**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC ĐẠI NAM**

Logo, company name

Description automatically generated

**BÀI TẬP LỚN**

**TÊN HỌC PHẦN: DỮ LIỆU LỚN**

**ĐỀ TÀI: PHÂN TÍCH XU HƯỚNG THUÊ PHÒNG KHÁCH SẠN**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Mã Sinh Viên | Họ và Tên | Ngày Sinh | Lớp |
| 1 | 1671020083 | Nguyễn Tiến Đạt | 10/05/2004 | CNTT 16-04 |
| 2 | 1671020094 | Nguyễn Văn Giang | 27/11/2004 | CNTT 16-04 |
| 3 | 1671020212 | Nguyễn Hoài Nam | 26/05/2003 | CNTT 16-04 |

**Hà Nội, năm 2025**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC ĐẠI NAM**

Logo, company name

Description automatically generated

**BÀI TẬP LỚN**

**TÊN HỌC PHẦN: DỮ LIỆU LỚN**

**ĐỀ TÀI: PHÂN TÍCH XU HƯỚNG THUÊ PHÒNG KHÁCH SẠN**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Mã Sinh Viên | Họ và Tên | Ngày Sinh | Điểm | |
| Bằng Số | Bằng Chữ |
| 1 | 1671020083 | Nguyễn Tiến Đạt | 10/05/2004 |  | 1 |
| 2 | 1671020094 | Nguyễn Văn Giang | 27/11/2004 |  | 2 |
| 3 | 1671020212 | Nguyễn Hoài Nam | 26/05/2003 |  | 3 |

CÁN BỘ CHẤM THI 1 CÁN BỘ CHẤM THI 2

**Hà Nội, năm 2025**

**LỜI NÓI ĐẦU**

Trong bối cảnh ngày càng gia tăng cạnh tranh trong ngành dịch vụ và du lịch, việc hiểu rõ hành vi của khách hàng và tối ưu hóa hoạt động kinh doanh là nhu cầu thiết yếu đối với các khách sạn. Nhờ sự phát triển của công nghệ thông tin và khả năng thu thập, phân tích dữ liệu lớn, doanh nghiệp có thể nhận diện xu hướng, từ đó đề xuất ra được các chiến lược kinh doanh hiệu quả.

Báo cáo này nhằm phân tích dữ liệu khách thuê phòng khách sạn dựa trên tệp dữ liệu gồm 500.000 đối tượng, bao gồm các thuộc tính như mã, số lượng người lớn và trẻ em, số đêm lại cuối tuần và trong tuần, loại bữa ăn, loại phòng đặt, nhu cầu bãi đỗ xe, thời gian đặt trước, ngày tháng năm đến, phân khúc thị trường, tình trạng khách quen, số lần hủy đặt trước, số lần đặt trước không bị hủy, giá trung bình phòng và số yêu cầu đặc biệt. Thông qua việc phân tích dữ liệu bằng Spark và R, chúng em nhằm tìm ra những mẫu hình quan trọng, dự đoán xu hướng hành vi của khách hàng, cũng như đề xuất những đề nghị giúp tối ưu hóa doanh thu và chất lượng dịch vụ khách sạn.

**MỤC LỤC**

[LỜI NÓI ĐẦU 3](#_Toc192982126)

[**CHƯƠNG 1: GIỚI THIỆU** 7](#_Toc192982127)

[**1.1 Bối cảnh.** 7](#_Toc192982128)

[**1.2 Xu hướng.** 7](#_Toc192982129)

[**1.3 Phân tích.** 8](#_Toc192982130)

[**1.4 Công cụ sử dụng.** 9](#_Toc192982131)

[**CHƯƠNG 2 : XÂY DỰNG CHƯƠNG TRÌNH.** 14](#_Toc192982132)

[**2.1** **Giới thiệu tập dữ liệu.** 14](#_Toc192982133)

[**2.2 Xây dựng code.** 15](#_Toc192982134)

[**2.2.1 Kết nối với Spark.** 15](#_Toc192982135)

[**2.2.2 Đọc dữ liệu từ file CSV.** 16](#_Toc192982136)

[**2.2.3 Tiền xử lý dữ liệu.** 16](#_Toc192982137)

[**2.2.4 Chia dữ liệu thành tập huấn luyện và tập kiểm tra.** 17](#_Toc192982138)

[**2.2.5 Xây dựng mô hình Random Forest.** 17](#_Toc192982139)

[**2.2.6 Dự đoán và đánh giá mô hình.** 17](#_Toc192982140)

[**2.2.7 Tạo biểu đồ tròn.** 18](#_Toc192982141)

[**2.2.8 Tạo biểu đồ cột dọc.** 19](#_Toc192982142)

[**2.2.9 Tạo biểu đồ thanh ngang.** 21](#_Toc192982143)

[**CHƯƠNG 3 : KẾT QUẢ CHƯƠNG TRÌNH** 24](#_Toc192982144)

[**3.1** **Biểu đồ tròn.** 24](#_Toc192982145)

[**3.1.1 Giải thích.** 24](#_Toc192982146)

[**3.1.2** **Phân tích kết quả.** 24](#_Toc192982147)

[**3.2 Biểu đồ cột dọc.** 25](#_Toc192982148)

[**3.2.1 Giải thích.** 25](#_Toc192982149)

[**3.2.2 Phân tích kết quả.** 26](#_Toc192982150)

[**3.3 Biểu đồ thanh ngang.** 26](#_Toc192982151)

[**3.3.1 Giải thích.** 26](#_Toc192982152)

[**3.3.2 Phân tích kết quả.** 27](#_Toc192982153)

[**KẾT LUẬN** 28](#_Toc192982154)

[**DANH MỤC TÀI LIỆU THAM KHẢO** 29](#_Toc192982155)

**CHƯƠNG 1: GIỚI THIỆU**

**1.1 Bối cảnh.**



Ngành khách sạn và du lịch đã và đang phát triển nhanh chóng nhờ sự gia tăng nhu cầu du lịch toàn cầu, sự phục hồi mạnh mẽ sau đại dịch, cùng với sự hỗ trợ của công nghệ số và dữ liệu lớn, giúp các doanh nghiệp tối ưu hóa dịch vụ, nâng cao trải nghiệm khách hàng và cạnh tranh hiệu quả trong bối cảnh thị trường ngày càng sôi động.

**1.2 Xu hướng.**

Những trải nghiệm mang tính cá nhân thường được xây dựng dựa trên hiểu biết về khách hàng của khách sạn. Điều này đồng nghĩa: khi hiểu biết càng nhiều, càng rõ về khách hàng, khách sạn càng có thể thu hút họ đặt phòng nhiều hơn. Để bắt đầu, hãy tìm hiểu về phân khúc thị trường khách sạn để xây dựng các chiến lược phù hợp với từng nhóm khách hàng của khách sạn.



Lý do các du khách hiện nay thích đặt phòng tại các khách sạn độc lập là bởi chúng cung cấp những trải nghiệm tích cực thân thiện, được cá nhân hóa theo từng du khách. Chất lượng này khác biệt, vượt trội khi so với trải nghiệm tương tự tại các [khách sạn boutique](https://bluejaypms.com/article/lam-the-nao-kinh-doanh-khach-san-boutique-tro-nen-thanh-cong-208), nhà nghỉ, dorm hay bed breakfast (loại hình lưu trú chỉ bao gồm giường ngủ và bữa sáng) từ các khách sạn lớn.



Điều này cũng đồng nghĩa rằng – nếu khách sạn tạo được những trải nghiệm cá nhân hóa tích cực cho du khách, thì có thể thu hút được nhiều lượt đặt phòng, cải thiện giá phòng trung bình hàng ngày và có nhiều đánh giá trực tuyến tích cực hơn.

