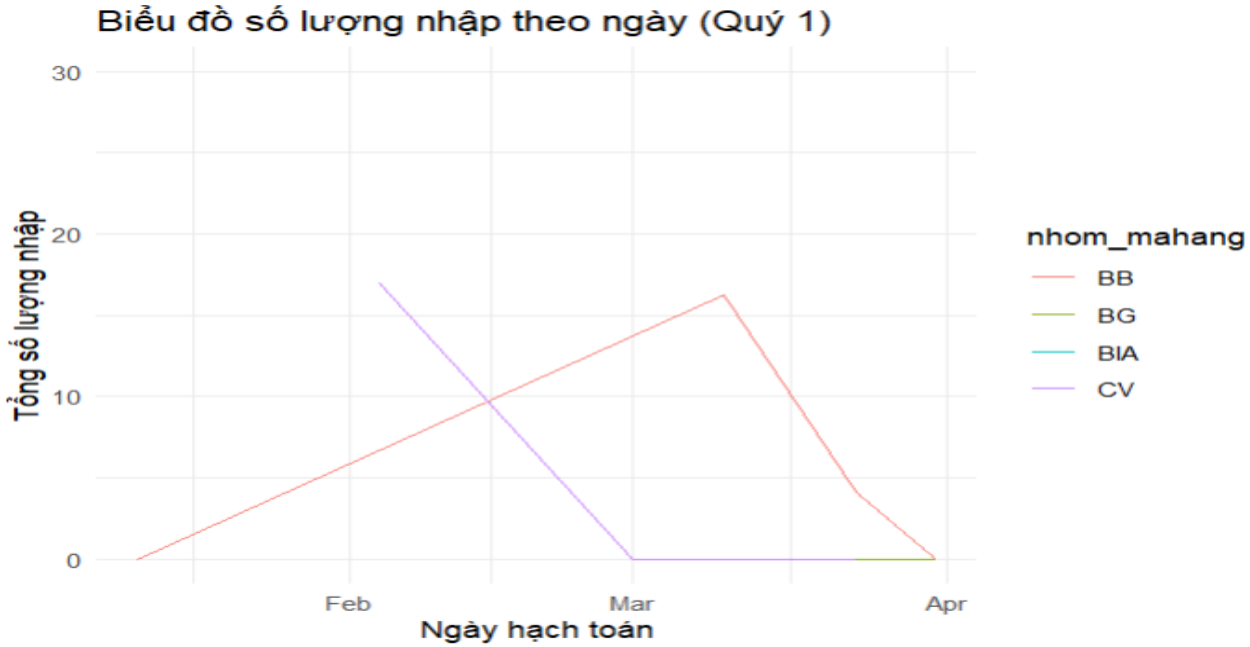
- Gom các mã hàng vào các nhóm tương ứng của quý 1:

```
# Gom các mã hàng vào các nhóm tương ứng quý 1
nhom_mahang <- c("BB", "BG", "BIA", "CV")
quy1 <- quy1 %>%
  mutate(nhom_mahang = case_when(
    grepl("BB", mahang) ~ "BB",
    grepl("BG", mahang) ~ "BG",
    grepl("BIA", mahang) ~ "BIA",
    grepl("CV", mahang) ~ "CV",
    TRUE ~ mahang  # Giữ nguyên nếu không thuộc bất kỳ nhóm nào
  ))
```

Sử dụng hàm mutate() để tạo một cột mới "nhom_mahang". Cột này sẽ chứa các nhóm mã hàng dựa trên điều kiện đã chỉ định, trong đó mỗi nhóm sẽ được gán một nhóm mã hàng tương ứng. Nếu mã hàng không thuộc bất kỳ nhóm nào, mã hàng đó sẽ được giữ nguyên.
Điều này được thực hiện thông qua hàm case_when() để kiểm tra các điều kiện và gán giá trị cho cột "nhom_mahang" dựa trên các điều kiện này.

- Vẽ biểu đồ Linechart số lượng nhập của các mã hàng trong quý 1:

```
# Vẽ biểu đồ line chart
ggplot(summary_data1, aes(x = ngayhachtoan, y = total_soluongnhap, color =
nhom_mahang, group = nhom_mahang)) +
  geom_line() +
  labs(title = "Biểu đồ số lượng nhập theo ngày (Quý 1)",
       x = "Ngày hạch toán", y = "Tổng số lượng nhập") +
  theme_minimal()
```



Hình 3.7. Biểu đồ số lượng nhập theo ngày – quý 1.

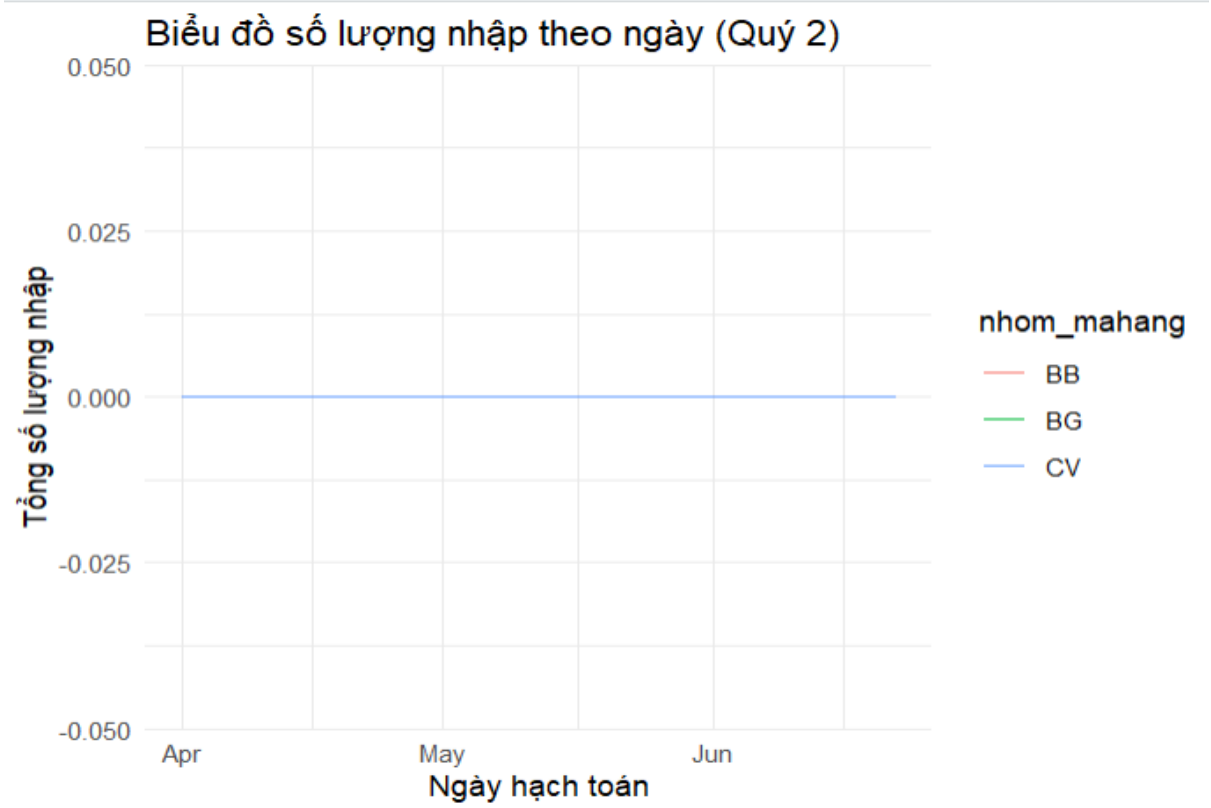
Từ biểu đồ này ra có thể thấy rằng:

- Vào tháng 1,tháng 2 mã BB được nhập với số lượng rất nhiều tăng dần theo thời gian ➜ xu hướng bán hàng tốt, nhưng từ giữa tháng 2 đến hết tháng 3 thì lại nhập với số lượng ít đi.
- Các mã CV được nhập với số lượng cũng khá cao ở giữa tháng 1, và giảm dần theo thời gian và đến đầu tháng 2 là không bán nữa. Các mã BG ở giai đoạn này không bán được,
- mã BIA không bán được có bởi vì đây là số lượng công ty mua tặng cho nhân viên, không tính giá, số lượng.

