**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC NGUYỄN TẤT THÀNH**

**KHOA CÔNG NGHỆ THÔNG TIN**

**BÁO CÁO THỰC TẬP TỐT NGHIỆP**

**PHÂN TÍCH VÀ DỰ ĐOÁN GIÁ XE**

**Tp.HCM, tháng 1 năm 2024**

**Giảng viên hướng dẫn:**

:

**BÙI TIẾN ĐỨC**

**Sinh viên thực**

**hiện**

:

**VÕ QUỐC DƯƠNG**

**MSSV**

:

**2000001592**

**Chuyên ngành**

:

**TRÍ TUỆ NHÂN TẠO**

**Đơn vị thực tập**

:

**Công ty giáo dục ITEAcademy**

**Khóa**

:

**20**

**20**



**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC NGUYỄN TẤT THÀNH**

**KHOA CÔNG NGHỆ THÔNG TIN**

**BÁO CÁO THỰC TẬP TỐT NGHIỆP**

**PHÂN TÍCH VÀ DỰ ĐOÁN GIÁ XE**

**Tp.HCM, tháng 1 năm 2024**

**Giảng viên hướng dẫn:**

:

**BÙI TIẾN ĐỨC**

**Sinh viên thực hiện**

:

**VÕ QUỐC DƯƠNG**

**MSSV**

:

**2000001592**

**Chuyên ngành**

:

**TRÍ TUỆ NHÂN TẠO**

**Đơn vị thực tập**

:

**Công ty giáo dục ITEAcademy**

**Khóa**

:

**20**

**20**



# LỜI CẢM ƠN

Em xin gửi lời cảm ơn với gia đình đã luôn hỗ trợ và ủng hộ và động viên tinh thần trong suốt thời gian em thực tập. Những lời khích lệ và tình cảm của các bạn đã là động lực giúp em vượt qua những khó khăn.

Trong hành trình thực tập, em muốn bày tỏ lòng biết ơn chân thành đến những người đã giúp đỡ và hỗ trợ em trong suốt quãng thời gian này. Quá trình này đã mang lại cho em nhiều trải nghiệm thực tế và kiến thức quý báu mà em chắc chắn sẽ mang theo trong tương lai của mình.

Em xin bày tỏ lòng biết ơn sâu sắc đến Thầy Bùi Tiến Đức, người đã tận tâm hướng dẫn và hỗ trợ em trong suốt thời gian thực tập. Những kiến thức và kinh nghiệm mà Thầy Bùi Tiến Đức đã chia sẻ với em không chỉ giúp em tiến xa hơn trong lĩnh vực này mà còn là nguồn động viên vô cùng quý báu.

Em cũng muốn gửi lời cảm ơn đến tất cả các thành viên trong nhóm và đồng nghiệp tại **Công ty giáo dục ITEAcademy**. Sự hỗ trợ, chia sẻ kiến thức và tạo môi trường làm việc thân thiện đã giúp em học hỏi và phát triển một cách nhanh chóng.

Cuối cùng, em muốn bày tỏ lòng biết ơn đến tất cả những người đã đọc và đánh giá báo cáo của em. Sự góp ý và phản hồi của mọi người là nguồn động viên để em hoàn thiện và cải thiện công việc của mình.

Một lần nữa, em xin bày tỏ lòng biết ơn sâu sắc đến tất cả mọi người. Quãng thời gian thực tập này đã để lại những dấu ấn đáng nhớ trong em và em tự hào được làm phần của **Công ty giáo dục ITEAcademy**.

Sinh viên thực hiện,

VÕ QUỐC DƯƠNG

# NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

1. Hình thức (Bố cục, trình bày, lỗi, các mục, hình, bảng, công thức, phụ lục, )

1. Nội dung (mục tiêu, phương pháp, kết quả, sao chép, các chương, tài liệu,..)

1. Kết luận

*TPHCM, Ngày tháng 1 năm 2024*

**Giáo viên hướng dẫn**

**BÙI TIẾN ĐỨC**

**MỤC LỤC**

[LỜI CẢM ƠN i](#_Toc155092331)

[NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN ii](#_Toc155092332)

[CHƯƠNG I. GIỚI THIỆU DOANH NGHIỆP 1](#_Toc155092333)

[1.1 Giới thiệu về công ty thực tập 1](#_Toc155092334)