Những trải nghiệm mang tính cá nhân thường được xây dựng dựa trên hiểu biết về khách hàng của khách sạn. Điều này đồng nghĩa: khi hiểu biết càng nhiều, càng rõ về khách hàng, khách sạn càng có thể thu hút họ đặt phòng nhiều hơn.

**1.3 Phân tích.**

Kéo theo sự gia tăng nhanh chóng của lượng dữ liệu từ các lượt đặt phòng, nhu cầu lưu trú và hành vi khách hàng. Nếu biết tận dụng nguồn dữ liệu này, các khách sạn có thể tối ưu hóa chiến lược kinh doanh, quản lý phòng hiệu quả và cải thiện trải nghiệm cho khách hàng. Đặc biệt, trong thời đại công nghệ, việc phân tích dữ liệu lớn giúp doanh nghiệp dự đoán xu hướng đặt phòng, điều chỉnh giá cả hợp lý và nâng cao khả năng cạnh tranh trên thị trường.



Việc khai thác và phân tích dữ liệu này không chỉ giúp hiểu rõ xu hướng của khách hàng mà còn hỗ trợ cải thiện dịch vụ, tối ưu doanh thu. Chính vì vậy, việc nghiên cứu xu hướng thuê phòng khách sạn trở thành một chủ đề quan trọng, giúp nhận diện các yếu tố ảnh hưởng đến quyết định đặt phòng, từ đó đưa ra những dự đoán hữu ích cho ngành.



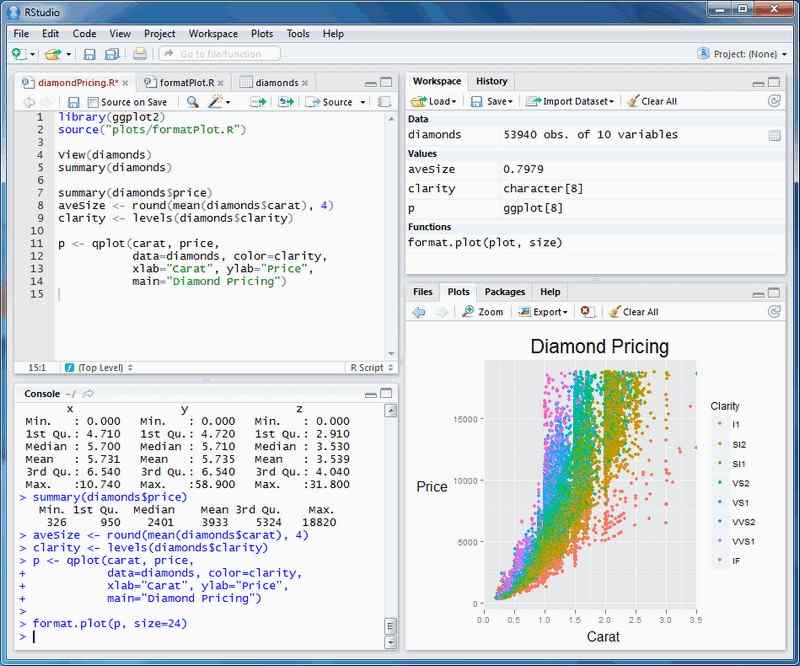
Phân tích dữ liệu trong lĩnh vực này không đơn thuần là quan sát số liệu mà còn đòi hỏi những phương pháp xử lý phù hợp để làm rõ mối quan hệ giữa các yếu tố như giá phòng, thời gian lưu trú, loại hình khách hàng hay tỉ lệ hủy đặt phòng. Thông qua quá trình này, có thể xác định được những đặc điểm chung trong hành vi đặt phòng, từ đó giúp các khách sạn điều chỉnh chiến lược kinh doanh sao cho hiệu quả hơn.

**1.4 Công cụ sử dụng.**



RStudio là một môi trường phát triển tích hợp (IDE) phổ biến và mạnh mẽ dành cho ngôn ngữ lập trình R. Nó cung cấp một giao diện thân thiện với người dùng để viết, chạy và gỡ lỗi mã R, cũng như các công cụ để quản lý dự án, trực quan hóa dữ liệu và tạo báo cáo.

RStudio có giao diện trực quan và thân thiện, giúp người dùng dễ dàng làm việc với ngôn ngữ R. Môi trường này chia thành nhiều khu vực như console để chạy lệnh trực tiếp, editor để viết và chỉnh sửa code, environment để quản lý biến và plots để hiển thị biểu đồ. Nhờ cách bố trí này, lập trình viên có thể tổ chức công việc khoa học hơn, dễ dàng theo dõi dữ liệu và kết quả phân tích.