Gom các mã hàng vào các nhóm tương ứng của quý 2:

```
# Gom các mã hàng vào các nhóm tương ứng quý 2
nhom_mahang <- c("BB", "BG", "CV")
quy2 <- quy2 %>%
  mutate(nhom_mahang = case_when(
    grepl("BB", mahang) ~ "BB",
    grepl("BG", mahang) ~ "BG",
    grepl("CV", mahang) ~ "CV",
    TRUE ~ mahang  # Giữ nguyên nếu không thuộc bất kỳ nhóm nào
  ))
```

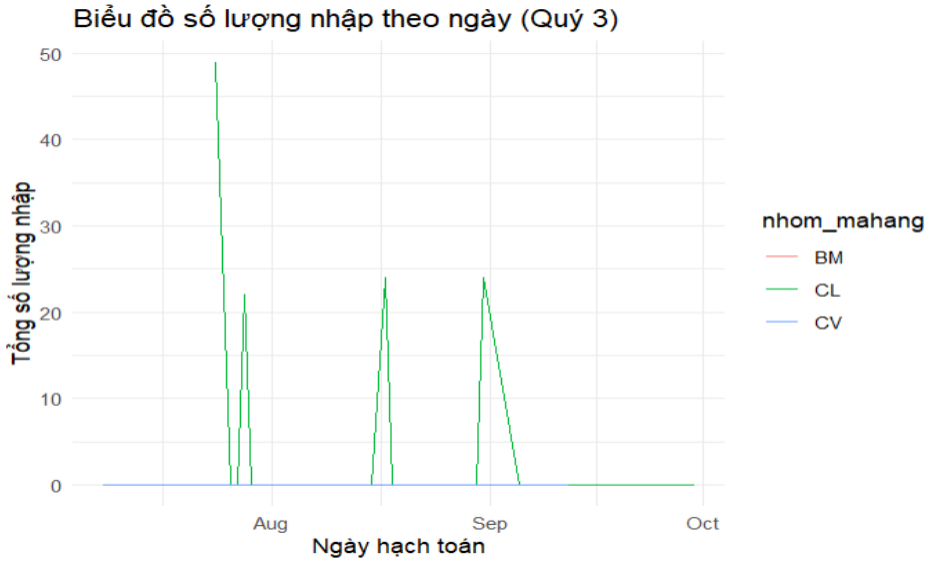
- Vẽ biểu đồ Linechart số lượng nhập của các mã hàng trong quý 2:



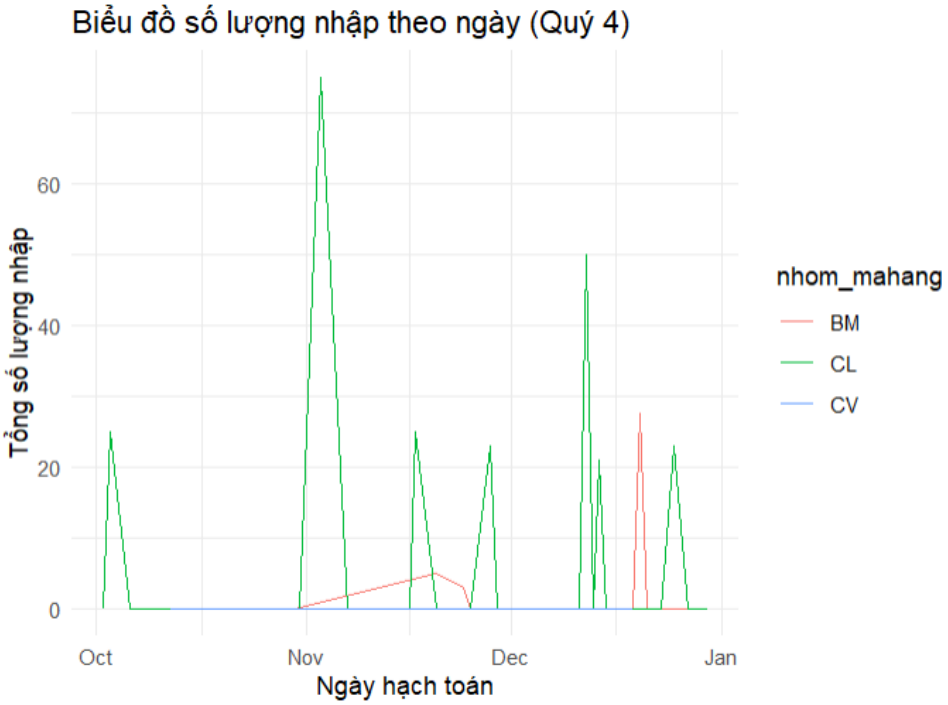
Hình 3.8. Biểu đồ số lượng nhập theo ngày – quý 2.

- Làm tương tự, ta sẽ có dữ liệu biểu đồ quý 3, quý 4:

```
# Gom các mã hàng vào các nhóm tương ứng của quý 3
nhom_mahang <- c("BM", "CL", "CV")
quy3 <- quy3 %>%
  mutate(nhom_mahang = case_when(
    grepl("BM", mahang) ~ "BM",
    grepl("CL", mahang) ~ "CL",
    grepl("CV", mahang) ~ "CV",
    TRUE ~ mahang  # Giữ nguyên nếu không thuộc bất kỳ nhóm nào
  ))
# Vẽ biểu đồ line chart
ggplot(summary_data3, aes(x = ngayhachtoan, y = total_soluongnhap, color =
nhom_mahang, group = nhom_mahang)) +
  geom_line() +
  labs(title = "Biểu đồ số lượng nhập theo ngày (Quý 3)",
       x = "Ngày hạch toán", y = "Tổng số lượng nhập") +
  theme_minimal()
# Gom các mã hàng vào các nhóm tương ứng của quý 4
nhom_mahang <- c("BM", "CL", "CV")
quy4 <- quy4 %>%
  mutate(nhom_mahang = case_when(
    grepl("BM", mahang) ~ "BM",
    grepl("CL", mahang) ~ "CL",
    grepl("CV", mahang) ~ "CV",
    TRUE ~ mahang  # Giữ nguyên nếu không thuộc bất kỳ nhóm nào
  ))
# Vẽ biểu đồ line chart
ggplot(summary_data4, aes(x = ngayhachtoan, y = total_soluongnhap, color =
nhom_mahang, group = nhom_mahang)) +
  geom_line() +
  labs(title = "Biểu đồ số lượng nhập theo ngày (Quý 4)",
       x = "Ngày hạch toán", y = "Tổng số lượng nhập") +
  theme_minimal()
```



Hình 3.9. Biểu đồ số lượng nhập theo ngày – quý 3.



Hình 3.10. Biểu đồ số lượng nhập theo ngày – quý 4.

Từ các biểu đồ quý 2,3,4 ta có thể nhận thấy rằng:

- Ở quý 2, số lượng hàng hóa không nhập nhiều
- Ở quý 3, mã hàng hóa CL được nhập nhiều và đều
- Ở quý 4, mã hàng hóa CL vẫn tiếp tục được nhập với số lượng đều, và nhập thêm các mã hàng hóa BM từ tháng 11 đến tháng 12.