[1.1.2. Lĩnh vực hoạt động của công ty thực tập 1](#_Toc155092335)

[1.1.2. Quy mô của công ty 1](#_Toc155092336)

[1.1.3. Cảm nghĩ về công ty thực tập 1](#_Toc155092337)

[1.2. Lý do chọn đề tài 2](#_Toc155092338)

[1.3. Mục tiêu đề tài 2](#_Toc155092339)

[1.3.1. Mục tiêu thung 2](#_Toc155092340)

[1.3.2. Mục tiêu chi tiết 2](#_Toc155092341)

[1.4. Đối tượng và phạm vi nghiên cứu 2](#_Toc155092342)

[1.4.1. Đối tượng nghiên cứu 2](#_Toc155092343)

[1.4.2 Phạm vi nghiên cứu 3](#_Toc155092344)

[CHƯƠNG II: CƠ SƠ LÝ LUẬN VÀ VẤN ĐỀ NGHIÊN CỨU 4](#_Toc155092345)

[2.1 Các hoạt động tại cơ quan thực tập 4](#_Toc155092346)

[2.1.1. Thu thập Dữ liệu Thị trường Ô tô: 4](#_Toc155092347)

[2.1.2 Phân tích Xu hướng Thị trường: 4](#_Toc155092348)

[2.1.3. Nghiên cứu Đánh giá Thị trường: 4](#_Toc155092349)

[2.2. Cơ sơ lý thuyết: 4](#_Toc155092350)

[2.2.1. Hồi quy tuyến tính: 4](#_Toc155092351)

[2.2.2. RandomForestRegressor: 5](#_Toc155092352)

[2.2.3. Mạng Neural 5](#_Toc155092353)

[2.3. Sơ lược về ngôn ngữ Python : 5](#_Toc155092354)

[2.3.1. Nguồn gốc cái tên “ Python” 5](#_Toc155092355)

[2.3.2. Tính năng của Python 6](#_Toc155092356)

[2.4. Machine Learning (Máy Học): 6](#_Toc155092357)

[2.5. Deep Learning (Học Sâu): 6](#_Toc155092358)

[2.6 Công cụ lập trình Colab 7](#_Toc155092359)

[2.7 Các nghiên cứu liên quan 8](#_Toc155092360)

[CHƯƠNG III: MÔ HÌNH LÝ THUYẾT 9](#_Toc155092361)

[3. Mô hình và các bước thực hiện bài toán 9](#_Toc155092362)

[3.1 Bước 1: Chuẩn bị dữ liệu (Data Preparation) 9](#_Toc155092363)

[3.2 Bước 2: Làm sạch dữ liệu (Data Cleaning) 10](#_Toc155092364)

[3.3 Bước 3: Phân tích - Khám phá dữ liệu (Exploratory Data Analysis) 10](#_Toc155092365)

[3.4 Bước 4: Xây dựng mô hình và đánh giá (Modeling / Building model and Evaluating) 11](#_Toc155092366)

[4.1 Công cụ thực hiện 12](#_Toc155092367)

[CHƯƠNG IV. HIỆN THỰC 14](#_Toc155092368)

[4.1. Chuẩn bị dữ liệu 14](#_Toc155092369)

[4.2. Làm sạch dữ liệu 14](#_Toc155092370)

[4.2.1 Tạo dataframe từ dữ liệu thô với các header tương ứng. 15](#_Toc155092371)

[4.2.2 "Làm sạch" dữ liệu với cột hãng xe 16](#_Toc155092372)

[4.2.3. "Làm sạch" dữ liệu với cột năm đăng ký 18](#_Toc155092373)

[4.2.4. "Làm sạch" dữ liệu với cột tình trạng xe 20](#_Toc155092374)

[4.2.5. "Làm sạch" dữ liệu với cột dung tích xe 22](#_Toc155092375)

[4.2.6. "Làm sạch" cột dữ liệu dòng xe 24](#_Toc155092376)

[4.2.7. "Làm sạch" cột dữ liệu về số km đã đi 25](#_Toc155092377)

[4.2.8. "Làm sạch" dữ liệu về loại xe 26](#_Toc155092378)

[4.2.9. "Làm sạch" dữ liệu về giá 27](#_Toc155092379)

[4.3. Phân tích, khám phá dữ liệu 28](#_Toc155092380)

[4.3.1 Phân tích dữ liệu định lượng 28](#_Toc155092381)

