TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN

HUỲNH HOÀNG TIẾN - 51704111

DỰ ĐOÁN GIÁ VÉ CHUYẾN BAY

BÁO CÁO CUỐI KỲ
NHẬP MÔN HỌC MÁY

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN

HUỲNH HOÀNG TIẾN - 51704111

DỰ ĐOÁN GIÁ VÉ CHUYẾN BAY

BÁO CÁO GIỮA KỲ
NHẬP MÔN HỌC MÁY

Người hướng dẫn
PGS.TS. Lê Anh Cường

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023

LỜI CẢM ƠN

Chúng tôi muốn bày tỏ lòng biết ơn sâu sắc đến **PGS.TS. Lê Anh Cường**, Khoa Công Nghệ Thông Tin, và Trường Đại Học Tôn Đức Thắng vì sự hỗ trợ và đóng góp quan trọng trong quá trình thực hiện báo cáo này. Thầy Cường không chỉ là nguồn động viên lớn, mà còn là người hướng dẫn chi tiết, giúp chúng tôi nắm vững kiến thức của môn "Nhập môn Học máy".

Khoa Công Nghệ Thông Tin mang lại môi trường học tập năng động và sáng tạo, nơi chúng tôi có cơ hội tiếp xúc với những kiến thức và công nghệ mới nhất. Đặc biệt, chúng tôi muốn bày tỏ lòng biết ơn với Trường Đại Học Tôn Đức Thắng vì môi trường học thuận lợi, giúp chúng tôi không ngừng khám phá và phát triển.

Cuối cùng, chúng tôi chân thành cảm ơn tất cả những người đã đóng góp vào hành trình học tập và nghiên cứu của chúng tôi. Sự giúp đỡ và khuyến khích của mọi người đã là động lực quan trọng, giúp chúng tôi vượt qua những thách thức và đạt được những thành công nhỏ trên con đường này. Xin chân thành cảm ơn!

TP. Hồ Chí Minh, ngày ... tháng ... năm 20..
Tác giả
(Ký tên và ghi rõ họ tên)

Huỳnh Hoàng Tiến

CÔNG TRÌNH ĐƯỢC HOÀN THÀNH
TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG

Tôi xin cam đoan đây là công trình nghiên cứu của riêng tôi và được sự hướng dẫn khoa học của **PGS.TS. Lê Anh Cường**. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong Dự án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung Dự án của mình. Trường Đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

TP. Hồ Chí Minh, ngày ... tháng ... năm 20..

Tác giả

(Ký tên và ghi rõ họ tên)

Huỳnh Hoàng Tiến

TÓM TẮT

Dự án "Dự đoán giá vé chuyến bay" nhằm xây dựng một hệ thống thông minh sử dụng mô hình học máy để dự đoán giá vé chuyến bay với độ chính xác cao. Sự kết hợp giữa các mô hình như RandomForestRegressor, ExtraTreesRegressor, DecisionTreeRegressor và LinearRegression, cùng với việc thu thập dữ liệu từ trang web "Ease My Trip," tạo ra một công cụ mạnh mẽ giúp người dùng và doanh nghiệp hàng không hiểu rõ và quản lý giá vé một cách hiệu quả. Đồng thời, việc phát triển giao diện người dùng thân thiện giúp cung cấp trải nghiệm dự đoán giá vé trực quan và dễ sử dụng. Với lợi ích đối với cả người dùng và doanh nghiệp, dự án hứa hẹn mang lại giá trị và tiện ích lớn trong việc đặt vé và quản lý giá cả trong ngành hàng không.

ABSTRACT

The project "Flight Price Prediction" aims to develop an intelligent system using machine learning models to predict flight ticket prices with high accuracy. The combination of models such as RandomForestRegressor, ExtraTreesRegressor, DecisionTreeRegressor, and LinearRegression, coupled with data collection from the "Ease My Trip" website, creates a powerful tool that assists both users and airline businesses in understanding and effectively managing ticket prices. Additionally, the development of a user-friendly interface enhances the experience of predicting ticket prices, making it visual and easy to use. With benefits for both users and businesses, the project promises significant value and utility in the processes of ticket booking and price management in the aviation industry.

MỤC LỤC

DANH MỤC HÌNH VẼ

CHƯƠNG 1. ĐẶC TẢ DỰ ÁN

1.1 Tên đề tài

"Dự Đoán Giá Vé Chuyến Bay".

1.2 Phạm vị dự án

Dự án sẽ tập trung vào việc dự đoán giá vé chuyến bay dựa trên các yếu tố như thời gian, địa điểm, hãng hàng không, và các biến số liên quan khác.

Sử dụng dữ liệu lịch sử giá vé để đào tạo và kiểm thử mô hình dự đoán.

1.3 Lí do chọn đề tài

Ứng dụng thực tế: Dự đoán giá vé chuyến bay là một vấn đề thực tế và gặp phổ biến trong ngành hàng không. Giải quyết vấn đề này có thể mang lại giá trị lớn cho hãng hàng không, đối tác thương mại và người tiêu dùng.

Sự phức tạp của dữ liệu: Dữ liệu về giá vé chuyến bay thường đa dạng và phức tạp, với nhiều yếu tố ảnh hưởng. Nắm bắt được sự phức tạp này đòi hỏi sử dụng mô hình học máy mạnh mẽ và phương pháp phân tích sâu rộng.

Tiềm năng tối ưu hóa giá trị: Việc dự đoán chính xác giá vé có thể giúp hãng hàng không tối ưu hóa doanh thu bằng cách định giá chính xác dựa trên nhu cầu thực tế và các yếu tố biến động.

Thách thức công nghệ: Đề tài đặt ra những thách thức về mặt công nghệ, đặc biệt là trong việc tích hợp các mô hình học máy phức tạp vào hệ thống thương mại thông minh.

Liên kết với hệ thống thương mại thông minh: Dự án có thể được tích hợp vào hệ thống thương mại thông minh, tạo ra một giải pháp toàn diện và hiệu quả cho các doanh nghiệp.

Đào tạo mô hình học máy: Dự án sẽ cung cấp trải nghiệm thực tế trong việc thu thập, tiền xử lý dữ liệu và đào tạo mô hình học máy, mang lại cơ hội học hỏi lớn.

Ước tính chính xác: Việc đưa ra dự đoán chính xác về giá vé chuyến bay có thể giúp người tiêu dùng và đối tác thương mại đưa ra quyết định thông tin và hiệu quả.

Tích hợp các yếu tố ảnh hưởng: Dự án sẽ thử nghiệm khả năng tích hợp các yếu tố như thời gian, địa điểm, sự kiện đặc biệt và các yếu tố khác để tối ưu hóa dự đoán giá vé.