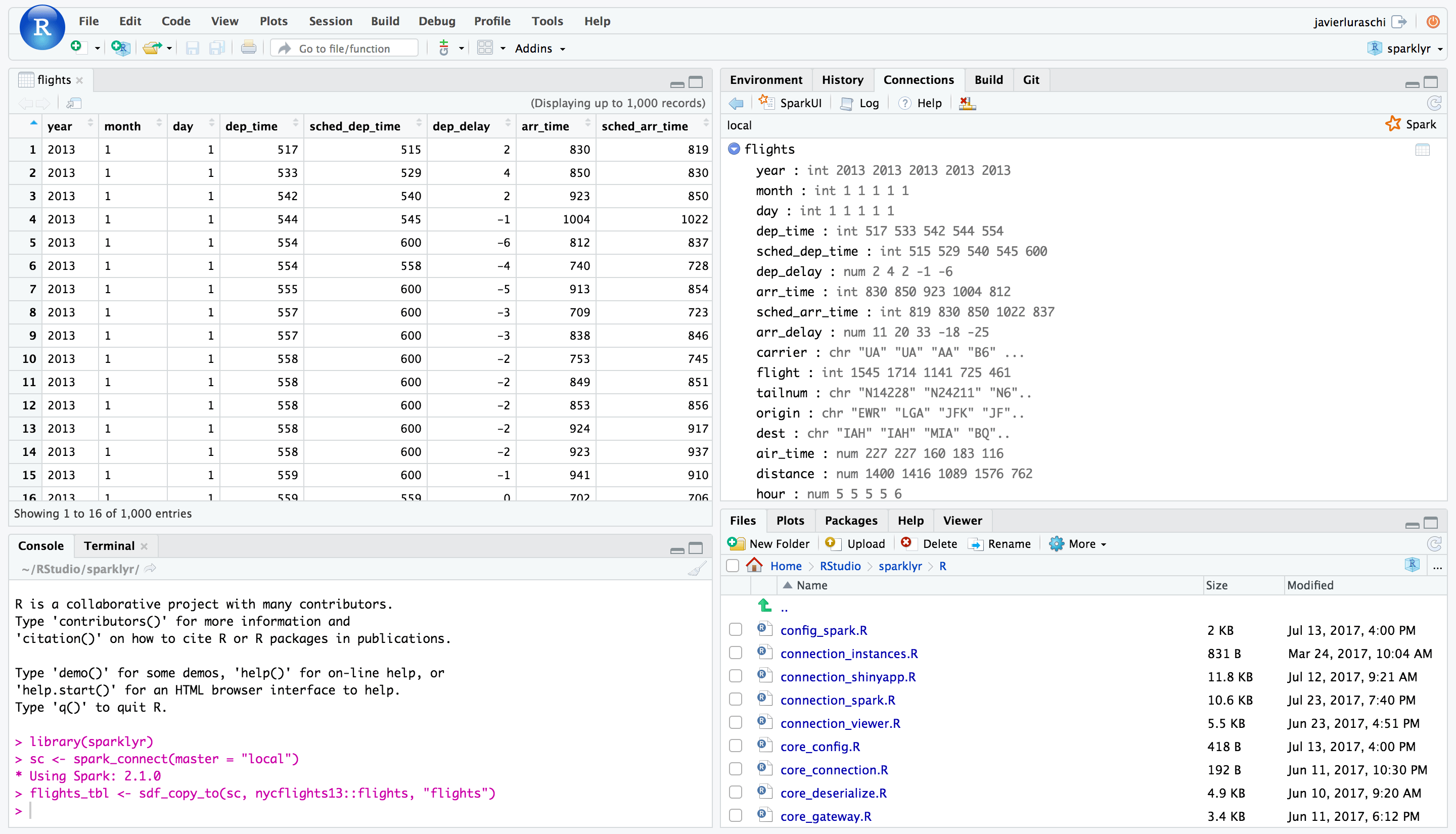


Bên cạnh khả năng hỗ trợ lập trình, RStudio còn mạnh mẽ trong việc phân tích và trực quan hóa dữ liệu. Người dùng có thể vẽ biểu đồ, tạo bảng số liệu và theo dõi kết quả ngay trong giao diện mà không cần chuyển đổi công cụ khác. RStudio còn hỗ trợ các gói như shiny để xây dựng ứng dụng tương tác, giúp trình bày dữ liệu một cách sinh động và trực quan. Điều này đặc biệt hữu ích cho các nhà khoa học dữ liệu và chuyên gia phân tích khi làm việc với lượng dữ liệu lớn và phức tạp.

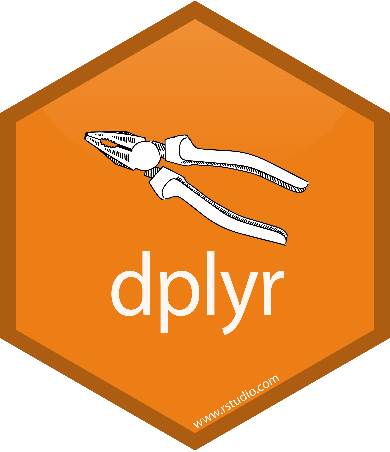


Thư viện sparklyr là một thư viện trong R giúp kết nối và làm việc với Apache Spark, một nền tảng xử lý dữ liệu lớn mạnh mẽ. Thư viện này cho phép người dùng khai thác sức mạnh của Spark ngay trong môi trường RStudio, tận dụng khả năng xử lý phân tán để phân tích dữ liệu lớn một cách hiệu quả. Với sparklyr, người dùng có thể thực hiện các thao tác như kết nối với Spark, xử lý dữ liệu bằng cú pháp dplyr, huấn luyện mô hình học máy và trực quan hóa dữ liệu mà không cần rời khỏi R.

Ưu điểm chính của sparklyr là khả năng làm việc với dữ liệu lớn mà không cần tải toàn bộ dữ liệu vào bộ nhớ. Thay vì xử lý dữ liệu trên RAM, sparklyr tận dụng sức mạnh của Spark để thực hiện các thao tác trên cụm máy chủ hoặc ngay cả trên máy tính cá nhân một cách tối ưu. Điều này đặc biệt hữu ích khi làm việc với hàng triệu hoặc hàng tỷ dòng dữ liệu, giúp giảm thời gian tính toán và tối ưu tài nguyên hệ thống.



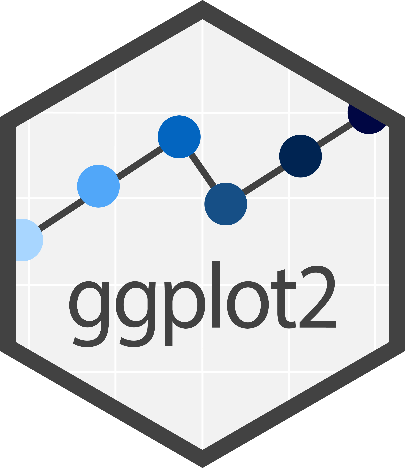
Bên cạnh đó, sparklyr hỗ trợ tích hợp với các thư viện phổ biến như dplyr để thao tác dữ liệu một cách dễ dàng, ggplot2 để trực quan hóa, và cả các mô hình học máy có sẵn trong Spark MLlib. Nhờ sự linh hoạt này, sparklyr trở thành một công cụ quan trọng trong phân tích dữ liệu lớn, giúp các nhà khoa học dữ liệu và lập trình viên có thể khai thác tiềm năng của Apache Spark ngay trong R mà không cần chuyển đổi sang các ngôn ngữ khác như Scala hay Python.



Thư viện dplyr được thiết kế để giúp xử lý và thao tác dữ liệu một cách nhanh chóng và dễ hiểu. Thư viện này cung cấp một bộ công cụ mạnh mẽ để làm việc với dữ liệu dạng bảng (data frames, tibbles) bằng cách sử dụng các hàm trực quan như filter(), select(), mutate(), summarise() và arrange(). Nhờ vào cú pháp đơn giản và trực quan, dplyr giúp rút ngắn thời gian viết code, đồng thời cải thiện tính dễ đọc và bảo trì của mã nguồn.

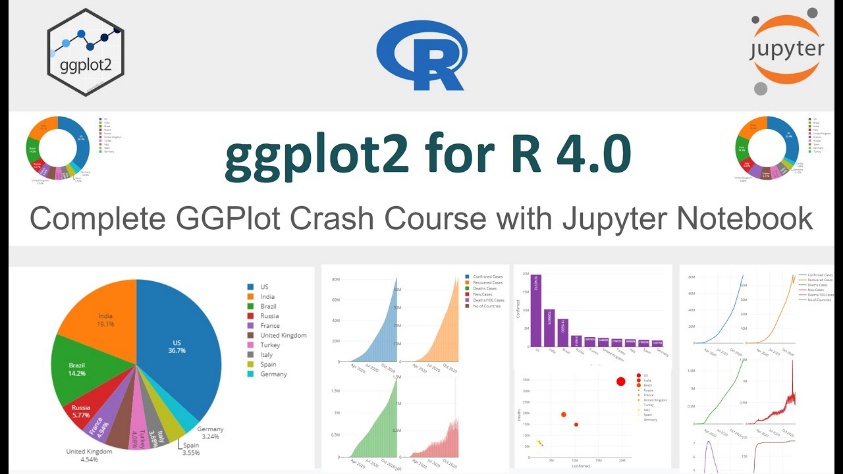


Một trong những điểm mạnh của dplyr là khả năng xử lý dữ liệu nhanh chóng và tối ưu. Thư viện này được xây dựng trên nền tảng C++ thông qua Rcpp, giúp tăng tốc độ thực thi so với các phương pháp truyền thống. Ngoài ra, dplyr còn hỗ trợ làm việc với nhiều nguồn dữ liệu khác nhau, không chỉ giới hạn ở data frames mà còn có thể tương tác với các cơ sở dữ liệu như SQL, Spark thông qua các backend như dbplyr và sparklyr. Điều này giúp người dùng dễ dàng mở rộng khả năng phân tích mà không cần thay đổi quá nhiều trong cú pháp.



Thư viện ggplot2 dùng để trực quan hóa dữ liệu. Được phát triển dựa trên nguyên tắc Grammar of Graphics, ggplot2 cho phép người dùng tạo ra các biểu đồ một cách linh hoạt, có cấu trúc rõ ràng và dễ dàng tùy chỉnh. Thay vì sử dụng các lệnh đơn lẻ để vẽ đồ thị, ggplot2 tổ chức việc tạo biểu đồ theo từng lớp (layers), giúp người dùng dễ dàng xây dựng, mở rộng và chỉnh sửa đồ thị theo nhu cầu.