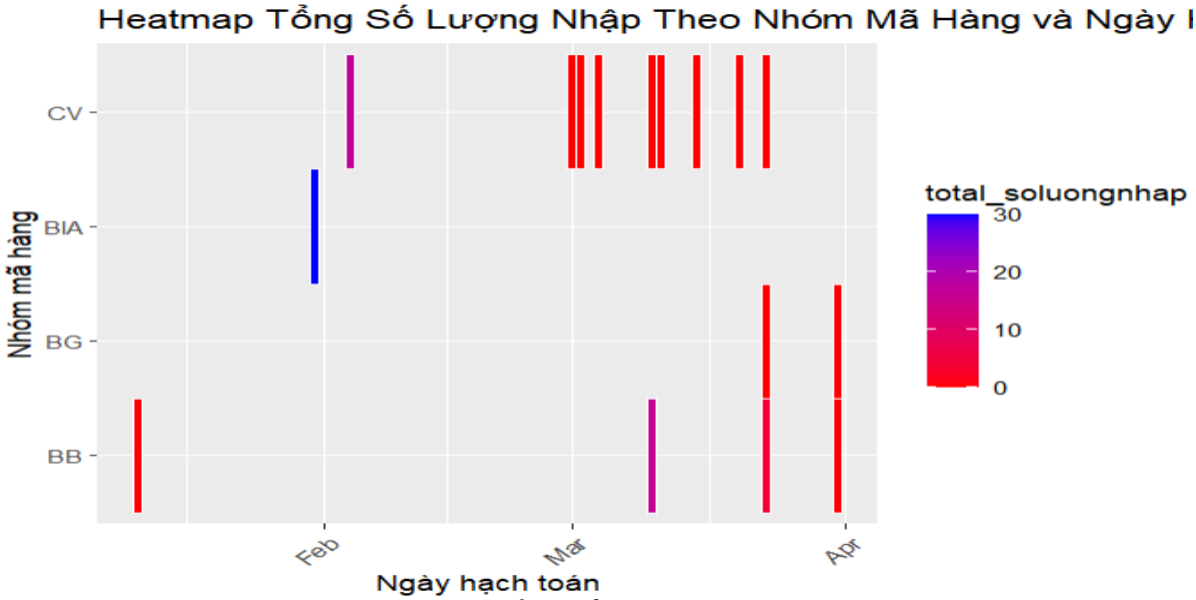
3.1.5. Vẽ biểu đồ headmap cho dữ liệu quý 1,2,3,4 đã trích xuất

Headmap giúp trực quan hiển thị tổng số lượng nhập của mỗi nhóm mã hàng theo từng ngày trong quý 1.

```
# Vẽ heatmap từ dữ liệu đã tóm tắt
#quý 1
ggplot(summary_data1, aes(x = ngayhachtoan, y = nhom_mahang, fill =
total_soluongnhap)) +
  geom_tile(color = "white") +
  scale_fill_gradient(low = "red", high = "blue") +
  labs(x = "Ngày hạch toán", y = "Nhóm mã hàng", title = "Heatmap Tổng Số Lượng
Nhập Theo Nhóm Mã Hàng và Ngày Hạch Toán", size = 20) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

- ggplot(summary_data1, aes(x = ngayhachtoan, y = nhom_mahang, fill = total_soluongnhap)): Đây là phần khai báo dữ liệu cho biểu đồ heatmap. Trục x sẽ là ngày hạch toán, trục y sẽ là nhóm mã hàng, và fill (màu nền) sẽ biểu diễn tổng số lượng nhập.

- geom_tile(color = "white"): Sử dụng geom_tile để tạo các ô vuông (tiles) trên heatmap, mỗi ô tương ứng với một cặp giá trị ngày hạch toán và nhóm mã hàng. Màu viền của mỗi ô được đặt là trắng.
- scale_fill_gradient(low = "red", high = "blue"): Điều chỉnh màu sắc cho heatmap. Trong trường hợp này, từ màu đỏ (thấp) đến màu xanh (cao), biểu thị mức độ của tổng số lượng nhập.
- labs(...): Thiết lập các nhãn cho trục và tiêu đề của biểu đồ.
- theme(axis.text.x = element_text(angle = 45, hjust = 1)): Điều chỉnh góc hiển thị của nhãn trên trục x để tránh việc chồng chéo khi có quá nhiều giá trị.



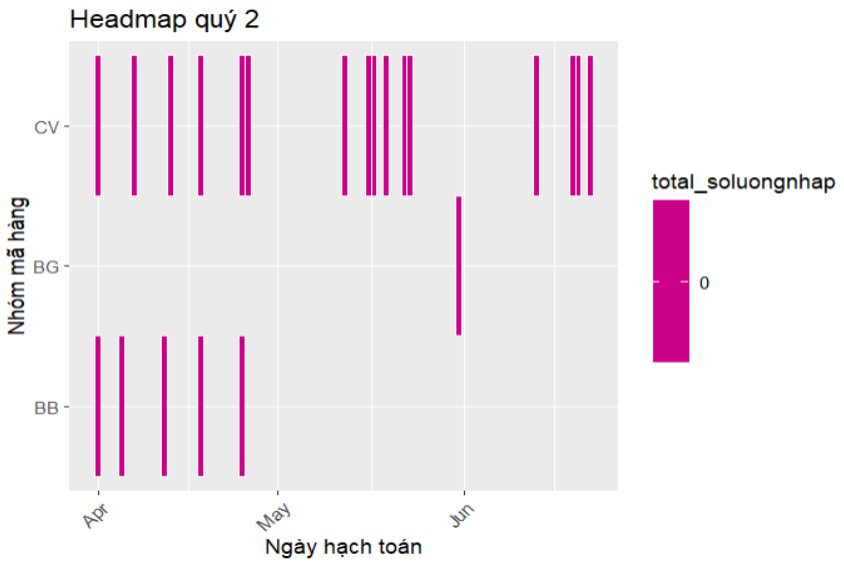
Hình 3.11. Heatmap tổng số lượng nhập theo nhóm mã hàng – quý 1.

Nhận xét: Nhóm mã hàng CV có số lượng nhập cao vào 1 vài ngày cụ thể trong tháng 3. Nhóm mã hàng BIA có số lượng nhập cao duy nhất vào tháng 2. Nhóm mã hàng BB và BG có số lượng nhập thấp và không ổn định suốt các tháng.

- Tương tự với dữ liệu quý 2,3,4 ta sẽ có 3 biểu đồ như sau:

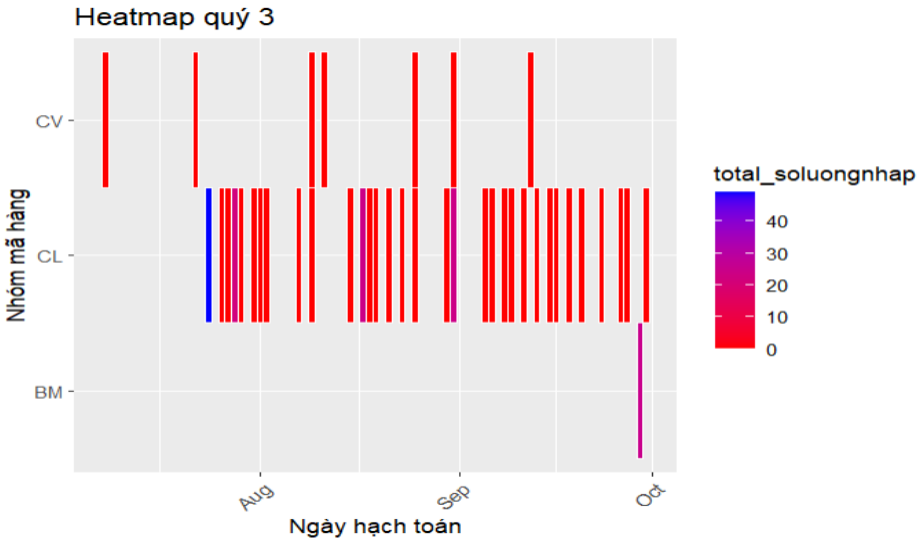
```
#quý 2
ggplot(summary_data2, aes(x = ngayhachtoan, y = nhom_mahang, fill =
total_soluongnhap)) +
  geom_tile(color = "white") +
  scale_fill_gradient(low = "red", high = "blue") +
  labs(x = "Ngày hạch toán", y = "Nhóm mã hàng", title = "Heatmap Tổng Số Lượng
Nhập Theo Nhóm Mã Hàng và Ngày Hạch Toán", size = 20) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
#quý 3
ggplot(summary_data3, aes(x = ngayhachtoan, y = nhom_mahang, fill =
total_soluongnhap)) +
  geom_tile(color = "white") +
  scale_fill_gradient(low = "red", high = "blue") +
  labs(x = "Ngày hạch toán", y = "Nhóm mã hàng", title = "Heatmap Tổng Số Lượng
Nhập Theo Nhóm Mã Hàng và Ngày Hạch Toán", size = 20) +
```