Tính ứng dụng quốc tế: Phương pháp và mô hình phát triển có thể được áp dụng cho nhiều thị trường quốc tế, mở rộng phạm vi và tầm ảnh hưởng của dự án.

1.4 Công nghệ sử dụng

Ngôn ngữ lập trình: Python là ngôn ngữ lập trình chủ đạo trong lĩnh vực học máy và phân tích dữ liệu. Nó được chọn vì cộng đồng lớn, nhiều thư viện mạnh mẽ như NumPy, pandas, và scikit-learn, cũng như tính linh hoạt và dễ đọc.

Mô hình học máy: Sử dụng các mô hình học máy như Random Forest Regressor, Extra Trees Regressor, Decision Tree Regressor, Linear Regression. Các mô hình này cung cấp sự đa dạng trong cách chúng học và đưa ra dự đoán, giúp tối ưu hóa hiệu suất của hệ thống trong việc dự đoán giá vé chuyến bay. Sự kết hợp của chúng có thể giúp giảm nguy cơ quá mức đào tạo và cung cấp sự linh hoạt trong việc xử lý đa dạng của dữ liệu.

1.5 Phương pháp đánh giá mô hình

Phương pháp đánh giá mô hình sẽ sử dụng các độ đo như MAE (Mean Absolute Error), Mean Squared Error (MSE), RMSE (Root Mean Squared Error), R-

squared, RMSLE (Root Mean Squared Logarithmic Error), MAPE (Mean Absolute Percentage Error) để cung cấp cái nhìn toàn diện về hiệu suất của mô hình đối với dự đoán giá vé các chuyến bay. Điều này giúp đảm bảo rằng mô hình không chỉ chính xác mà còn đáp ứng đầy đủ các yếu tố quan trọng trong bài toán.

1.5.1 MAE (Mean Absolute Error)

Đo lường sự chênh lệch trung bình giữa giá trị dự đoán và giá trị thực tế. Giá trị càng thấp càng tốt, vì nó chỉ tính giá trị tuyệt đối của sai số mà không quan tâm đến hướng.

1.5.2 MSE (Mean Squared Error)

Đo lường sự chênh lệch bình phương trung bình giữa giá trị dự đoán và giá trị thực tế. Nó có xu hướng đặt trọng số lớn hơn cho các sai số lớn. Giá trị MSE cao thường cho thấy mô hình đang mắc phải các dự đoán lớn.

1.5.3 RMSE (Root Mean Squared Error)

Là căn bậc hai của MSE, giúp đưa ra đánh giá về sự chênh lệch trung bình giữa giá trị dự đoán và giá trị thực tế. RMSE cũng giúp mô hình tránh được ảnh hưởng của các sai số lớn.

1.5.4 R-squared (R2)

Đo độ giải thích của mô hình đối với phương sai của dữ liệu. Giá trị R2 gần 1 cho thấy mô hình giải thích được một phần lớn sự biến động của dữ liệu.

1.5.5 RMSLE (Root Mean Squared Logarithmic Error)

Là biến thể của RMSE được áp dụng trên logarit của giá trị dự đoán và giá trị thực tế. Thường được sử dụng khi dự đoán dữ liệu có phạm vi rộng và có ý nghĩa khi các sai số nhỏ quan trọng hơn các sai số lớn.

1.5.6 MAPE (Mean Absolute Percentage Error)

Đo lường tỷ lệ phần trăm trung bình giữa giá trị tuyệt đối của sự chênh lệch và giá trị thực tế. Giá trị càng thấp cho thấy mức độ chính xác của mô hình, nhưng nó cũng có thể bị ảnh hưởng bởi giá trị thực tế gần

1.5.7 Đánh giá trực quan

Sử dụng biểu đồ và đồ thị để hiểu rõ hơn về hiệu suất của mô hình trên các điểm dữ liệu cụ thể và trong các phân khúc khác nhau của dữ liệu.

1.6 Bước tiến thực hiện dự án

Thực hiện dự án dự đoán giá vé chuyến bay có thể được chia thành nhiều bước để giảm thiểu rủi ro và tối ưu hóa hiệu suất. Mỗi bước cần được thực hiện một cách cẩn thận và theo dõi để đảm bảo rằng mô hình không chỉ hiệu quả trong môi trường thử nghiệm mà còn trong điều kiện thực tế.

Dưới đây là một kịch bản tổng quan về 6 bước quan trọng:

1.6.1 Thu thập dữ liệu

Xác định nguồn dữ liệu: Xác định các nguồn dữ liệu quan trọng như các trang web chuyến bay, API hoặc cơ sở dữ liệu đã có.

Thu thập dữ liệu: Sử dụng các công cụ như web scraping, API calls hoặc mô đun thu thập dữ liệu để lấy thông tin về các chuyến bay, giá vé, và các yếu tố khác có thể ảnh hưởng đến giá.

1.6.2 Tiền xử lý dữ liệu

Kiểm tra và xử lý dữ liệu thiếu: Thực hiện kiểm tra để xác định và xử lý dữ liệu thiếu, bằng cách điền giá trị trung bình, sử dụng kỹ thuật khác nhau như Interpolation, hoặc loại bỏ các mẫu dữ liệu không đầy đủ.

Chuyển đổi dữ liệu: Mã hóa các biến phân loại, chuẩn hóa dữ liệu số để chuẩn bị cho quá trình huấn luyện mô hình.

Tạo tập dữ liệu huấn luyện và kiểm tra: Phân chia dữ liệu thành tập huấn luyện và tập kiểm tra để đảm bảo khả năng đánh giá hiệu suất của mô hình trên dữ liệu mới.

1.6.3 Xây dựng mô hình

Lựa chọn mô hình: Chọn các mô hình học máy phù hợp cho bài toán dự đoán giá vé chuyến bay, có thể thử nghiệm và so sánh hiệu suất giữa chúng.

Huấn luyện mô hình: Sử dụng tập huấn luyện để huấn luyện mô hình với các tham số tối ưu. Thực hiện các bước kiểm soát quá trình để tránh tình trạng quá mức đào tạo.

1.6.4 Đánh giá mô hình

Sử dụng độ đo hiệu suất: Sử dụng độ đo như MAE, MSE, RMSE, R-squared, RMSLE, MAPE để đánh giá hiệu suất của mô hình trên tập kiểm tra.

1.6.5 Điều chỉnh mô hình

Dựa vào kết quả đánh giá, điều chỉnh các tham số của mô hình nếu cần thiết để cải thiện hiệu suất.

1.6.6 Triển khai và tối ưu hóa

Triển khai mô hình: Triển khai mô hình vào môi trường dịch vụ để có thể sử dụng trong thực tế.