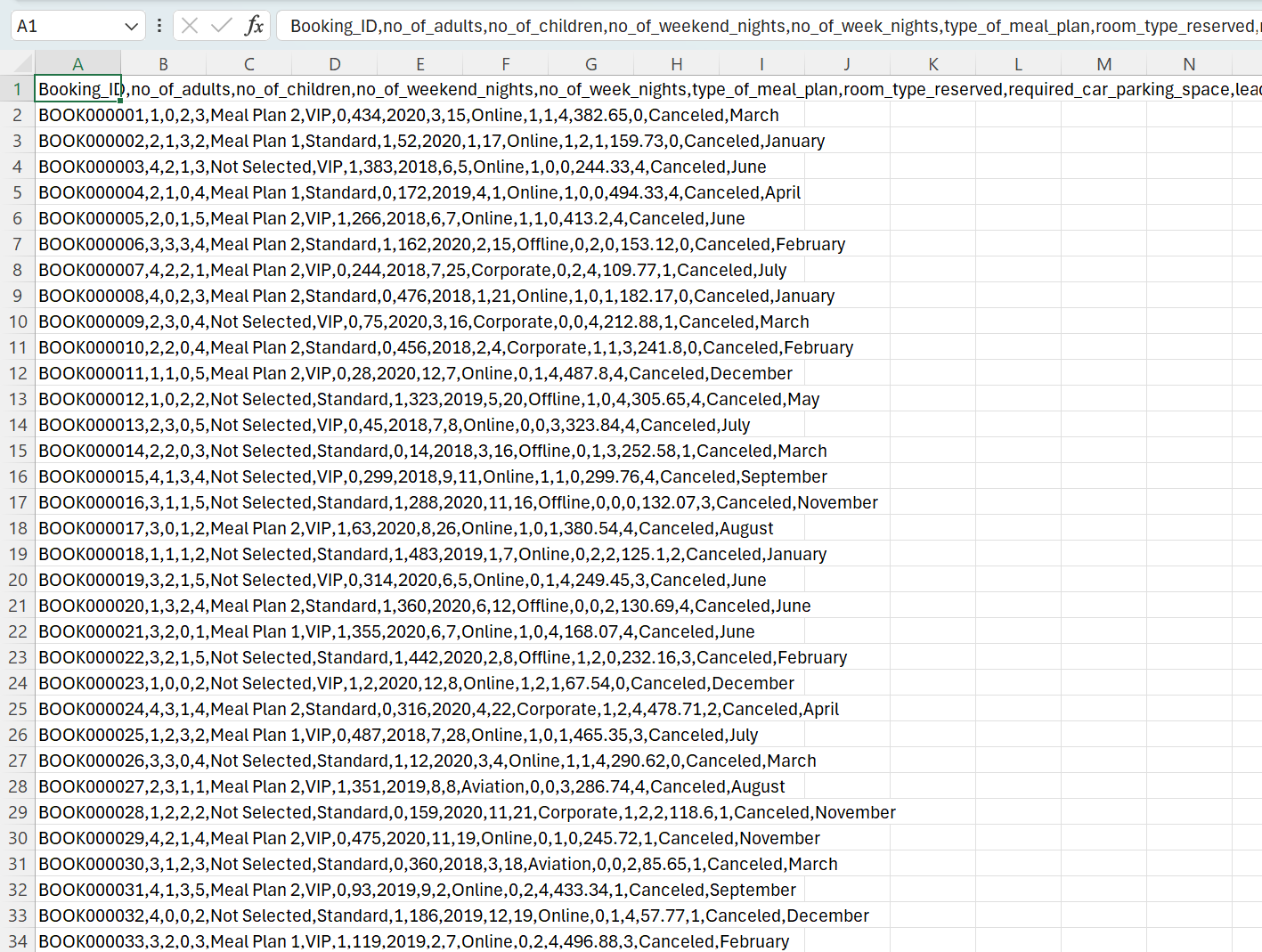
ggplot2 có khả năng trực quan hóa dữ liệu từ cơ bản đến nâng cao với cú pháp trực quan và ngắn gọn. Người dùng có thể tạo biểu đồ dạng biểu đồ cột (bar chart), biểu đồ phân tán (scatter plot), biểu đồ đường (line chart), biểu đồ hộp (boxplot) và nhiều dạng khác chỉ với một vài dòng lệnh. Bên cạnh đó, thư viện này còn hỗ trợ tùy chỉnh màu sắc, kích thước, nhãn trục, tiêu đề và các thành phần khác một cách linh hoạt, giúp biểu đồ trở nên trực quan và dễ hiểu hơn.



Không chỉ hỗ trợ dữ liệu tĩnh, ggplot2 còn có thể kết hợp với các thư viện khác để tạo biểu đồ tương tác, chẳng hạn như plotly hoặc gganimate để tạo hiệu ứng động. Ngoài ra, ggplot2 tích hợp tốt với hệ sinh thái tidyverse, cho phép kết hợp dễ dàng với dplyr, tidyr để làm sạch và xử lý dữ liệu trước khi trực quan hóa.

**CHƯƠNG 2 : XÂY DỰNG CHƯƠNG TRÌNH.**

* 1. **Giới thiệu tập dữ liệu.**



Bộ dữ liệu về đặt phòng khách sạn cung cấp cái nhìn toàn diện về hành vi của khách hàng, từ quá trình đặt phòng đến việc lưu trú và các yêu cầu đặc biệt. Với hơn 500.000 bản ghi, bộ dữ liệu này giúp phân tích xu hướng thuê phòng dựa trên nhiều yếu tố như số lượng khách, số đêm lưu trú, loại phòng, phân khúc thị trường và giá phòng trung bình. Ngoài ra, thông tin về thời gian đặt trước, yêu cầu đỗ xe và lịch sử hủy đặt phòng cũng hỗ trợ trong việc đánh giá mức độ cam kết của khách hàng. Đặc biệt, dữ liệu về trạng thái đặt phòng (đã xác nhận hoặc bị hủy) giúp xác định các yếu tố ảnh hưởng đến tỷ lệ hủy đặt, từ đó tối ưu hóa chiến lược kinh doanh của khách sạn. Việc khai thác bộ dữ liệu này có thể hỗ trợ trong dự báo nhu cầu, cá nhân hóa trải nghiệm khách hàng và cải thiện chiến lược giá nhằm tăng doanh thu và nâng cao chất lượng dịch vụ.

Các thuộc tính chính dùng để sử dụng trong chương trình :

* Booking\_ID: Mã đặt phòng duy nhất cho mỗi giao dịch đặt phòng.
* no\_of\_adults: Số lượng người lớn trong đơn đặt phòng.
* no\_of\_children: Số lượng trẻ em đi cùng trong đơn đặt phòng.
* no\_of\_weekend\_nights: Số đêm khách lưu trú vào cuối tuần (thứ Bảy và Chủ Nhật).
* no\_of\_week\_nights: Số đêm khách lưu trú vào các ngày trong tuần (Thứ Hai - Thứ Sáu).
* type\_of\_meal\_plan: Loại kế hoạch ăn uống được khách hàng chọn (ví dụ: chỉ bữa sáng, nửa ngày, toàn phần hoặc không có).
* room\_type\_reserved: Loại phòng mà khách hàng đã đặt (ví dụ: Standard, Deluxe, Suite).
* required\_car\_parking\_space: Khách hàng có yêu cầu chỗ đậu xe hay không (1 = Có, 0 = Không).
* lead\_time: Số ngày từ khi đặt phòng đến ngày nhận phòng.
* arrival\_year: Năm khách hàng đến nhận phòng.
* arrival\_month: Tháng khách hàng đến nhận phòng (dạng số, từ 1 đến 12).
* arrival\_date: Ngày khách hàng đến nhận phòng trong tháng đó.
* market\_segment\_type: Phân khúc thị trường mà khách hàng thuộc vào (ví dụ: đặt phòng qua công ty du lịch, đặt trực tiếp, đặt qua công ty).
* repeated\_guest: Khách hàng có phải là khách quay lại hay không (1 = Có, 0 = Không).
* no\_of\_previous\_cancellations: Số lần khách hàng đã hủy đặt phòng trước đó.
* no\_of\_previous\_bookings\_not\_canceled: Số lần khách hàng đã đặt phòng trước đó mà không hủy.
* avg\_price\_per\_room: Giá trung bình của phòng (tính theo đơn vị tiền tệ của khách sạn).
* no\_of\_special\_requests: Số yêu cầu đặc biệt mà khách hàng đưa ra (ví dụ: giường đôi, tầng cao, tầm nhìn đẹp, dịch vụ đặc biệt).
* booking\_status: Trạng thái đặt phòng (Confirmed = Đã xác nhận, Canceled = Đã hủy).