```
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
#quý 4
ggplot(summary_data4, aes(x = ngayhachtoan, y = nhom_mahang, fill =
total_soluongnhap)) +
  geom_tile(color = "white") +
  scale_fill_gradient(low = "red", high = "blue") +
  labs(x = "Ngày hạch toán", y = "Nhóm mã hàng", title = "Heatmap Tổng Số Lượng
Nhập Theo Nhóm Mã Hàng và Ngày Hạch Toán", size = 20) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



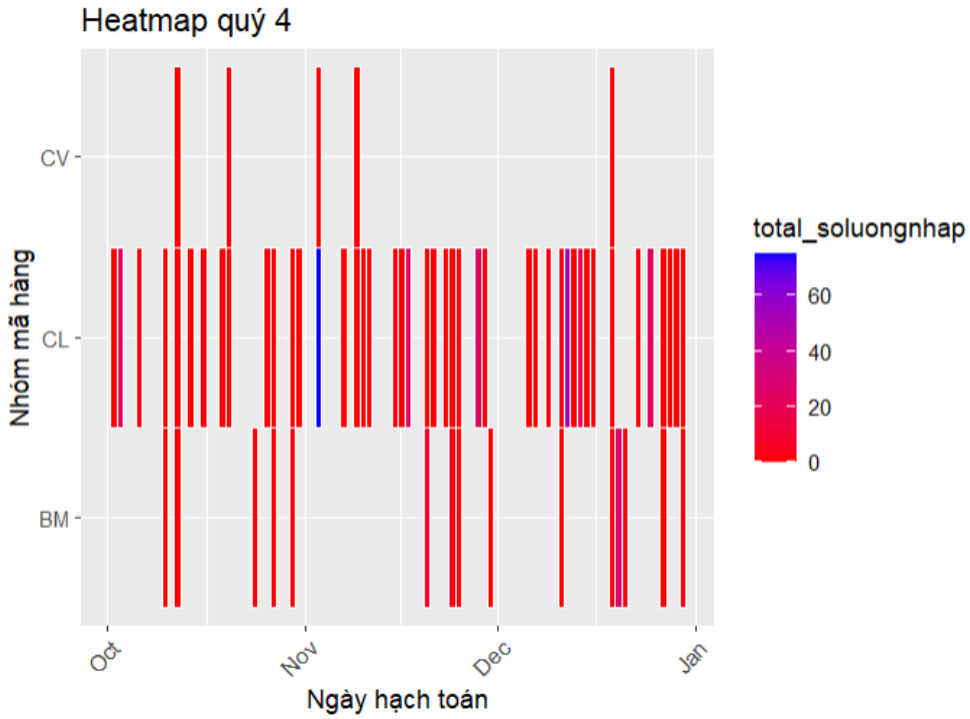
Hình 3.12. Heatmap tổng số lượng nhập theo nhóm mã hàng – quý 2.

Nhận xét: Số lượng nhập bằng 0 trong suốt quý 2. Không có sự thay đổi về số lượng nhập qua các tháng 4, 5 và 6. Giai đoạn này cần có sự kiểm tra lại để xác nhận nguyên nhân không có hoạt động diễn ra.



Hình 3.13. Heatmap tổng số lượng nhập theo nhóm mã hàng – quý 3.

Nhận xét: Nhóm mã hàng CL có hoạt động nhập hàng phong phú nhất và đa dạng về lượng hàng nhập. Nhóm CV và BM có hoạt động nhập hàng ít hơn, với CV có sự ổn định ở mức thấp và BM rất ít hoạt động nhập hàng. Sự biến động về lượng hàng nhập tập trung chủ yếu ở nhóm CL, trong khi CV và BM có sự ổn định nhưng với lượng hàng nhập thấp hơn.



Hình 3.14. Heatmap tổng số lượng nhập theo nhóm mã hàng – quý 4.

Nhận xét: Nhóm CL tiếp tục có hoạt động nhập hàng nhiều và đều đặn nhất. Nhóm CV có nhập hàng đều nhưng với lượng thấp. Nhóm BM có ít hoạt động nhập hàng nhất và lượng nhập rất thấp.

3.2. Phân tích thành phần chính PCA

3.2.1. Giảm chiều và biểu đồ hóa dữ liệu

Sau khi áp dụng phân tích thành phần chính(PCA) lên dữ liệu sản phẩm, ta thu được một biểu đồ biểu diễn dữ liệu trong không gian hai chiều của hai thành phần chính đầu tiên. Biểu đồ này giúp chúng ta nhìn nhận được cách mà quan sát phân bố trong không gian hai chiều được giảm chiều từ không gian nhiều chiều xuống.

```
 # Áp dụng PCA
  pca_result <- prcomp(hanghoa[, c("dongia", "soluongnhap", "giatrinhap",
"soluongxuat",
                                   "giatrixuat", "soluongton", "giatriton",
"dongiaban")], scale = TRUE)

  # Biểu đồ hóa dữ liệu sau khi giảm chiều
  pca_data <- as.data.frame(pca_result$x[, 1:2]) # Lấy 2 thành phần chính đầu tiên
  colnames(pca_data) <- c("PC1", "PC2")

  ggplot(pca_data, aes(x = PC1, y = PC2)) +
    geom_point() +
    xlab("Principal Component 1") +
    ylab("Principal Component 2") +
    ggtitle("PCA on Product Data")
```

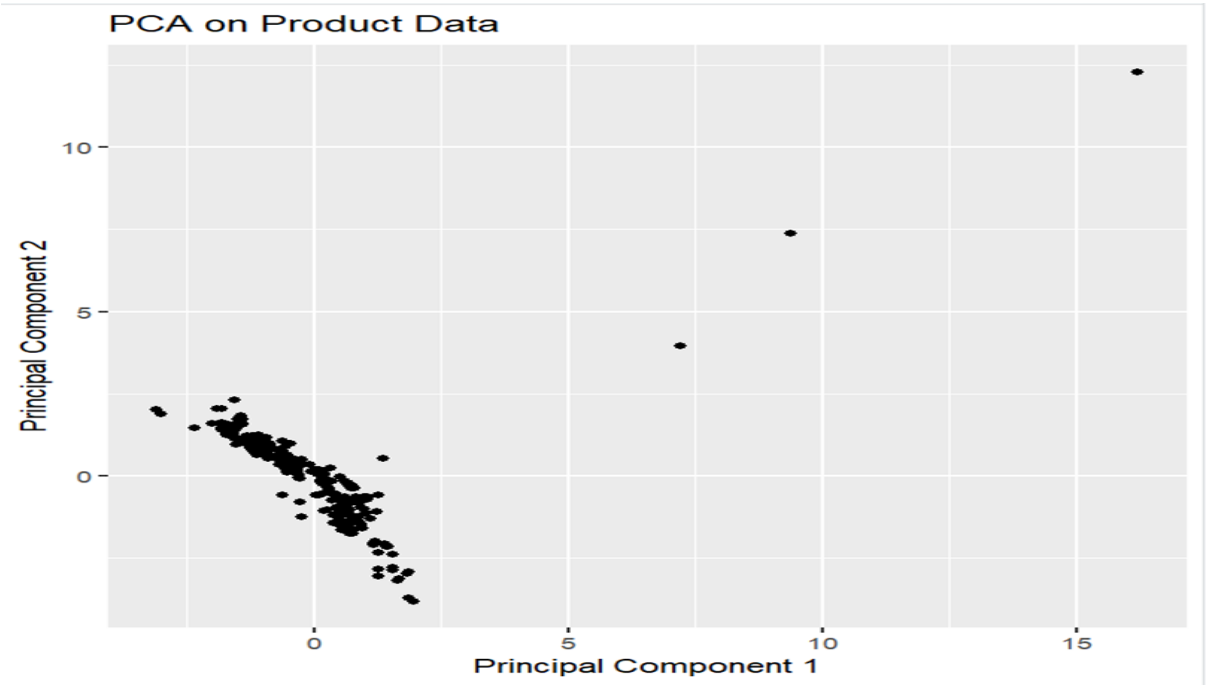
Kết quả:

	PC1	PC2
dongia	-0.17130304	-0.48189152
soluongnhap	0.51589486	-0.19716664
giatrinhap	0.52439463	-0.24732703
soluongxuat	0.01081780	0.51628635
giatrixuat	-0.40080950	-0.09360109
soluongton	0.03892011	0.51041936
giatriton	0.19781726	-0.23476829
dongiaban	-0.47760476	-0.27708914

Nhận xét:

- **Thành phần chính 1(PC1):** Là thành phần chính đầu tiên mà PCA xác định, được hiểu là một hình chiếu của dữ liệu gốc xuống một chiều mới sao cho nó giải thích được phần lớn biến thiên trong dữ liệu. PC1 thường được biểu thị cho sự biến động lớn nhất trong dữ liệu.