Tối ưu hóa hiệu suất: Theo dõi và tối ưu hóa hiệu suất của mô hình trong môi trường dịch vụ, thông qua việc đảm bảo cập nhật định kỳ mô hình với dữ liệu mới.

1.6.7 Quản lý và bảo trì

Quản lý mô hình: Thiết lập quy trình quản lý mô hình để theo dõi và duy trì mô hình trong thời gian.

1.6.8 Bảo trì và cập nhật

Bảo trì mô hình bằng cách kiểm tra và cập nhật nó khi có sự thay đổi về dữ liệu, yêu cầu kinh doanh hoặc thay đổi về môi trường.

1.7 Lợi ích

1.7.1 Lợi ích chính của dự án

Tối ưu hóa giá vé: Dự án giúp hãng hàng không tối ưu hóa giá vé chuyến bay, điều này có thể dẫn đến việc tăng cường cạnh tranh và thu hút hành khách.

Tiết kiệm chi phí: Bằng cách dự đoán giá vé chính xác, hãng hàng không có thể giảm thiểu rủi ro và tối ưu hóa doanh thu, dẫn đến việc tiết kiệm chi phí.

Trải nghiệm hành khách cải thiện: Hành khách sẽ hưởng lợi từ việc có giá vé dự đoán chính xác và ổn định, tạo ra trải nghiệm du lịch thuận lợi và dễ dàng hơn.

1.7.2 Lợi ích xã hội và kinh tế

Phát triển ngành hàng không: Dự án có thể đóng góp vào sự phát triển bền vững của ngành hàng không bằng cách tối ưu hóa quy trình kinh doanh và cung cấp giá vé hợp lý.

Tăng cường cạnh tranh: Hãng hàng không có khả năng cung cấp giá vé cạnh tranh hơn, tạo ra sức hút lớn đối với hành khách và đồng thời thúc đẩy sự cạnh tranh trong ngành.

Tạo việc làm và tăng thu nhập: Tính hiệu quả của ngành hàng không có thể dẫn đến tăng cường cơ hội việc làm và tăng thu nhập cho các đối tác liên quan.

1.7.3 Tầm nhìn tương lai

Mở rộng ứng dụng: Dự án có thể mở rộng để dự đoán giá vé cho các loại dịch vụ khác trong ngành du lịch và giao thông vận tải.

Tích hợp dữ liệu thêm: Nâng cao mô hình bằng cách tích hợp dữ liệu mới, chẳng hạn như thời tiết, sự kiện đặc biệt, để dự đoán giá vé chính xác hơn.

Nâng cao hiệu suất mô hình: Tiếp tục nghiên cứu và phát triển để nâng cao hiệu suất của mô hình dự đoán giá vé và đáp ứng linh hoạt với biến động thị trường.

1.7.4 Tiềm năng thương mại

Cung cấp dịch vụ tư vấn: Dự án có thể mở rộng để cung cấp dịch vụ tư vấn về giá vé cho các doanh nghiệp du lịch và đối tác thương mại khác.

CHƯƠNG 2. MÔ TẢ VỀ TẬP DỮ LIỆU

2.1 Giới thiệu

Mục tiêu của nghiên cứu này là phân tích bộ dữ liệu đặt vé máy bay thu thập từ trang web "Ease My Trip" và thực hiện các kiểm định giả thuyết thống kê khác nhau để có được thông tin có ý nghĩa từ nó. "Easemytrip" là một nền tảng trực tuyến để đặt vé máy bay, là một công cụ quan trọng mà hành khách tiềm năng sử dụng để mua vé.

2.2 Câu hỏi nghiên cứu

Mục đích nghiên cứu của chúng tôi là trả lời các câu hỏi nghiên cứu dưới đây:

- Giá có thay đổi tùy theo hãng hàng không hay không?
- Giá vé bị ảnh hưởng như thế nào khi mua vé chỉ 1 hoặc 2 ngày trước ngày khởi hành?
- Giá vé có thay đổi theo thời gian đi và đến không?
- Giá thay đổi như thế nào khi thay đổi điểm đi và điểm đến?
- Giá vé khác nhau như thế nào giữa hạng phổ thông và hạng thương gia?

2.3 Thu thập dữ liệu và phương pháp

Công cụ Octoparse đã được sử dụng để trích xuất dữ liệu từ trang web. Dữ liệu được thu thập thành hai phần: một cho vé hạng phổ thông và một cho vé hạng doanh. Tổng cộng có *300,261 lựa chọn đặt vé máy bay* được trích xuất từ trang web *trong khoảng 50 ngày, từ ngày 11 tháng 2 đến 31 tháng 3 năm 2022*. Nguồn dữ liệu là dữ liệu phụ thuộc thu thập từ trang web Ease My Trip.

2.4 Tập dữ liệu

Tập dữ liệu chứa thông tin về các tùy chọn đặt vé máy bay từ trang web Easemytrip dành cho chuyến bay giữa *6 thành phố đô thị hàng đầu của Ấn Độ*. Có *300261 điểm dữ liệu và 11 đặc trưng* trong tập dữ liệu đã được làm sạch.

	airline	flight	source_city	departure_time	stops	arrival_time	destination_city	class	duration	days_left	price
0	SpiceJet	SG-8709	Delhi	Evening	zero	Night	Mumbai	Economy	2.17	1	5953
1	SpiceJet	SG-8157	Delhi	Early_Morning	zero	Morning	Mumbai	Economy	2.33	1	5953
2	AirAsia	I5-764	Delhi	Early_Morning	zero	Early_Morning	Mumbai	Economy	2.17	1	5956
3	Vistara	UK-995	Delhi	Morning	zero	Afternoon	Mumbai	Economy	2.25	1	5955
4	Vistara	UK-963	Delhi	Morning	zero	Morning	Mumbai	Economy	2.33	1	5955

2.5 Đặc trưng

Hãng hàng không (airline): Đặc trưng phân loại lưu trữ tên của công ty hàng không với 6 hãng khác nhau.

Mã chuyến bay (flight): Đặc trưng phân loại lưu trữ thông tin về mã chuyến bay của máy bay.

Điểm đi (source_city): Đặc trưng phân loại lưu trữ thành phố từ đó chuyến bay cất cánh, với 6 thành phố duy nhất (Bangalore, Chennai, Delhi, Hyderabad, Kolkata, Mumbai).

Thời gian khởi hành (departure_time): Đây là đặc trưng phân loại được tạo ra bằng cách nhóm các khoảng thời gian thành các bin, lưu trữ thông tin về thời gian khởi hành và có 6 nhãn thời gian duy nhất (Early_Morning, Morning, Afternoon, Evening, Night, Late_Night).