- arrival\_date\_month: Tháng khách đến nhận phòng (dạng chữ, ví dụ: "January", "February").

**2.2 Xây dựng code.**

Trước khi bắt đầu, các thư viện cần thiết được tải và nạp vào môi trường R:

**sparklyr**: Hỗ trợ kết nối và tương tác với Apache Spark.

**dplyr**: Giúp xử lý, thao tác dữ liệu.

**ggplot2**: Trực quan hóa dữ liệu bằng biểu đồ.

**caret**: Hỗ trợ đánh giá mô hình máy học.

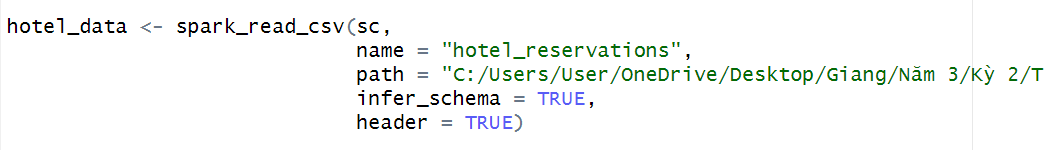
**e1071**: Chứa các hàm hỗ trợ máy học như SVM, Naive Bayes.

**2.2.1 Kết nối với Spark.**

****

Chương trình kết nối với Spark để tận dụng khả năng xử lý dữ liệu lớn, giúp tăng tốc độ và tối ưu bộ nhớ.

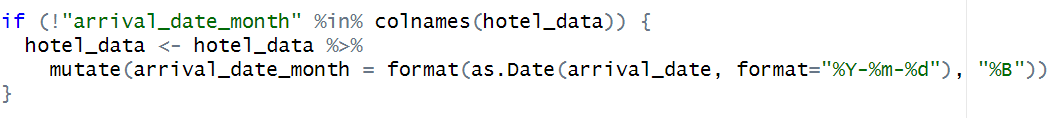
**2.2.2 Đọc dữ liệu từ file CSV.**

****

Bộ dữ liệu khách sạn được lưu trữ trong một file CSV, và Spark sẽ đọc file này và tự động xác định kiểu dữ liệu.

infer\_schema = TRUE: Tự động xác định kiểu dữ liệu của từng cột.

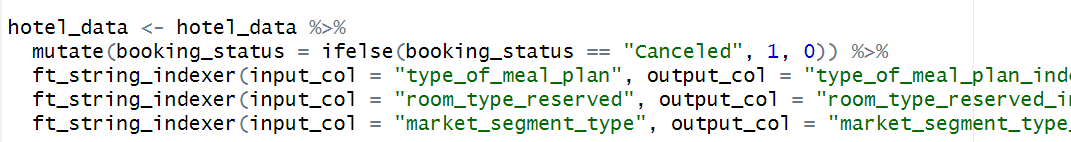
header = TRUE: Xác định dòng đầu tiên là tiêu đề cột.



Kiểm tra xem cột arrival\_date\_month có tồn tại không.

Nếu chưa có, tạo cột mới từ arrival\_date và chuyển sang dạng tháng (January, February, ...).

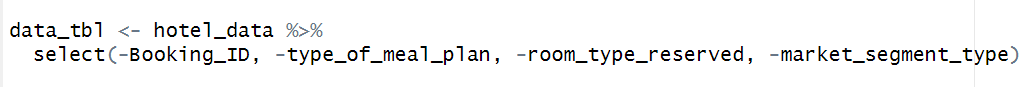
**2.2.3 Tiền xử lý dữ liệu.**



Chuyển đổi cột booking\_status thành giá trị nhị phân (1: hủy, 0: nếu không).

Chuyển đổi các cột chữ thành dữ liệu số sử dụng StringIndexer đổi các cột dạng chuỗi (type\_of\_meal\_plan, room\_type\_reserved, market\_segment\_type) thành số để mô hình có thể xử lý.

Loại bỏ các cột không cần thiết.



Bỏ đi các cột không có giá trị cho việc dự đoán (Booking\_ID, type\_of\_meal\_plan, room\_type\_reserved, market\_segment\_type).

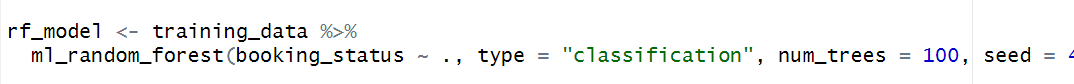
**2.2.4 Chia dữ liệu thành tập huấn luyện và tập kiểm tra.**

****

Dữ liệu được chia tỷ lệ 70-30, trong đó 70% dùng để huấn luyện mô hình, 30% dùng để kiểm tra độ chính xác.

seed = 42: Đảm bảo việc chia dữ liệu là ngẫu nhiên nhưng có thể tái lập.

**2.2.5 Xây dựng mô hình Random Forest.**

****

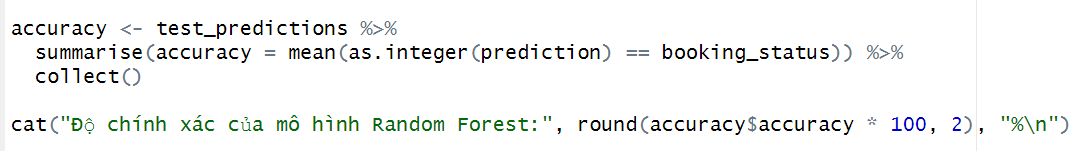
Mô hình Random Forest được huấn luyện với 100 cây quyết định (“num\_trees = 100”) nhằm tối ưu hiệu suất phân loại tỉ lệ hủy phòng và đồng thời đảm bảo kết quả có thể tái lập

**2.2.6 Dự đoán và đánh giá mô hình.**

Mô hình dự đoán kết quả và tính độ chính xác trung bình. Kết quả sẽ hiển thị để đánh giá hiệu quả.



Dự đoán booking\_status trên tập kiểm tra.



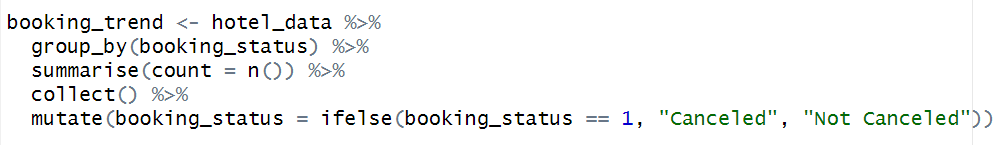
Tính toán độ chính xác: số dự đoán đúng / tổng số dự đoán.

Hiển thị kết quả với cat().

**2.2.7 Tạo biểu đồ tròn.**

**Xử lý dữ liệu ban đầu.**

Chuẩn bị dữ liệu từ tập hotel\_data, sau đó nhóm dữ liệu và đếm số lượt đặt phòng theo trạng thái:



**Bước 1:** group\_by(booking\_status) → Gom nhóm dữ liệu theo trạng thái đặt phòng, tức là chia thành hai nhóm: **đặt phòng bị hủy** (1) và **không hủy** (0).

**Bước 2:** summarise(count = n()) → Đếm số lượng đặt phòng ở mỗi nhóm để biết chính xác có bao nhiêu lượt bị hủy và không bị hủy.