- **Thành phần chính 2(PC2):** Là thành phần chính thứ 2 mà PCA xác định, thường là hình chiếu của dữ liệu gốc xuống một chiều mới khác, không tương quan với PC1, nhưng vẫn giải thích được một phần của biến thiên trong dữ liệu.



Hình 3.15. Biểu đồ hóa dữ liệu sau khi giảm chiều.

Nhận xét:

- Biểu diễn không gian mới: Biểu đồ này biểu diễn dữ liệu trong một không gian mới, được tạo ra từ việc giảm chiều dữ liệu ban đầu thành chỉ hai thành phần chính (PC1 và PC2) bằng PCA.
- Phân tán của dữ liệu: Các điểm trên biểu đồ đại diện cho các quan sát trong dữ liệu sau khi đã được biểu diễn trong không gian mới. Sự phân tán của các điểm trên biểu đồ có thể cung cấp thông tin về cách mà dữ liệu được phân tán trong không gian mới này.
- Mối quan hệ giữa các quan sát: Các điểm gần nhau trên biểu đồ thường biểu thị sự tương đồng giữa các quan sát trong dữ liệu. Nếu các điểm cách xa nhau, điều này có thể ngụ ý rằng các quan sát tương ứng không giống nhau.
- Hiểu rõ hơn về cấu trúc của dữ liệu: Biểu đồ này giúp hiểu rõ hơn về cấu trúc của dữ liệu sau khi đã giảm chiều. Bằng cách hiểu cách các quan sát được phân tán trong không gian mới, chúng ta có thể nhận biết các nhóm, mẫu, hoặc mối quan hệ giữa các điểm dữ liệu.
- Tiêu đề và nhãn trục: Tiêu đề và nhãn trục trên biểu đồ giúp làm rõ nội dung của biểu đồ, giúp người đọc hiểu được biểu đồ đang biểu diễn thông tin gì.

3.2.2. Thành phần chính với tỷ lệ phương sai

a. Tỷ lệ phương sai

```
 # In ra các thành phần chính và tỷ lệ phương sai giải thích
print("Principal Components:")
print(pca_result$rotation[, 1:2])
print("Proportion of Variance Explained:")
print(pca_result$sdev^2 / sum(pca_result$sdev^2))
```

Kết quả:

In ra tỷ lệ phương sai giải thích của mỗi thành phần chính.
> print("Proportion of Variance Explained:")
[1] "Proportion of Variance Explained:"
> print(pca_result$sdev^2 / sum(pca_result$sdev^2))
[1] 0.33082648 0.20973715 0.14719306 0.12584329 0.07840875 0.06144319
[7] 0.03435304 0.01219504

Nhận xét:

- Hiển thị thông tin quan trọng: Bằng cách in ra tỷ lệ phương sai giải thích của từng thành phần chính, chúng ta có cái nhìn tổng quan về mức độ mà mỗi thành phần chính giải thích phương sai của dữ liệu gốc. Điều này giúp chúng ta hiểu được vai trò của mỗi thành phần chính trong việc biểu diễn và mô tả dữ liệu.
- Xác định thành phần quan trọng: Các giá trị tỷ lệ phương sai giải thích thấp hơn có thể cho thấy rằng các thành phần chính tương ứng không đóng góp nhiều vào việc biểu diễn đặc trưng của dữ liệu. Trong trường hợp này, chúng ta có thể cân nhắc loại bỏ hoặc giảm số lượng các thành phần chính này mà không làm mất đi quá nhiều thông tin.

b. Phần trăm phương sai

Sau khi xác định các thành phần chính PCA, chúng ta quan tâm đến việc hiểu độ quan trọng của mỗi biến đối với mỗi thành phần chinh. Điều này giúp chúng ta biết được độ quan trọng của mỗi biến đối với mỗi thành phần chính PC1, PC2. Do đó, chúng ta biết được biến nào đóng vai trò quan trọng nhất trong việc xác định biến đổi của dữ liệu trong từng thành phần chính.

```
# Tính toán phần trăm phương sai giải thích bởi thành phần chính 1 và 2
variance_explained_PC1 <- (pca_result$sdev[1]^2 / sum(pca_result$sdev^2)) * 100
variance_explained_PC2 <- (pca_result$sdev[2]^2 / sum(pca_result$sdev^2)) * 100
# In ra phần trăm phương sai giải thích
print(paste("Phần trăm phương sai giải thích bởi thành phần chính 1:",
variance_explained_PC1, "%"))
print(paste("Phần trăm phương sai giải thích bởi thành phần chính 2:",
variance_explained_PC2, "%"))
```

Kết quả:

Thành phần chính 1(PC1):

- Biến quan trọng: “soluongnhap” và “giatrinhap”

- hai biến trên quan trọng với PC1. Đóng vai trò quan trọng trong việc xác định biến đổi của dữ liệu trong thành phần chính PC1, hai biến trên có trọng số cao trong việc hình thành nên thành phần chính 1(PC1) ảnh hưởng một cách mạnh mẽ đến sự biến đổi của dữ liệu theo hướng của thành phần chính 1(PC1)
- Tỷ lệ phương sai Giải thích: Phần trăm phương sai giải thích bởi PC1 cũng cung cấp thông tin về mức độ biến động của dữ liệu mà PC1 có khả năng giải thích được

[1] "Phần trăm phương sai giải thích bởi thành phần chính 1: 33.0826479962811 %"

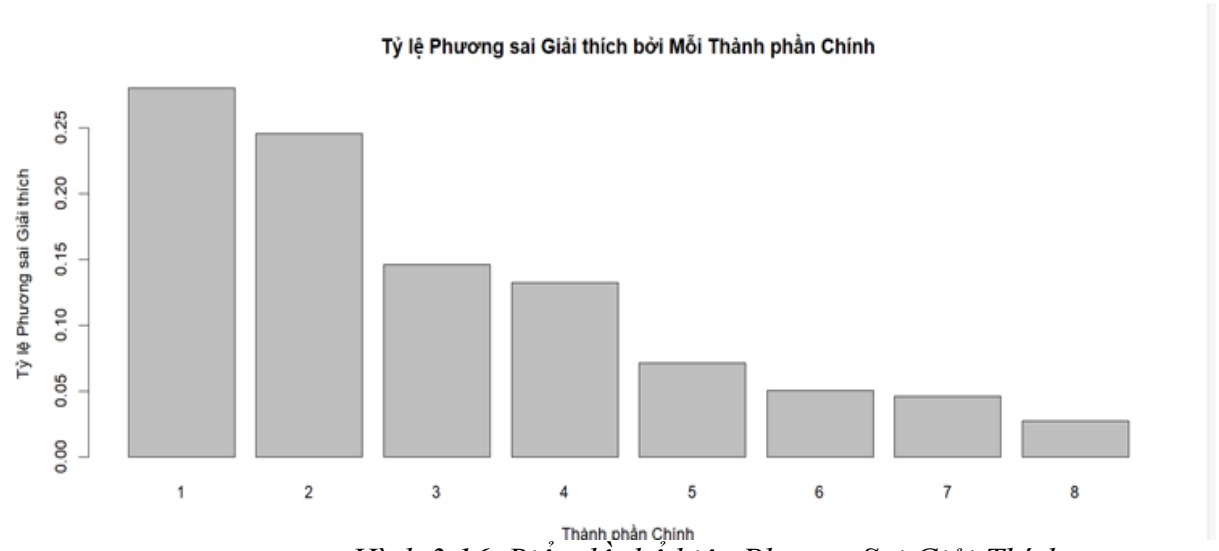
Thành phần chính 2(PC2):

- Biến quan trọng: “soluongxuat” va “soluongton”
- hai biến này giống như 2 biến “soluongnhap” và “giatrinhap” của PC1 vì chúng quan trọng trong việc xác định biến đổi dữ liệu trong thành phần chính này. Ảnh hưởng mạnh mẽ trong việc hình thành PC2
- Tỷ lệ phương sai giải thích: phần trăm phương sai giải thích bởi PC2 cung cấp thông tin về mức độ biến động của dữ liệu mà PC2 có khả năng giải thích được.