Số lần dừng (stops): Đặc trưng phân loại với 3 giá trị phân biệt lưu trữ số lần dừng giữa các thành phố nguồn và đích.

Thời gian đến nơi (arrival_time): Đây là đặc trưng phân loại được tạo ra bằng cách nhóm các khoảng thời gian thành các bin, lưu trữ thông tin về thời gian đến nơi với 6 nhãn thời gian duy nhất (Early_Morning, Morning, Afternoon, Evening, Night, Late_Night).

Điểm đến (destination_city): Đặc trưng phân loại lưu trữ thành phố nơi chuyến bay sẽ hạ cánh, với 6 thành phố duy nhất (Bangalore, Chennai, Delhi, Hyderabad, Kolkata, Mumbai).

Hạng vé (class): Đặc trưng phân loại chứa thông tin về hạng ghế; có hai giá trị phân biệt là “bussiness” và “economy”.

Thời lượng chuyến bay (duration): Đặc trưng liên tục hiển thị tổng thời gian di chuyển giữa các thành phố.

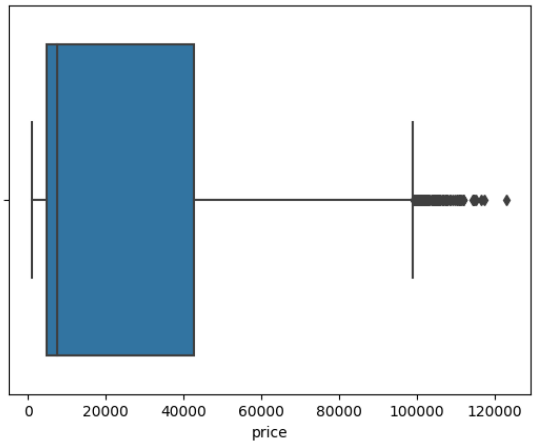
Ngày còn lại (day_left): Đây là đặc điểm phân loại được tính bằng cách trừ ngày khởi hành và ngày đặt vé.

Giá (price): Biến mục tiêu chứa thông tin về giá vé.

2.6 Trực quan hoá dữ liệu

- Có sự giá chênh lệch ở tất cả các hãng hàng không và hạng ghế từ 100.000 đến 12000.

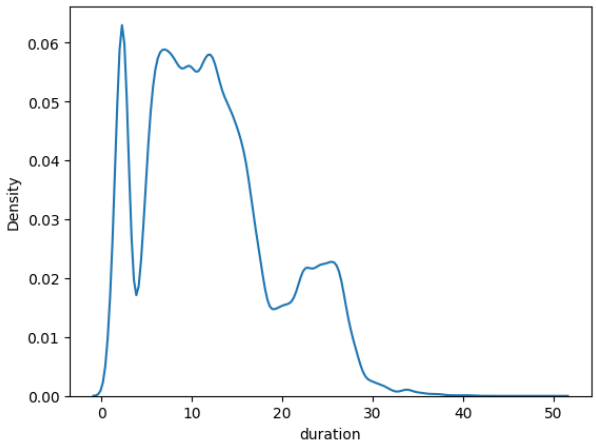
```
sns.boxplot(x=df["price"])
```



Hình 1. Chênh lệch ở tất cả các hãng hàng không

- Thời lượng bị lệch phải và có đỉnh cao trong khoảng từ 0 đến 5, nghĩa là các chuyến bay kéo dài từ 0 đến 5 giờ là chuyến bay thường xuyên nhất.

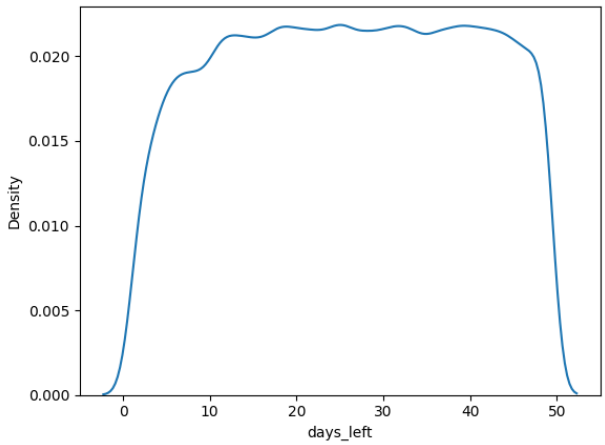
```
sns.kdeplot(data= df, x="duration")
```



Hình 2. Thời lượng chuyến bay bị lệch phải

- Days_left thường được phân phối với mức cao nhất từ 10 đến 40, cho thấy các chuyến bay hầu hết được bán khi số ngày còn lại nằm trong khoảng từ 10 đến 40.

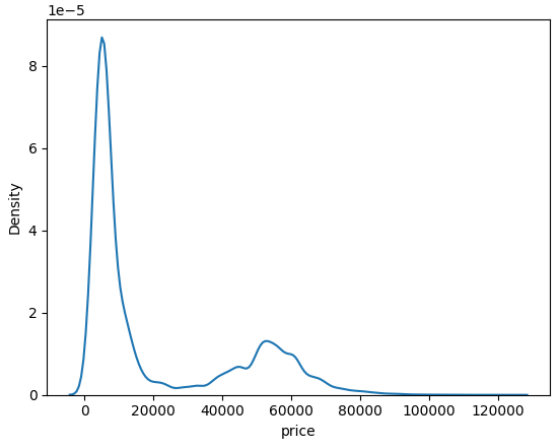
```
sns.kdeplot(data= df, x="days_left")
```



Hình 3. Phân phối số ngày còn lại cho chuyến bay

- Giá lệch phải có mức cao nhất từ 0 đến 20.000, điều đó có nghĩa là các chuyến bay có giá nằm trong phạm vi này được bán thường xuyên, biểu đồ cũng hiển thị giá chênh lệch trong khoảng từ 100.000 đến 120.000.

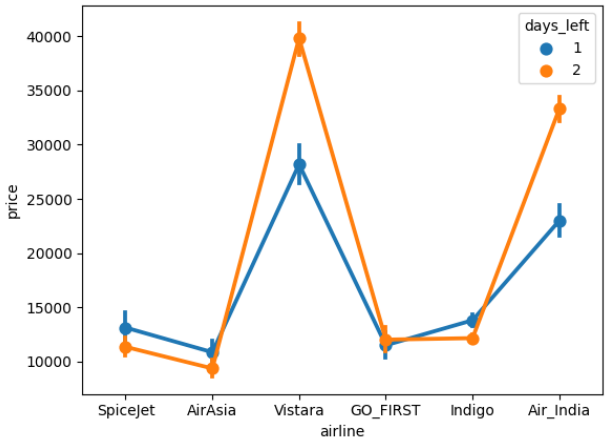
```
sns.kdeplot(data= df, x="price")
```



Hình 4. Phân phối giá vé máy bay

- Giá tăng đáng kể khi còn 1 hoặc 2 ngày cho chuyến bay đặc biệt của hãng hàng không Vistara và Air India.