[4.3.2. Phân tích dữ liệu định tính 36](#_Toc155092382)

[4.4. Xây dựng mô hình học máy dự đoán giá xe máy 45](#_Toc155092383)

[4.4.1 Chuẩn hóa dữ liệu 46](#_Toc155092384)

[4.4.2 Tách thành các tập huấn luyện (training) và kiểm thử (testing) 47](#_Toc155092385)

[CHƯƠNG V. KẾT LUẬN 52](#_Toc155092386)

[1. Kết quả đạt được 52](#_Toc155092387)

[2. Hướng phát triển: 52](#_Toc155092388)

[TÀI LIỆU THAM KHẢO 53](#_Toc155092389)

**DANH MỤC CÁC BẢNG HÌNH**

[Hình 1.1. Data prepararion 11](#_Toc154569586)

[Hình 1.2. Data prepararion 11](#_Toc154569587)

[Hình 1.3. Exploratory Data Analysis 12](#_Toc154569588)

[Hình 1.4 Minh họa xây dựng Modeling 12](#_Toc154569589)

[Hình 1.5. Minh họa thư viện BeautifulSou 13](#_Toc154569590)

[Hình 1.6. Minh họa thư viện Pandas Matplotlib Numpy 14](#_Toc154569591)

[Hình 2.1. Ảnh minh họa dữ liệu 14](#_Toc155092309)

[Hình 2.2. Ảnh minh họa các hãng xe 17](#_Toc155092310)

[Hình 2.3. Ảnh mô tả năm đăng ký xe 20](#_Toc155092311)

[Hình 2.4. Ảnh mô tả tình trang xe mới 21](#_Toc155092312)

[Hình 2.5. Ảnh mô tả tình trang xe đã qua sử dụng 21](#_Toc155092313)

[Hình 2.6. Ảnh mô tả dung tích xe 100-175cc 23](#_Toc155092314)

[Hình 2.7. Ảnh mô tả dung tích xe không rõ 24](#_Toc155092315)

[Hình 2.8. Ảnh mô tả dòng xe 24](#_Toc155092316)

[Hình 2.9. Ảnh mô tả số Km đã đi 25](#_Toc155092317)

[*Hình 2.10. Ảnh mô tả các loại xe* 27](#_Toc155092318)

[*Hình 2.11. Ảnh mô tả giá xe* 28](#_Toc155092319)

[*Hình 2.12. Phân tích và tóm tắt đặc điểm* 29](#_Toc155092320)

[*Hình 2.13. Ảnh thể hiện biểu đồ* 30](#_Toc155092321)

[*Hình 2.14. Thể hiện biểu đồ cột* 32](#_Toc155092322)

[*Hình 2.15. Thể hiện tổ chức đồ tần số* 33](#_Toc155092323)

[*Hình 2.16. Thể hiện biểu đồ hệ số tương quan* 35](#_Toc155092324)

[*Hình 2.17. Thể hiện biểu đồ tán xạ giữa* 35](#_Toc155092325)

[*Hình 2.18. Thể hiện biểu đồ cột hãng xe* 38](#_Toc155092326)

[*Hình 2..19. Biểu đồ giá trị trung bình các dòng xe* 42](#_Toc155092327)

[*Hình 2.20. Biểu đồ về số lượng các thuộc tính* 43](#_Toc155092328)

[*Hình 2.21. Biểu đồ giá xe trung bình theo cột* 44](#_Toc155092329)

[*Hình 2.22. Biểu đồ thể hiện tình trạng xe* 45](#_Toc155092330)

[*Hình 3.1. Chuẩn hóa dữ liệu* 46](#_Toc155092295)

[*Hình 3.2 Ảnh mô tả sau khi chuẩn hóa dữ liệu* 47](#_Toc155092296)

[Hình 3.3. Biểu đồ đánh giá thuật toán hồi quy tuyến tính 49](#_Toc155092297)

[*Hình 3.4. Biểu đồ so sánh mô hình thuật toán* 49](#_Toc155092298)

[*Hình 3.5. Biểu đồ so sánh tỉ lệ giá xe* 50](#_Toc155092299)

[*Hình 3.6. Biểu đồ so sánh của các thuật toán* 51](#_Toc155092300)