[1] "Phần trăm phương sai giải thích bởi thành phần chính 2: 20.9737154378661 %"

c. Biểu đồ phần trăm phương sai

```
# Biểu đồ biểu diễn phần trăm phương sai giải thích
prop_var <- pca_result$sdev^2 / sum(pca_result$sdev^2)
barplot(prop_var, names.arg = 1:length(prop_var),
        main = "Tỷ lệ Phương sai Giải thích bởi Mỗi Thành phần Chính",
        xlab = "Thành phần Chính", ylab = "Tỷ lệ Phương sai Giải thích")
```

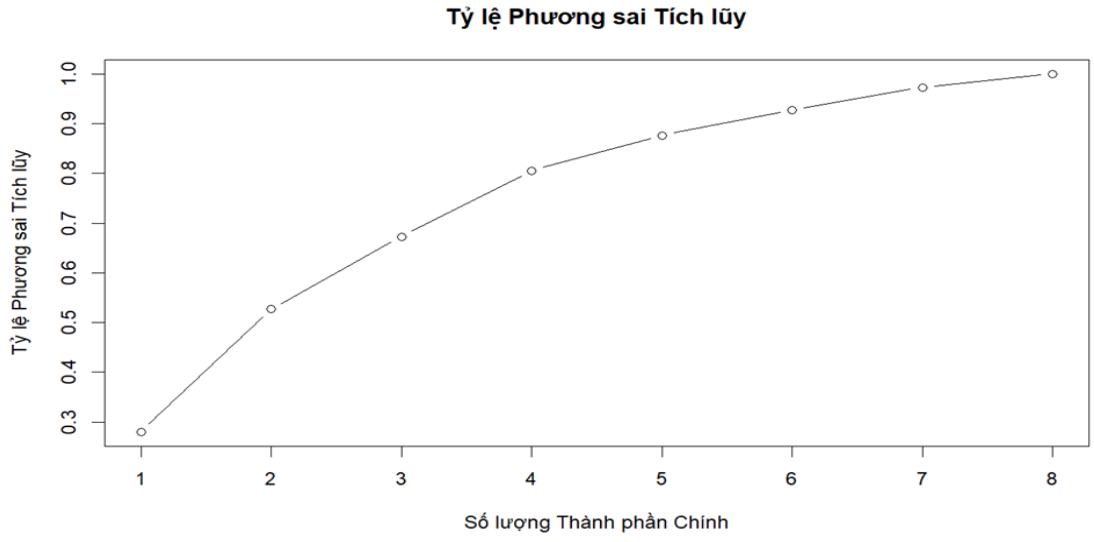


Hình 3.16. Biểu đồ thể hiện Phương Sai Giải Thích.

Nhận xét:

Bằng cách xem xét phần trăm phương sai giải thích bởi mỗi thành phần chính, ta có thể xác định những thành phần nào quan trọng nhất trong việc mô tả sự biến đổi trong dữ liệu. Các thành phần chính với tỷ lệ phương sai lớn hơn đóng góp nhiều hơn vào việc mô tả dữ liệu.

```
# Biểu đồ biểu diễn tỷ lệ phần trăm phương sai tích lũy
cum_var <- cumsum(prop_var)
plot(cum_var, type = "b",
     main = "Tỷ lệ Phương sai Tích lũy",
     xlab = "Số lượng Thành phần Chính", ylab = "Tỷ lệ Phương sai Tích lũy")
```

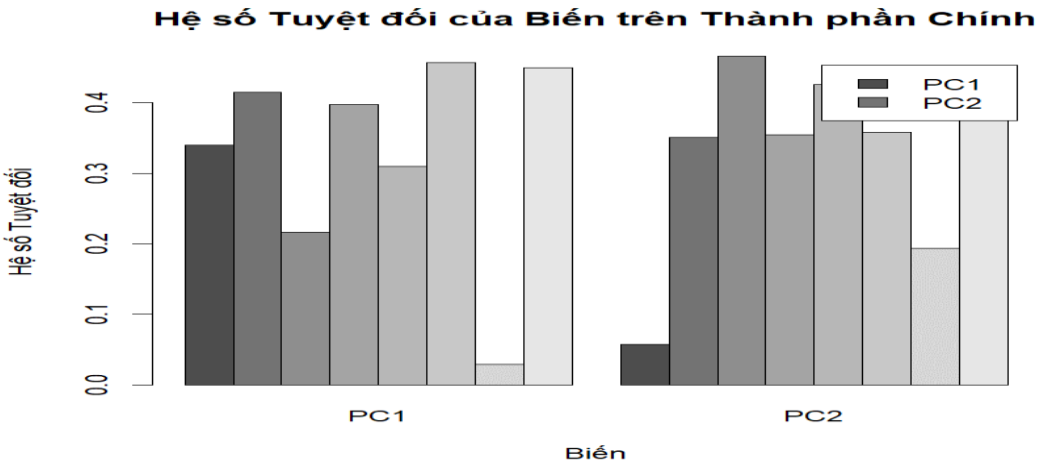


Hình 3.17. Biểu đồ thể hiện Tỷ lệ Phương Sai Tích Lũy.

Nhận xét:

Có thể quyết định số lượng thành phần chính cần giữ lại dựa trên mức độ phương sai mà bạn muốn giữ lại trong dữ liệu. Thông thường, ngưỡng phổ biến là giữ lại đủ số lượng thành phần chính để có tỷ lệ phần trăm phương sai tích lũy đạt được một ngưỡng nhất định.

```
# Biểu đồ biểu diễn hệ số của các biến đối với các thành phần chính
barplot(abs(pca_result$rotation[, 1:2]), beside = TRUE,
        main = "Hệ số Tuyệt đối của Biến trên Thành phần Chính",
        xlab = "Biến", ylab = "Hệ số Tuyệt đối",
        legend.text = c("PC1", "PC2"))
```



Hình 3.18. Biểu đồ biểu diễn hệ số tuyệt đối của các biến đối với các thành phần chính.

Nhận xét:

- Các biến có hệ số tuyệt đối cao trên thanh cột tương ứng với một hoặc cả hai thành phần chính có ảnh hưởng lớn đến sự biến thiên trong dữ liệu. Điều này cho thấy rằng các biến này có sự đóng góp quan trọng trong việc phân chia dữ liệu vào các thành phần chính.
- Bằng cách so sánh hệ số của các biến trên các thành phần chính khác nhau, ta có thể nhận biết mức độ tương quan giữa các biến và các thành phần chính. Các biến có hệ số tương đối lớn trên cả hai thành phần chính có thể có sự tương quan cao với cả hai thành phần chính.
- Biểu đồ cho phép phân biệt rõ ràng giữa các biến dựa trên mức độ ảnh hưởng của chúng đối với các thành phần chính. Các biến với hệ số tuyệt đối lớn hơn thường có ảnh hưởng lớn hơn đối với sự biến đổi của dữ liệu.
- Biểu đồ giúp phân tích mối quan hệ giữa các biến và các thành phần chính trong PCA, giúp hiểu rõ hơn về cách mà các biến ảnh hưởng đến sự biến thiên trong dữ liệu.