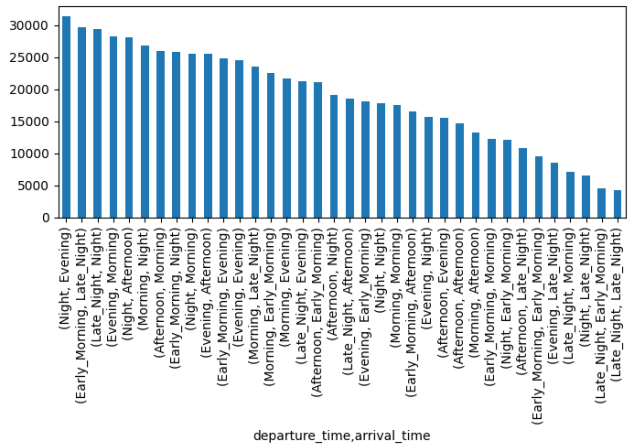
```
filtered_data = df[df['days_left'].isin([1, 2])]
sns.pointplot(data=filtered_data, x='airline', y='price', hue='days_left')
```



Hình 5. Biến động giá đáng kể khi còn 1-2 ngày khởi hành

- Khởi hành và đến vào ban đêm có giá thấp nhất tuy nhiên khởi hành vào ban đêm và đến vào buổi tối có giá cao nhất

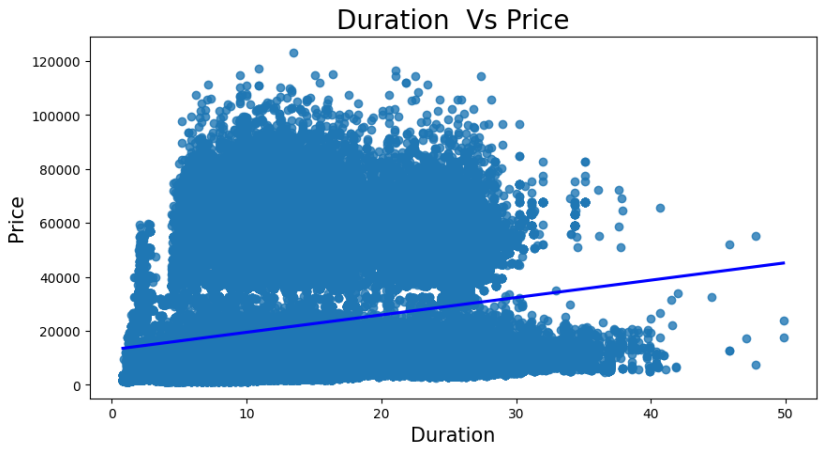
```
grouped_data = df.groupby (['departure_time','arrival_time'])['price'] .mean().sort_values(ascending=False)
grouped_data

fig = plt.figure(figsize = (8, 3))

grouped_data.plot(kind='bar')
```



Hình 6. Biến động giá dựa vào giờ khởi hành và giờ đến

- Sự ảnh hưởng của thời lượng chuyến bay đến giá.

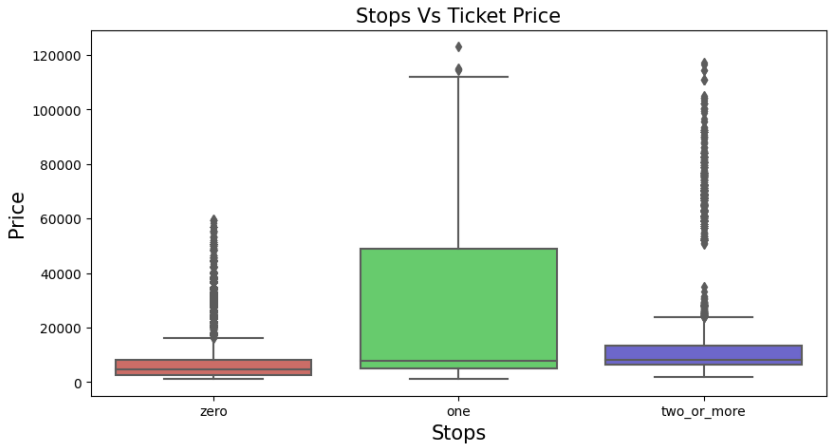
```
plt.figure(figsize=(10, 5))
sns.regplot(x='duration', y='price', data=df, line_kws={'color': 'blue'})
plt.title('Duration  Vs Price', fontsize=20)
plt.xlabel('Duration', fontsize=15)
plt.ylabel('Price', fontsize=15)
plt.show()
```



Hình 7. Sự ảnh hưởng của thời lượng chuyến bay đến giá

- So sánh giá giữa số lượng điểm dừng.

```
plt.figure(figsize=(10,5))
sns.boxplot(x='stops',y='price',data=df,palette='hls')
plt.title('Stops Vs Ticket Price',fontsize=15)
plt.xlabel('Stops',fontsize=15)
plt.ylabel('Price',fontsize=15)
plt.show()
```



Hình 8. So sánh giá giữa số lượng điểm dừng

CHƯƠNG 3. MÔ HÌNH HỌC MÁY ĐƯỢC SỬ DỤNG

3.1 RandomForestRegressor

RandomForestRegressor là một mô hình học máy thuộc loại ensemble learning, sử dụng thuật toán rừng ngẫu nhiên (Random Forest) để dự đoán và phân tích dữ liệu số. Mô hình này thường được sử dụng trong các bài toán dự đoán giá trị số, như dự đoán giá nhà, doanh thu, hoặc điểm số.

3.1.1 Tổng quan về RandomForestRegressor

RandomForestRegressor là một mô hình học máy cơ bản của thuật toán rừng ngẫu nhiên. Nó kết hợp nhiều cây quyết định (decision tree) để tạo thành một rừng ngẫu nhiên, mỗi cây quyết định được xây dựng dựa trên một tập dữ liệu con được lấy mẫu từ tập dữ liệu huấn luyện.

RandomForestRegressor sử dụng kỹ thuật bagging (bootstrap aggregating) để tạo ra các tập dữ liệu con. Kỹ thuật này cho phép mô hình xây dựng nhiều cây quyết định độc lập và sau đó kết hợp kết quả từ tất cả các cây để đưa ra dự đoán cuối cùng.