**Bước 3:** collect() → Vì dữ liệu có thể đang xử lý trên Spark (một hệ thống tính toán phân tán), lệnh này giúp lấy dữ liệu về để làm việc trực tiếp trong R.

**Bước 4:** mutate(booking\_status = ifelse(booking\_status == 1, "Canceled", "Not Canceled"))

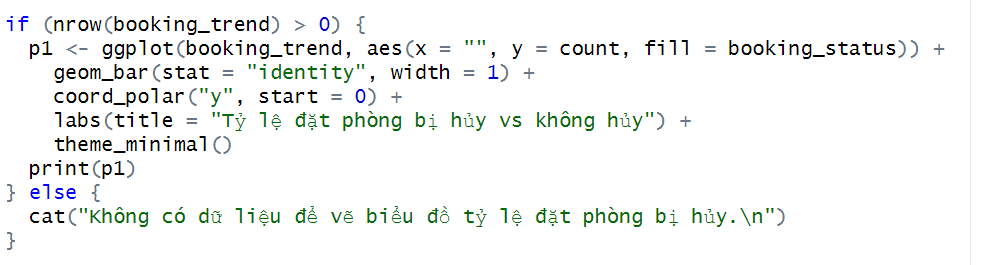
* + Dữ liệu gốc chỉ có số 0 và 1, nhìn khá khó hiểu.
  + Dòng lệnh này giúp chuyển đổi 1 thành "Canceled" (bị hủy) và 0 thành "Not Canceled" (không hủy), giúp biểu đồ hiển thị rõ ràng hơn.

Sau bước này, chúng ta có một bảng dữ liệu mới chứa hai cột:

|  |  |
| --- | --- |
| **booking\_status** | **count** |
| Canceled | 50000 |
| Not Canceled | 450000 |

**Kiểm tra dữ liệu và vẽ biểu đồ**

Sau khi có dữ liệu, ta sẽ kiểm tra xem có dữ liệu không rồi mới vẽ biểu đồ.



**Bước 1: Kiểm tra dữ liệu**

Kiểm tra xem bảng dữ liệu có dữ liệu không (có dòng nào không).

Nếu không có dữ liệu, chương trình sẽ không cố vẽ mà chỉ in ra dòng "Không có dữ liệu để vẽ biểu đồ tỷ lệ đặt phòng bị hủy." để tránh lỗi.

**Bước 2: Xây dựng biểu đồ**

Dùng thư viện ggplot2 để tạo biểu đồ.

aes(x = "", y = count, fill = booking\_status):

x = "": Không cần trục X, vì đây là biểu đồ tròn.

y = count: Dữ liệu theo trục Y là số lượng đặt phòng ở từng nhóm.

fill = booking\_status: Phân biệt màu sắc dựa vào trạng thái đặt phòng.

**geom\_bar(stat = "identity", width = 1)**

Đây là một biểu đồ cột (bar chart).

stat = "identity" nghĩa là vẽ trực tiếp từ số lượng đặt phòng (thay vì tự đếm lại).

width = 1: Đặt chiều rộng cột để dễ dàng chuyển thành hình tròn.

**coord\_polar("y", start = 0)**

Chuyển biểu đồ cột thành biểu đồ hình tròn bằng cách sử dụng hệ tọa độ cực (polar).

"y": Dữ liệu trên trục Y sẽ được trải ra theo dạng vòng tròn.

**labs(title = "Tỷ lệ đặt phòng bị hủy vs không hủy")**

Đặt tiêu đề cho biểu đồ giúp dễ hiểu hơn.

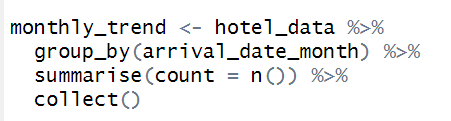
**theme\_minimal()**

Sử dụng một giao diện tối giản, không có quá nhiều chi tiết rườm rà, giúp biểu đồ rõ ràng hơn.

**Bước 3: Hiển thị biểu đồ**

**2.2.8 Tạo biểu đồ cột dọc.**

Lấy dữ liệu từ tập hotel\_data và nhóm theo tháng để đếm số lượt đặt phòng.



**Bước 1:** group\_by(arrival\_date\_month) → Gom nhóm dữ liệu theo tháng khách đến (arrival\_date\_month).

Ví dụ, nếu có các tháng "January", "February", ..., "December", dữ liệu sẽ được nhóm theo các tháng này.

**Bước 2:** summarise(count = n()) → Đếm số lượng đặt phòng cho mỗi tháng.

Chúng ta không quan tâm đến chi tiết từng đặt phòng, chỉ cần biết có bao nhiêu lượt đặt trong mỗi tháng.

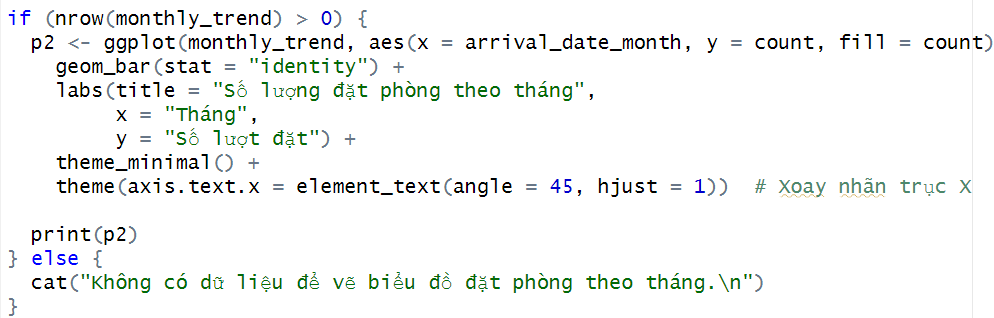
**Bước 3:** collect() → Nếu dữ liệu đang xử lý trên Spark, lệnh này giúp lấy dữ liệu về để làm việc trong R.

Sau bước này, dữ liệu sẽ có dạng:

|  |  |
| --- | --- |
| **arrival\_date\_month** | **count** |
| January | 32000 |
| February | 29000 |
| March | 35000 |

**Kiểm tra dữ liệu và vẽ biểu đồ**

Sau khi có dữ liệu, ta cần kiểm tra xem có dữ liệu không trước khi vẽ biểu đồ.



**Bước 1: Kiểm tra dữ liệu**

Kiểm tra xem bảng dữ liệu có dữ liệu không (có dòng nào không).

Nếu không có dữ liệu (nrow(monthly\_trend) == 0), chương trình sẽ in thông báo:

Điều này giúp tránh lỗi khi cố vẽ biểu đồ với dữ liệu rỗng.

**Bước 2: Xây dựng biểu đồ**

**ggplot(monthly\_trend, aes(x = arrival\_date\_month, y = count, fill = count))**

Sử dụng thư viện ggplot2 để tạo biểu đồ.

aes(x = arrival\_date\_month, y = count, fill = count):

x = arrival\_date\_month: Dữ liệu theo trục X là tên tháng (January, February, ...).

y = count: Dữ liệu theo trục Y là số lượng đặt phòng.

fill = count: Sử dụng số lượng đặt phòng để tô màu cột, giúp trực quan hơn.

**geom\_bar(stat = "identity")**

Đây là một biểu đồ cột (bar chart).

stat = "identity" nghĩa là vẽ trực tiếp từ số lượng đặt phòng (thay vì tự đếm lại).

**labs(title = "Số lượng đặt phòng theo tháng", x = "Tháng", y = "Số lượt đặt")**

Đặt tiêu đề và nhãn cho trục X, Y để người xem dễ hiểu.

**theme\_minimal()**

Sử dụng giao diện tối giản, giúp biểu đồ gọn gàng hơn.

**theme(axis.text.x = element\_text(angle = 45, hjust = 1))**

Xoay nhãn tháng trên trục X một góc 45 độ để tránh bị chồng chéo, giúp dễ đọc hơn.