3.2.3. Phân tích biến quan trọng và tương quan

a. Biến quan trọng

```
# Xác định các biến quan trọng gần với 1 hoặc -1
important_variables_PC1 <- names(which(abs(pca_components[, 1]) >= 0.5)) # Thay
0.5 bằng ngưỡng phù hợp
important_variables_PC2 <- names(which(abs(pca_components[, 2]) >= 0.5)) # Thay
0.5 bằng ngưỡng phù hợp

# In ra các biến quan trọng cho từng thành phần chính
cat("Biến quan trọng cho thành phần chính 1:\n")
print(important_variables_PC1)
cat("Biến quan trọng cho thành phần chính 2:\n")
print(important_variables_PC2)
```

Kết quả:
Biến quan trọng cho thành phần chính 1:
> print(important_variables_PC1)
[1] "soluongnhap" "giatrinhap"
>
> cat("Biến quan trọng cho thành phần chính 2:\n")
Biến quan trọng cho thành phần chính 2:
> print(important_variables_PC2)
[1] "soluongxuat" "soluongton"

Nhận xét:
Thành phần chính 1 (PC1):

- Các biến quan trọng cho PC1 là "soluongnhap" và "giatrinhap".
- Điều này ngụ ý rằng "soluongnhap" và "giatrinhap" có mức độ ảnh hưởng lớn đến sự biến đổi và phân phối của dữ liệu theo PC1. Có thể hiểu rằng PC1 có thể liên quan chặt chẽ đến các biến liên quan đến lượng nhập và giá trị nhập của sản phẩm.

Thành phần chính 2 (PC2):

- Các biến quan trọng cho PC2 là "soluongxuat" và "soluongton".
- Điều này cho thấy rằng "soluongxuat" và "soluongton" đóng vai trò quan trọng trong việc xác định sự biến đổi và phân phối của dữ liệu theo PC2. Có thể PC2 liên quan đến các biến liên quan đến lượng xuất và lượng tồn kho của sản phẩm.

b. Mối quan hệ tương quan

- Sau khi áp dụng PCA và xác định biến quan trọng cho mỗi thành phần chính, chúng ta tiến hành phân tích tương quan để hiểu rõ về mối quan hệ giữa các biến quan trọng và các thành phần chính.
- **Ma trận tương quan:** Chúng ta tính toán ma trận tương quan giữa các biến quan trọng và các thành phần chính để đo lường mức độ tương quan giữa chúng.Mỗi ô trong ma trận tương quan đại diện cho mức độ tương quan giữa mỗi cặp biến. Việc phân tích ma trận này giúp chúng ta hiểu rõ hơn về mối quan hệ giữa các biến quan trọng và các thành phần chính. Điều này có thể làm rõ về cách mỗi biến ảnh hưởng đến hình thành và biến đổi của PCA.

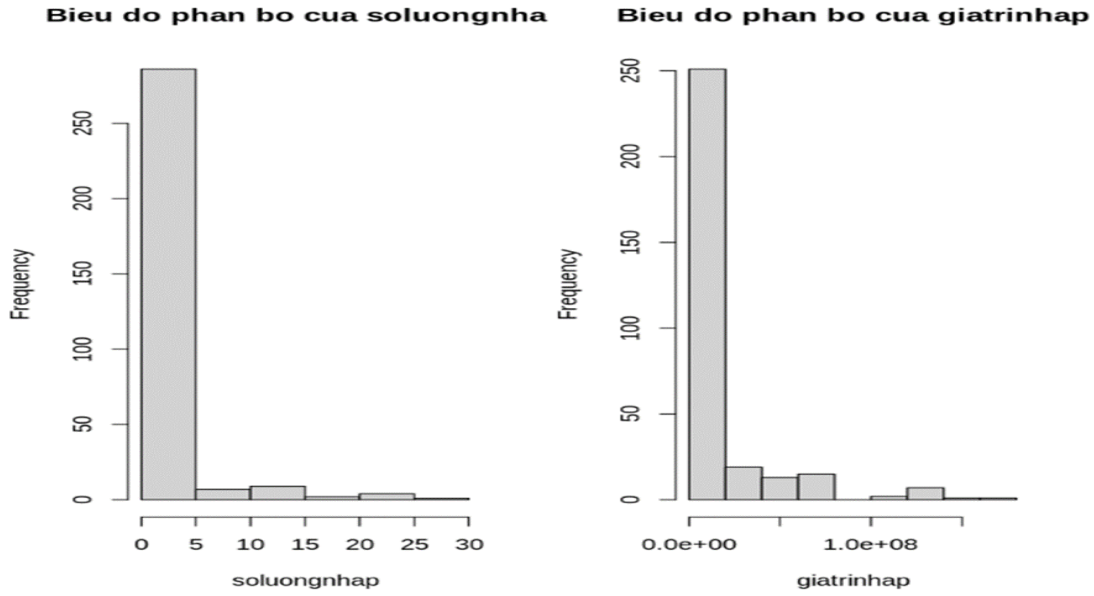
```
# Tính toán ma trận tương quan giữa các biến quan trọng và các thành phần chính
correlation_matrix <- cor(hanghoa[, c(important_variables_PC1,
important_variables_PC2)])
print("Ma trận tương quan giữa các biến quan trọng và các thành phần chính")
print(correlation_matrix)

# Biểu đồ phân bố của các biến quan trọng
par(mfrow = c(1,2))
for (var in important_variables_PC1) {
  hist(hanghoa[[var]], main = paste("Bieu do phan bo cua", var), xlab = var)}
```

Kết quả:

- Nhận thấy rằng, "soluongnhap", và "giatrinhap" đóng vai trò quan trọng trong xác định thành phần chính 1 (PC1) trong khi "soluongxuat" và "soluongton" đóng vai trò quan trọng trong xác định thành phần chính 2(PC2). Điều này giúp chúng ta hiểu rõ hơn, dễ dàng nhận thấy hơn về cách mỗi biến ảnh hưởng đến cấu trúc của dữ liệu sản phẩm.

"Ma trận tương quan giữa các biến quan trọng và các thành phần chính"

	soluongnhap	giatrinhap	soluongxuat	soluongton
soluongnhap	1.00000000	0.85253775	-0.03802536	-0.02449542
giatrinhap	0.85253775	1.00000000	-0.04309972	-0.03216709
soluongxuat	-0.03802536	-0.04309972	1.00000000	0.39007661
soluongton	-0.02449542	-0.03216709	0.39007661	1.00000000



Hình 3.19. Biểu đồ phân bố của các biến quan trọng.

Nhận xét:

Biến quan trọng cho thành phần chính 1 (PC1):

- Biểu đồ phân bố của "soluongnhap" và "giatrinhap" cho thấy mức độ phân phối của các biến này trong dữ liệu.
- Có thể quan sát hình dạng và phân phối của các biến để hiểu được đặc điểm của chúng và cách chúng ảnh hưởng đến thành phần chính 1.

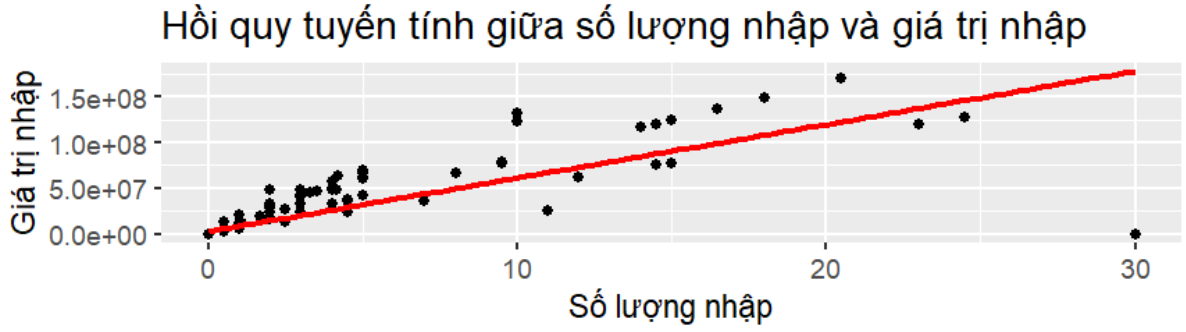
Biến quan trọng cho thành phần chính 2 (PC2):

- Biểu đồ phân bố của "soluongxuat" và "soluongton" cũng cho thấy cách mà các biến này phân phối trong dữ liệu.
- So sánh phân bố của các biến này giúp hiểu sâu hơn về cách chúng ảnh hưởng đến thành phần chính 2.
- Bằng cách phân tích biểu đồ phân bố của các biến quan trọng cho từng thành phần chính, chúng ta có thể đánh giá được đặc điểm và ảnh hưởng của chúng đối với cấu trúc của dữ liệu trong PCA.