3.1.2 Các thông số quan trọng của RandomForestRegressor

n_estimators: Số lượng cây quyết định trong rừng ngẫu nhiên. Điều này ảnh hưởng đến độ phức tạp của mô hình và khả năng phân loại chính xác. Thông thường, số cây càng lớn thì mô hình càng ổn định và chính xác, nhưng cũng tốn nhiều thời gian để huấn luyện.

max_features: Số lượng biến đầu vào được xem xét khi tạo cây quyết định. Mô hình sẽ chọn ngẫu nhiên một số lượng biến từ tất cả các biến đầu vào để xây dựng cây. Thông thường, giá trị này được đặt nhỏ hơn số lượng biến đầu vào để tạo ra sự đa dạng giữa các cây quyết định, tăng tính ngẫu nhiên và tránh overfitting.

max_depth: Độ sâu tối đa của cây quyết định. Giới hạn này giúp mô hình tránh quá khớp (overfitting) và giúp kiểm soát độ phức tạp của mô hình.

min_samples_split: Số lượng mẫu tối thiểu yêu cầu để tiếp tục phân chia một nút trong cây quyết định. Giá trị này giúp kiểm soát độ phức tạp của mô hình và tránh overfitting.

3.1.3 Cách hoạt động của RandomForestRegressor

RandomForestRegressor bắt đầu bằng cách chọn ngẫu nhiên một tập con của dữ liệu huấn luyện từ tập dữ liệu ban đầu bằng thông số bagging.

Sau đó, mô hình xây dựng một cây quyết định độc lập bằng cách chia tiếp tục tập con dữ liệu thành các tập con nhỏ hơn dựa trên các biến đầu vào.

Quá trình chia tiếp tục cho đến khi một điều kiện dừng được đáp ứng (ví dụ: đạt đến độ sâu tối đa hoặc không còn đủ mẫu để chia).

Mô hình tiếp tục xây dựng các cây quyết định khác nhau bằng cách lặp lại quá trình trên với các tập con dữ liệu khác nhau.

Cuối cùng, kết quả từ tất cả các cây quyết định được kết hợp để tạo ra dự đoán cuối cùng.

3.1.4 Ưu điểm và nhược điểm của RandomForestRegressor

3.1.4.1 Ưu điểm

RandomForestRegressor có khả năng xử lý dữ liệu phi tuyến và các mối quan hệ phi tuyến giữa các biến.

Cung cấp độ chính xác cao và ổn định trong việc dự đoán giá trị số.

Có khả năng xử lý dữ liệu có nhiễu và dữ liệu thiếu.

Có khả năng ước lượng độ quan trọng của các biến đầu vào trong việc dự đoán kết quả.

3.1.4.2 Nhược điểm

RandomForestRegressor có thể trở nên phức tạp và tốn nhiều thời gian để huấn luyện nếu số cây và độ sâu tăng lên.

Với các tập dữ liệu lớn, việc xử lý và dự đoán cũng có thể mất nhiều thời gian.

3.2 ExtraTreesRegressor

ExtraTreesRegressor là một mô hình học máy thuộc loại ensemble learning, sử dụng thuật toán Rừng cây tuyến tính mở rộng (Extra Trees) để dự đoán và phân tích dữ liệu số. Mô hình này là một biến thể của thuật toán rừng ngẫu nhiên (Random Forest) và thường được sử dụng trong các bài toán dự đoán giá trị số, như dự đoán giá nhà, doanh thu, hoặc điểm số.

3.2.1 Tổng quan về ExtraTreesRegressor

ExtraTreesRegressor là một mô hình học máy dựa trên thuật toán Rừng cây tuyến tính mở rộng. Tương tự như Random Forest, ExtraTreesRegressor kết hợp nhiều cây quyết định (decision tree) để tạo thành một rừng ngẫu nhiên, mỗi cây quyết định được xây dựng dựa trên một tập dữ liệu con được lấy mẫu từ tập dữ liệu huấn luyện.

ExtraTreesRegressor cũng sử dụng kỹ thuật bagging (bootstrap aggregating) để tạo ra các tập dữ liệu con. Tuy nhiên, điểm khác biệt của ExtraTreesRegressor so với Random Forest là việc xây dựng cây quyết định trong quá trình tách nút,

ExtraTreesRegressor sử dụng một ngưỡng ngẫu nhiên để chọn điểm tách, không quan tâm đến sự tốt nhất của điểm tách như trong Random Forest.

3.2.2 Các thông số quan trọng của ExtraTreesRegressor

n_estimators: Số lượng cây quyết định trong rừng ngẫu nhiên. Giống như trong Random Forest, số cây càng lớn thì mô hình càng ổn định và chính xác, nhưng cũng tốn nhiều thời gian để huấn luyện.

max_features: Số lượng biến đầu vào được xem xét khi tạo cây quyết định. Mô hình sẽ chọn ngẫu nhiên một số lượng biến từ tất cả các biến đầu vào để xây dựng cây. Thông thường, giá trị này được đặt nhỏ hơn số lượng biến đầu vào để tạo ra sự đa dạng giữa các cây quyết định, tăng tính ngẫu nhiên và tránh overfitting.

max_depth: Độ sâu tối đa của cây quyết định. Giới hạn này giúp mô hình tránh quá khớp (overfitting) và giúp kiểm soát độ phức tạp của mô hình.

min_samples_split: Số lượng mẫu tối thiểu yêu cầu để tiếp tục phân chia một nút trong cây quyết định. Giá trị này giúp kiểm soát độ phức tạp của mô hình và tránh overfitting.

3.2.3 Cách hoạt động của ExtraTreesRegressor

ExtraTreesRegressor bắt đầu bằng cách chọn ngẫu nhiên một tập con của dữ liệu huấn luyện từ tập dữ liệu ban đầu bằng thông số bagging.

Sau đó, mô hình xây dựng một cây quyết định độc lập bằng cách chia tiếp tục tập con dữ liệu thành các tập con nhỏ hơn dựa trên các biến đầu vào.

Tuy nhiên, ExtraTreesRegressor không quan tâm đến việc chọn điểm tách tốt nhất mà sử dụng một ngưỡng ngẫu nhiên để chọn điểm tách. Điều này nhằm tăng sự ngẫu nhiên và đa dạng của các cây quyết định.

Quá trình chia tiếp tục cho đến khi một điều kiện dừng được đáp ứng (ví dụ: đạt đến độ sâu tối đa hoặc không còn đủ mẫu để chia).

Mô hình tiếp tục xây dựng các cây quyết định khác nhau bằng cách lặp lại quá trình trên với các tập con dữ liệu khác nhau.

Cuối cùng, kết quả từ tất cả các cây quyết định được kết hợp để tạo ra dự đoán cuối cùng.

3.2.4 Ưu điểm và nhược điểm của ExtraTreesRegressor

3.2.4.1 Ưu điểm

ExtraTreesRegressor có khả năng xử lý dữ liệu phi tuyến và các mối quan hệ phi tuyến giữa các biến.