**Bước 3: Hiển thị biểu đồ**

**2.2.9 Tạo biểu đồ thanh ngang.**

**Xử lý dữ liệu về phân khúc khách hàng**

****

**hotel\_data**: Đây là dataset chứa thông tin về các đặt phòng khách sạn.

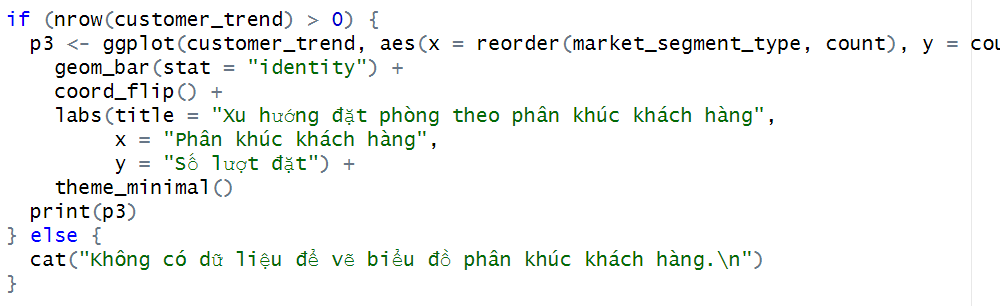
**group\_by(market\_segment\_type)**: Nhóm dữ liệu theo từng phân khúc khách hàng (market\_segment\_type). Điều này có nghĩa là chúng ta gom các đặt phòng theo từng loại phân khúc, ví dụ như "Online", "Corporate", "Direct", v.v.

**summarise(count = n())**: Sau khi nhóm, chúng ta đếm số lượng đặt phòng trong mỗi phân khúc bằng cách dùng n(), kết quả sẽ lưu vào cột count.

**collect()**: Vì dữ liệu đang được xử lý trong Spark, lệnh này giúp lấy dữ liệu về để xử lý tiếp trong R.

Kết quả sau bước này là một dataframe customer\_trend gồm 2 cột:

|  |  |
| --- | --- |
| **market\_segment\_type** | **count** |
| Online | 15000 |
| Corporate | 12000 |
| Direct | 18000 |



Tương tự như các biểu đồ khác, ta kiểm tra xem dataframe customer\_trend có dữ liệu hay không (nrow(customer\_trend) > 0).

Nếu có dữ liệu, chương trình tiếp tục vẽ biểu đồ, còn nếu không, nó sẽ hiển thị thông báo "Không có dữ liệu để vẽ biểu đồ phân khúc khách hàng."

**ggplot(customer\_trend, aes(...))**: Tạo biểu đồ với dữ liệu customer\_trend, xác định trục x, y, và màu sắc.

**aes(x = reorder(market\_segment\_type, count), y = count, fill = market\_segment\_type)**

Trục **x**: Sử dụng reorder(market\_segment\_type, count) để sắp xếp phân khúc khách hàng theo số lượt đặt từ thấp đến cao, giúp biểu đồ dễ đọc hơn.

Trục **y**: Số lượng đặt phòng (count).

**fill = market\_segment\_type**: Mỗi phân khúc khách hàng sẽ có một màu sắc riêng để dễ phân biệt.

**geom\_bar(stat = "identity")**: Dùng biểu đồ cột (bar chart) với chiều cao của cột đúng bằng giá trị count.

**coord\_flip()**: Xoay biểu đồ ngang để dễ quan sát. Nếu không có dòng này, biểu đồ sẽ thẳng đứng.

**labs(...)**: Thêm tiêu đề và nhãn cho trục.

**theme\_minimal()**: Áp dụng giao diện đơn giản, giúp biểu đồ dễ nhìn hơn.

**CHƯƠNG 3 : KẾT QUẢ CHƯƠNG TRÌNH**

* 1. **Biểu đồ tròn.**



**3.1.1 Giải thích.**

**Màu đỏ (Canceled)**: Đại diện cho các đơn đặt phòng đã bị hủy.

**Màu xanh (Not Canceled)**: Đại diện cho các đơn đặt phòng vẫn giữ nguyên.

**Tỷ lệ trên biểu đồ**:

Màu đỏ chiếm phần lớn biểu đồ → Số lượng đặt phòng bị hủy cao hơn đáng kể so với số lượng đặt phòng không bị hủy.

Màu xanh chiếm phần nhỏ hơn → Tỷ lệ đặt phòng thành công thấp hơn.

* + 1. **Phân tích kết quả.**

**Xu hướng hủy phòng cao**:

Từ biểu đồ, có thể thấy tỷ lệ Canceled (bị hủy) chiếm khoảng 60-70% tổng số đặt phòng, trong khi Not Canceled (không hủy) chỉ chiếm khoảng 30-40%.

Điều này cho thấy có rất nhiều đơn đặt phòng không được thực hiện thành công, có thể do nhiều lý do như khách đổi kế hoạch, giá cả thay đổi, hoặc các yếu tố khác.

**Tác động kinh doanh**:

Ảnh hưởng đến doanh thu khách sạn: Khi nhiều khách đặt phòng rồi hủy, khách sạn sẽ gặp khó khăn trong việc đảm bảo công suất phòng tối đa.

Cần có chính sách hủy phòng hợp lý: Nếu tỷ lệ hủy quá cao, khách sạn có thể cần áp dụng các chính sách như đặt cọc trước, giảm giá cho các đơn đặt phòng không hoàn tiền, hoặc có các ưu đãi cho khách hàng trung thành để giảm tỷ lệ hủy.

**3.2 Biểu đồ cột dọc.**



**3.2.1 Giải thích.**

Trục X (Tháng): Hiển thị tên các tháng trong năm (April, August, December, v.v.), được xoay chéo để dễ đọc.

Trục Y (Số lượt đặt phòng): Hiển thị tổng số lượt đặt phòng cho từng tháng, với giá trị dao động khoảng 41.400 - 41.900.

Màu sắc thanh cột :

Sắc độ màu đậm nhạt khác nhau phản ánh số lượt đặt phòng.

Màu tối hơn (June, July) → Số lượng đặt phòng thấp hơn một chút so với các tháng còn lại.

Màu sáng hơn (August, December, March, v.v.) → Số lượt đặt phòng nhỉnh hơn.

**3.2.2 Phân tích kết quả.**

**Số lượng đặt phòng khá đồng đều trong năm**:

Không có sự khác biệt lớn giữa các tháng, cho thấy khách sạn có lượng khách ổn định quanh năm.

Tháng June và July có số đặt phòng thấp nhất (màu tối hơn), có thể do đây không phải là mùa du lịch cao điểm.