3.2.4 Phân tích Hồi Quy

- Hồi quy tuyến tính: Chúng ta tiến hành phân tích hồi quy để kiểm tra mối quan hệ tuyến tính giữa các biến quan trọng và xác định yếu tố ảnh hưởng đến biến phụ thuộc.

```
#Phân tích hồi quy giữa các thành phần chính PC1
# Tạo mô hình hồi quy tuyến tính
lm_model <- lm(soluongnhap ~ giatrinhap, data = hanghoa)

# Hiển thị kết quả của mô hình
summary(lm_model)
# Biểu đồ scatter plot cho biến nhập và tồn
ggplot(hanghoa, aes(x = soluongnhap, y = giatrinhap)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(x = "Số lượng nhập", y = "Giá trị nhập") +
  ggtitle("Hồi quy tuyến tính giữa số lượng nhập và giá trị nhập")
```

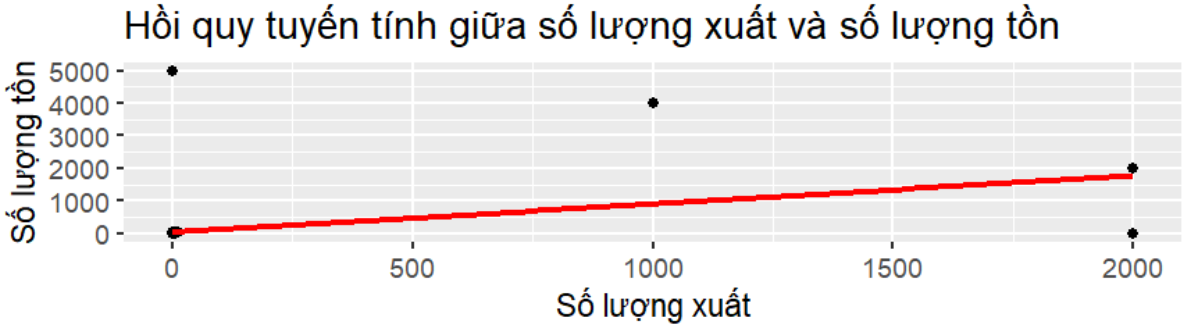


Hình 3.20. Biểu đồ trực quan cho biến số lượng nhập và giá trị.

Kết quả của PC1:

- **Hồi quy giữa Số lượng xuất và số lượng tồn:** Mô hình hồi quy tuyến tính phản ánh mối quan hệ giữa "soluongxuat" và "soluongton". Kết quả này giúp chúng ta hiểu được cách biến "soluongton" ảnh hưởng đến "soluongxuat" và mức độ ảnh hưởng của nó.

```r
#Phân tích hồi quy giữa các thành phần chính PC2
# Tạo mô hình hồi quy tuyến tính
lm_model <- lm(soluongxuat ~ soluongton, data = hanghoa)

# Hiển thị kết quả của mô hình
summary(lm_model)
# Biểu đồ scatter plot cho biến nhập và tồn
ggplot(hanghoa, aes(x = soluongxuat, y = soluongton)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(x = "Số lượng xuất", y = "Số lượng tồn") +
  ggtitle("Hồi quy tuyến tính giữa số lượng xuất và số lượng tồn")
```



Hình 3.21. Biểu đồ trực quan cho biến số lượng xuất và số lượng tồn.

Kết quả của PC2:

- **Hồi quy giữa số lượng nhập và giá trị nhập:** Mô hình hồi quy tuyến tính cho thấy mối quan hệ giữa "soluongnhap" và "giatrinhap". Kết quả của mô hình giúp chúng ta hiểu được cách biến "giatrinhap" ảnh hưởng đến "soluongnhap" và mức độ ảnh hưởng của nó.

Nhận xét: Phân tích hồi quy này giúp chúng ta hiểu rõ hơn về mối quan hệ giữa các biến quan trọng và làm rõ cách chúng tác động lẫn nhau trong dữ liệu sản phẩm. Nó cũng cung cấp cái nhìn sâu hơn về cách mỗi biến ảnh hưởng đến biến phụ thuộc, chúng ta sẽ hiểu rõ hơn về cấu trúc và mối quan hệ trong dữ liệu.

KẾT LUẬN

Với báo cáo Đồ án 1, chúng em đã hoàn thành:

Giới thiệu tổng quan về thư viện ggplot và phân tích thống kê (Chương 1): Trong chương này, chúng em đã giới thiệu về thư viện ggplot trong ngôn ngữ R và khái quát về phân tích thống kê. Điều này đã giúp chúng em xây dựng nền tảng cơ bản để tiếp cận và hiểu rõ hơn về công cụ và phương pháp được sử dụng trong phân tích dữ liệu.

Phân tích thăm dò dữ liệu (Chương 2): Trong chương này, chúng em đã thực hiện phân tích thăm dò dữ liệu bằng cách sử dụng các phương pháp và công cụ thống kê. Qua đó, chúng em đã phát hiện ra các mô hình, xu hướng và thông tin quan trọng từ dữ liệu, từ đó đưa ra những nhận định và suy luận hợp lý.

Trực quan dữ liệu (Chương 3): Chương này tập trung vào việc trực quan hóa dữ liệu bằng cách sử dụng thư viện ggplot. Chúng em đã biến các số liệu và thông tin từ dữ liệu thành các biểu đồ và đồ thị sinh động, giúp hiểu rõ hơn về các mối quan hệ và xu hướng trong dữ liệu.

Đồ án của chúng em chỉ dừng lại ở mức ứng dụng đơn giản và chưa thực hiện được triển khai thực tế, nhưng chúng em vẫn hy vọng nhận được ý kiến đóng góp xây dựng từ thầy/cô. Chúng em xin cảm ơn sự hỗ trợ và hướng dẫn của thầy trong suốt quá trình thực hiện đồ án này.

Chúng em sẽ tiếp tục nghiên cứu và áp dụng các mô hình tiên tiến hơn, phát triển mở rộng hơn.

Chúng em xin chân thành cảm ơn!

TÀI LIỆU THAM KHẢO

[1]. Trần Chí Lê, Nguyễn Thị Hạnh Lê (2024), *Tài liệu học tập Đồ án 1: Trực quan hóa dữ liệu bằng R*, Khoa Khoa Học Ứng Dụng, Trường Đại học Kinh tế - Kỹ thuật Công nghiệp.

[2]. Trần Thị Kim Thanh, Trần Chí Lê (2023), *Tài liệu học tập Thống kê Toán học cho ngành Khoa học dữ liệu*, Khoa Khoa Học Ứng Dụng, Trường Đại học Kinh tế - Kỹ thuật Công nghiệp.

[3]. Bắt đầu với trực quan hóa dữ liệu trong R.

Link: https://ichi.pro/vi/bat-dau-voi-truc-quan-hoa-du-lieu-trong-r-54458832168301

[4]. Modern Data Visualization with R – GitHub Pages

Link: https://rkabacoff.github.io/datavis/Time.html#TimeSeries

[5]. Thống kê mô tả với R.

Link: https://mosl.vn/thong-ke-mo-ta-trong-voi-r/

[6]. Hướng dẫn phân tích số liệu và vẽ biểu đồ bằng R.

Link:
https://ykhoa.net/baigiang/GS_Nguyen_Van_Tuan/phuong_phap_thong_ke_r/index.htm

[7]. Hướng dẫn sử dụng R trong phân tích PCA, MCA.

Link: https://rstudio-pubs-static.s3.amazonaws.com/1068824_1bbecd4b6ecc4d4a91588b6f6b7b2573.html

[8]. Dữ liệu hàng hóa lấy từ Công ty TNHH phân phối lân, đạm VADFO.