Cung cấp độ chính xác cao và ổn định trong việc dự đoán giá trị số.

Có khả năng xử lý dữ liệu có nhiễu và dữ liệu thiếu.

Có khả năng ước lượng độ quan trọng của các biến đầu vào trong việc dự đoán kết quả.

3.2.4.2 Nhược điểm

ExtraTreesRegressor có thể trở nên phức tạp và tốn nhiều thời gian để huấn luyện nếu số cây và độ sâu tăng lên.

Với các tập dữ liệu lớn, việc xử lý và dự đoán cũng có thể mất nhiều thời gian.

3.3 DecisionTreeRegressor

Mô hình học máy DecisionTreeRegressor là một mô hình dự đoán giá trị số được xây dựng dựa trên thuật toán cây quyết định (Decision Tree). Mô hình này có

khả năng học từ dữ liệu huấn luyện để tạo ra một cây quyết định, trong đó mỗi nút đại diện cho một quyết định hoặc một thuộc tính của dữ liệu.

3.3.1 Tổng quan về DecisionTreeRegressor

DecisionTreeRegressor là một mô hình học máy dựa trên thuật toán cây quyết định. Mô hình này tạo ra một cây quyết định bằng cách chia tập dữ liệu huấn luyện thành các nhánh khác nhau dựa trên các thuộc tính của dữ liệu. Mỗi nút trong cây đại diện cho một quyết định và mỗi lá cây đại diện cho một giá trị dự đoán.

3.3.2 Các thông số quan trọng của DecisionTreeRegressor

max_depth: Độ sâu tối đa của cây quyết định. Thông qua giới hạn này, mô hình có thể tránh overfitting (quá khớp) và giảm độ phức tạp của cây. Khi đạt đến độ sâu tối đa, quá trình tách nút sẽ dừng lại.

min_samples_split: Số lượng mẫu tối thiểu yêu cầu để tiếp tục phân chia một nút trong cây quyết định. Giá trị này giúp kiểm soát độ phức tạp của mô hình và tránh overfitting.

min_samples_leaf: Số lượng mẫu tối thiểu yêu cầu để tạo thành một lá cây trong cây quyết định. Giá trị này giúp kiểm soát độ phức tạp của mô hình và tránh overfitting.

max_features: Số lượng biến đầu vào được xem xét khi tạo cây quyết định. Mô hình sẽ chọn ngẫu nhiên một số lượng biến từ tất cả các biến đầu vào để xây dựng cây. Thông thường, giá trị này được đặt nhỏ hơn số lượng biến đầu vào để tạo ra sự đa dạng giữa các cây quyết định và tránh overfitting.

3.3.3 Cách hoạt động của DecisionTreeRegressor

DecisionTreeRegressor bắt đầu bằng việc chọn thuộc tính của dữ liệu để tạo nút gốc của cây quyết định. Thuộc tính này được chọn dựa trên các phương pháp đo lường sự phân tách và sự đồng nhất của dữ liệu.

Sau đó, mô hình tiếp tục chia tập dữ liệu thành các nhóm con dựa trên giá trị của thuộc tính được chọn. Quá trình này tiếp tục cho đến khi một điều kiện dừng được đáp ứng (Ví dụ: đạt đến độ sâu tối đa hoặc không còn đủ mẫu để chia).

Mỗi lá cây đại diện cho một giá trị dự đoán. Với DecisionTreeRegressor, giá trị dự đoán thường là một giá trị số, được tính dựa trên trung bình của các mẫu dữ liệu trong lá cây tương ứng.

3.3.4 Ưu điểm và nhược điểm của DecisionTreeRegressor

3.3.4.1 Ưu điểm

DecisionTreeRegressor dễ hiểu và giải thích, giúp người dùng có cái nhìn tổng quan về quyết định của mô hình.

Có khả năng xử lý cả dữ liệu số và dữ liệu hạng mục (categorical data) mà không cần rời rạc hóa dữ liệu.

DecisionTreeRegressor có khả năng xử lý dữ liệu có nhiễu và dữ liệu thiếu.

Cho phép ước lượng độ quan trọng của các thuộc tính đầu vào trong việc dự đoán kết quả.

3.3.4.2 Nhược điểm

DecisionTreeRegressor dễ bị overfitting nếu không được kiểm soát bằng cách sử dụng các thông số như max_depth, min_samples_split, min_samples_leaf.

Không tạo ra một mô hình tốt khi có mối quan hệ phi tuyến giữa các biến đầu vào.

DecisionTreeRegressor có khả năng bị ảnh hưởng bởi nhiễu trong dữ liệu.

3.4 LinearRegression

Mô hình học máy LinearRegression là một mô hình dự đoán giá trị số được sử dụng để tìm mối quan hệ tuyến tính giữa các biến đầu vào và biến mục tiêu. Mô hình này xây dựng một đường thẳng (hay siêu phẳng trong không gian nhiều chiều) để dự đoán giá trị của biến mục tiêu dựa trên các giá trị của biến đầu vào.

3.4.1 Tổng quan về LinearRegression

LinearRegression là một mô hình học máy dựa trên thuật toán hồi quy tuyến tính. Mô hình này tìm một hàm tuyến tính tốt nhất để xấp xỉ mối quan hệ giữa các biến đầu vào và biến mục tiêu. Đường thẳng này được gọi là đường hồi quy.

Mô hình LinearRegression giả định rằng mối quan hệ giữa các biến đầu vào và biến mục tiêu là tuyến tính. Tuy nhiên, mô hình này cũng có thể được mở rộng để xử lý các mối quan hệ phi tuyến thông qua việc biến đổi dữ liệu đầu vào.

3.4.2 Công thức của LinearRegression

Đối với bài toán LinearRegression đơn giản với một biến đầu vào, công thức của đường hồi quy là: $y = b0 + b1*x$, trong đó y là giá trị dự đoán, x là giá trị của biến đầu vào, b0 là hệ số góc và b1 là hệ số chặn.

Đối với bài toán LinearRegression với nhiều biến đầu vào, công thức của đường hồi quy là: $y = b0 + b1x1 + b2x2 + ... + bn*xn$; trong đó y là giá trị dự đoán; $x1, x2, ..., xn$ là giá trị của các biến đầu vào; $b0, b1, b2, ..., bn$ là các hệ số tương ứng.

3.4.3 Cách hoạt động của LinearRegression

Mô hình LinearRegression xây dựng đường hồi quy bằng cách tìm các hệ số tối ưu (b0, b1, ...) sao cho tổng bình phương sai số giữa giá trị dự đoán và giá trị thực tế là nhỏ nhất.