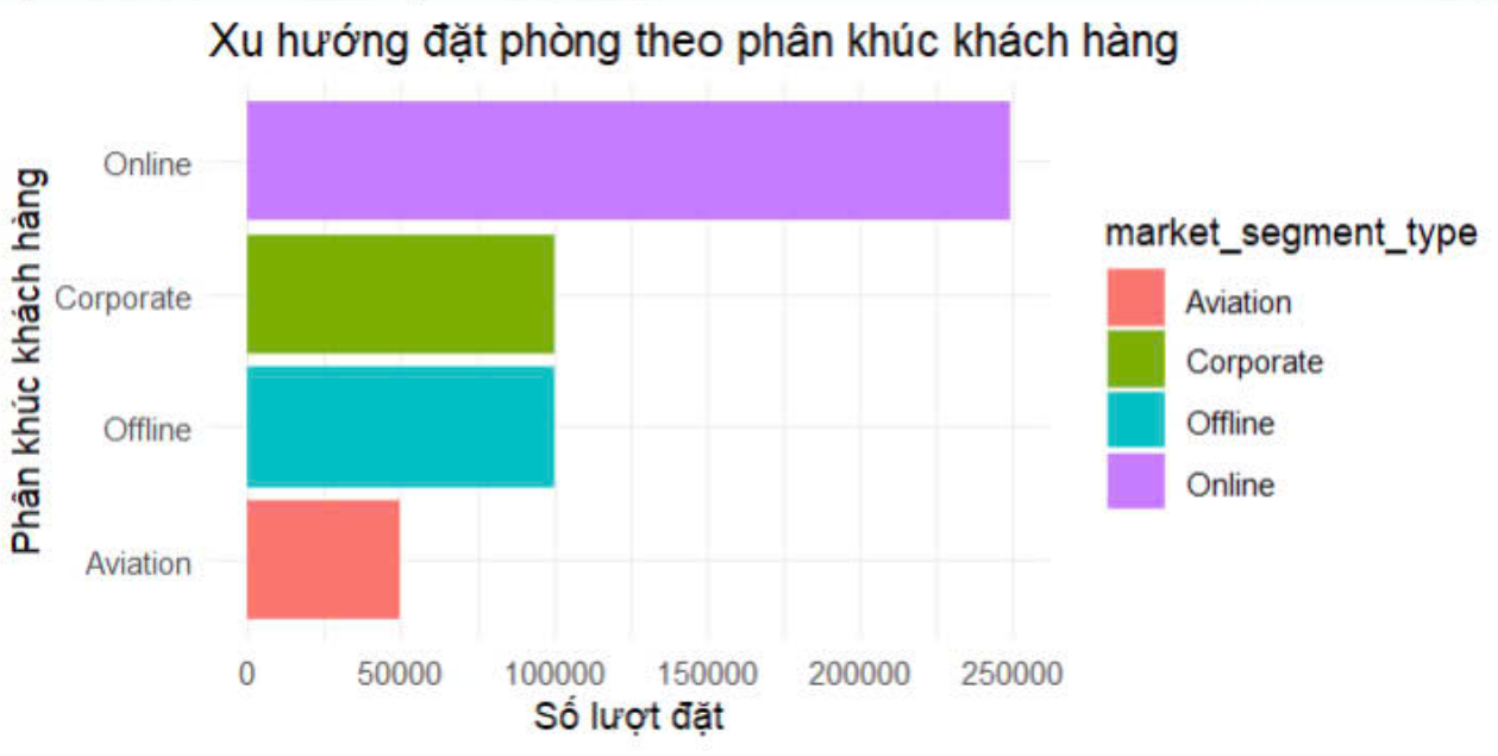
Các tháng như August, December, March có số đặt phòng cao hơn một chút, có thể do rơi vào kỳ nghỉ lễ hoặc mùa cao điểm du lịch.

**Tác động kinh doanh**:

**Khách sạn có thể duy trì hoạt động ổn định** mà không gặp tình trạng mùa vắng khách quá rõ ràng.

**Tập trung vào chiến lược giá**: Vì lượng đặt phòng không thay đổi nhiều, khách sạn có thể tập trung vào việc tối ưu giá phòng, chương trình khuyến mãi để đảm bảo lợi nhuận tốt nhất.

**3.3 Biểu đồ thanh ngang.**



**3.3.1 Giải thích.**

Trục X (Số lượt đặt phòng): Hiển thị tổng số lượt đặt phòng của từng phân khúc khách hàng, giá trị dao động từ khoảng dưới 50.000 đến hơn 250.000.

Trục Y (Phân khúc khách hàng): Bao gồm các loại khách hàng:

Online (Màu tím)

Corporate (Doanh nghiệp) (Màu xanh lá)

Offline (Màu xanh dương)

Aviation (Hàng không) (Màu đỏ)

Màu sắc phân khúc khách hàng :

Giúp phân biệt các nhóm khách hàng khác nhau.

Online (màu tím) chiếm ưu thế với số lượt đặt phòng cao nhất.

Corporate (màu xanh lá) và Offline (màu xanh dương) có số lượt đặt tương đương.

Aviation (màu đỏ) có số lượt đặt thấp nhất.

**3.3.2 Phân tích kết quả.**

**Phân khúc khách hàng Online dẫn đầu với số lượt đặt phòng cao nhất** (~250.000 lượt đặt).

Điều này cho thấy xu hướng đặt phòng qua các nền tảng trực tuyến (website, ứng dụng) ngày càng phổ biến.

Khách sạn có thể tập trung vào các chiến dịch quảng cáo trực tuyến, tối ưu website, hợp tác với các nền tảng đặt phòng như Agoda, Booking.com, Traveloka, v.v.

**Phân khúc Corporate (Doanh nghiệp) và Offline có số lượt đặt phòng tương đương (~100.000 lượt).**

Corporate có thể bao gồm các công ty đặt phòng cho nhân viên đi công tác.

Offline có thể là các khách đặt trực tiếp tại khách sạn hoặc qua đại lý du lịch.

Dù ít hơn Online, nhưng hai nhóm này vẫn chiếm thị phần đáng kể, có thể tận dụng chương trình ưu đãi dành cho khách hàng doanh nghiệp hoặc đại lý du lịch.

**Phân khúc Aviation có số lượt đặt thấp nhất (~50.000 lượt).**

Có thể đây là nhóm khách liên quan đến hàng không, ví dụ như phi hành đoàn, nhân viên sân bay.

Số lượng khách này thấp do họ thường có hợp đồng cố định với một số khách sạn cụ thể.

Khách sạn có thể tìm kiếm cơ hội hợp tác với các hãng hàng không để thu hút thêm lượng khách này.

**KẾT LUẬN**

Từ các phân tích trên, có thể thấy rằng xu hướng thuê phòng khách sạn hiện nay chịu ảnh hưởng lớn từ công nghệ, với kênh đặt phòng trực tuyến là lựa chọn phổ biến nhất. Các khách sạn cần tập trung đầu tư vào nền tảng đặt phòng trực tuyến, tối ưu hóa trải nghiệm người dùng và tăng cường chiến lược tiếp thị số để thu hút khách hàng. Ngoài ra, tỷ lệ hủy phòng cao cho thấy cần có các biện pháp kiểm soát như yêu cầu đặt cọc, chính sách hủy phòng linh hoạt hoặc ưu đãi dành cho những khách đặt sớm để giảm tỷ lệ hủy. Việc phân tích theo phân khúc khách hàng cũng giúp các khách sạn xác định đối tượng tiềm năng để đưa ra chiến lược phù hợp, chẳng hạn như cung cấp chương trình ưu đãi dành riêng cho khách doanh nghiệp hoặc hợp tác với các công ty du lịch để mở rộng nguồn khách hàng.

Hơn nữa, việc ứng dụng mô hình Random Forest trong phân tích giúp dự đoán hành vi đặt phòng và nhận diện các yếu tố quan trọng nhất ảnh hưởng đến quyết định của khách hàng. Điều này giúp các khách sạn tối ưu hóa giá cả, phân bổ nguồn lực hợp lý và nâng cao hiệu quả kinh doanh. Với dữ liệu lớn, các khách sạn có thể sử dụng các mô hình phân tích hiện đại để đưa ra dự báo chính xác hơn về xu hướng đặt phòng trong tương lai, từ đó tối ưu hóa công suất phòng và nâng cao trải nghiệm khách hàng.

**DANH MỤC TÀI LIỆU THAM KHẢO**

Dưới đây là một số tài liệu tham khảo hữu ích về lĩnh vực Dữ liệu lớn mà bạn có thể tham khảo :

1. BIG DATA - CÔNG NGHỆ CỐT LÕI TRONG KỶ NGUYÊN SỐ - Thomas Davenport.
2. BIG DATA CHO NHÀ QUẢN LÝ - Tim Berners-Lee- người sáng lập ra Word While Web.
3. MARKETING THEO DỮ LIỆU LỚN - LISA ARTHUR.