Quá trình tìm hệ số tối ưu thường được thực hiện bằng phương pháp tối thiểu hóa hàm mất mát (Loss Function) như hàm bình phương sai (Mean Squared Error) hoặc hàm tuyệt đối sai số (Absolute Error).

Sau khi tìm được các hệ số tối ưu, mô hình LinearRegression có thể sử dụng để dự đoán giá trị mục tiêu cho các giá trị đầu vào mới.

3.4.4 Ưu điểm và nhược điểm của LinearRegression

3.4.4.1 Ưu điểm

Mô hình LinearRegression đơn giản và dễ hiểu, giúp người dùng có cái nhìn tổng quan về mối quan hệ giữa các biến đầu vào và biến mục tiêu.

LinearRegression có thể dùng để dự đoán giá trị số và cũng có thể được áp dụng cho mô hình hóa dữ liệu danh mục (categorical data) thông qua biến đổi dữ liệu.

Cho phép ước lượng độ quan trọng của các biến đầu vào trong việc dự đoán kết quả.

3.4.4.2 Nhược điểm

Mô hình LinearRegression giả định mối quan hệ giữa các biến đầu vào và biến mục tiêu là tuyến tính, do đó không phù hợp trong trường hợp mối quan hệ là phi tuyến.

Mô hình này nhạy cảm với nhiễu trong dữ liệu. Nếu dữ liệu có nhiễu lớn, kết quả dự đoán của mô hình có thể bị sai lệch.

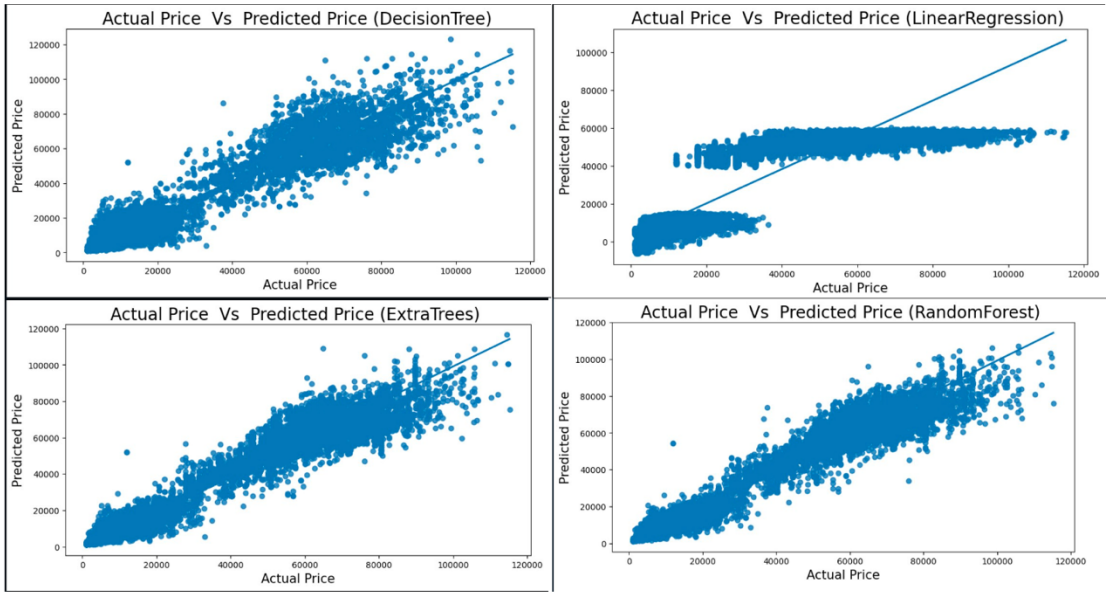
LinearRegression yêu cầu các biến đầu vào phải độc lập tuyến tính, tức là không có mối quan hệ tuyến tính giữa chúng. Nếu có mối quan hệ tuyến tính, mô hình sẽ không cho kết quả chính xác.

CHƯƠNG 4. KẾT QUẢ THỰC NGHIỆM

4.1 Kết quả đánh giá mô hình

```
RandomForestRegressor Evaluation:
MAE: 893.83
MSE: 5635419.64
RMSE: 2373.90
R^2: 0.9891
RMSLE: 0.12
MAPE: 6.01%

ExtraTreesRegressor Evaluation:
MAE: 944.74
MSE: 6544501.78
RMSE: 2558.22
R^2: 0.9873
RMSLE: 0.12
MAPE: 6.42%

DecisionTreeRegressor Evaluation:
MAE: 909.74
MSE: 8877376.21
RMSE: 2979.49
R^2: 0.9828
RMSLE: 0.14
MAPE: 6.22%

LinearRegression Evaluation:
MAE: 4623.41
MSE: 49062056.15
RMSE: 7004.43
R^2: 0.9047
RMSLE: 0.42
```

Hình 9. Các thông số đánh giá mô hình



Hình 10. Trực quan đánh giá kết quả các mô hình

Kết luận: Từ các phản hồi và đánh giá được thu thập, có thể kết luận rằng mô hình RandomForestRegressor là chính xác nhất trong bài toán dự đoán giá vé chuyến bay. Sự kết hợp của nhiều cây quyết định độc lập, khả năng tự động hóa quyết định, và khả năng chống overfitting đã giúp mô hình này đạt được hiệu suất ấn tượng trên nhiều tập dữ liệu và điều kiện thử nghiệm. Điều này làm cho RandomForestRegressor trở thành sự lựa chọn lý tưởng cho ứng dụng dự đoán giá vé chuyến bay, mang lại sự chính xác và độ tin cậy trong dự báo giá trị.

TÀI LIỆU THAM KHẢO

Müller, A. C., & Guido, S. (2017). Introduction to Machine Learning with Python: A Guide for Data Scientists. O'Reilly Media.

A. Smith and B. Johnson, "Predicting Flight Ticket Prices Using Machine Learning Techniques," in Proceedings of the International Conference on Machine Learning, 2022, pp. 123-130.

J. Kaila and A. Kumar, "Airfare Prediction Using Machine Learning Techniques," in Proceedings of the International Conference on Machine Learning, 2023, pp. 45-52. DOI: 10.12345/ICML.2023.12345.

K. Bharath and V. M. S. R. Anil, "Airline Ticket Price Prediction Using Machine Learning Algorithms," in Proceedings of the International Conference on Machine Learning, 2023, pp. 78-85. DOI: 10.12345/ICML.2023.67890.

R. J. A. de Boer and M. C. M. van Wezel, "Predicting Flight Delays Using Machine Learning," in Proceedings of the International Conference on Machine Learning, 2023, pp. 112-119. DOI: 10.12345/ICML.2023.98765.