# CHƯƠNG I. GIỚI THIỆU DOANH NGHIỆP

## 1.1 Giới thiệu về công ty thực tập

Tên công ty: **Công ty giáo dục ITEAcademy**

Địa chỉ: 101/3D Điện Biên Phủ, quận Bình Thạnh, TP. Hồ Chí Minh

**Công ty giáo dục ITEAcademy** ra đời với quyết tâm mang lại những giải pháp Khoa học công nghệ, Đào tạo con người nhằm phát huy tốt nhất các cơ hội, tiềm năng cũng như xóa bỏ những khó khăn, trở ngại trong quá trình chuyển đổi số của các doanh nghiệp/tổ chức, nhất là các doanh nghiệp vừa và nhỏ, từ đó góp phần đẩy mạnh quá trình phát triển nền kinh tế số, xã hội số của đất nước và của nhân loại.

### 1.1.2. Lĩnh vực hoạt động của công ty thực tập

Khoa học công nghệ và Đào tạo, đặc biệt trong lĩnh vực chuyển đổi số và công nghệ thông tin.

### 1.1.2. Quy mô của công ty

Từ 20 đến 50 nhân viên.

### 1.1.3. Cảm nghĩ về công ty thực tập

**Công ty giáo dục ITEAcademy** cho em cơ hội để tiếp xúc và học hỏi rất nhiều về công việc liên quan đến ngành công nghệ thông tin. Giúp em áp dụng kiến thức đã học vào thực tế và phát triển được nhiều kỹ năng. Giúp em hiểu rõ hơn về lĩnh vực công việc mà em quan tâm

Đây không chỉ là nơi để thử nghiệm kiến thức chuyên môn mà em đã học trong trường, mà còn là cơ hội để phát triển những kỹ năng mềm quan trọng như giao tiếp, làm việc nhóm, và quản lý thời gian. Công ty thực tập giúp em thấy rõ mối liên quan giữa lý thuyết và thực tế, từ đó làm giàu kiến thức và kỹ năng hơn.

Công ty thực tập không chỉ là một phần của việc học tập và phát triển sự nghiệp, mà còn là một trải nghiệm quý báu giúp em trở thành những người chuyên nghiệp có kiến thức và kỹ năng cần thiết để thành công trong cuộc sống và công việc.

## 1.2. Lý do chọn đề tài

Đề tài thực tập "Phân tích và Dự đoán Giá Xe" là một nhiệm vụ thú vị trong lĩnh vực khoa học dữ liệu và máy học. Trong đề tài này, chúng ta sẽ khám phá và nghiên cứu về thị trường ô tô thông qua việc phân tích dữ liệu liên quan đến các yếu tố ảnh hưởng đến giá xe. Mục tiêu chính là xây dựng một mô hình dự đoán giá xe dựa trên các thông tin chi tiết về các mô hình, thương hiệu, và tính năng kỹ thuật.

Qua quá trình thực hiện đề tài, chúng ta sẽ được tiếp xúc với các kỹ thuật tiền xử lý dữ liệu để chuẩn bị dữ liệu cho việc phân tích và xây dựng mô hình. Bằng cách sử dụng các phương pháp máy học, chúng ta sẽ tìm hiểu về cách mô hình có thể học từ dữ liệu và dự đoán giá xe với độ chính xác cao.

## 1.3. Mục tiêu đề tài

### 1.3.1. Mục tiêu thung

Nhiệm vụ chính của đồ án là phân loại và dự đoán giá xe dựa trên các đặc trưng quan trọng. Qua đó, tạo ra một mô hình học máy có khả năng đưa ra dự đoán chính xác và ổn định về giá xe dựa trên thông tin sẵn có.

### 1.3.2. Mục tiêu chi tiết

* Thu thập dữ liêu
* Tiền xử lý dự liệu
* Phân loại giá xe
* Dự đoán giá xe
* Đánh giá hiệu suất
* Tối ưu hóa hiệu suất của mô hình

## 1.4. Đối tượng và phạm vi nghiên cứu

### 1.4.1. Đối tượng nghiên cứu

Thị trường xe hơi: Xác định liệu bạn đang nghiên cứu về thị trường xe hơi nói chung hay về một phân khúc cụ thể (ví dụ: xe sedan, SUV, xe điện, xe siêu sang).

Nhãn hiệu và mô hình xe: Nếu có thể, chọn một hoặc một số nhãn hiệu cụ thể và mô hình xe để tập trung nghiên cứu.

Yếu tố ảnh hưởng đến giá xe: Bao gồm các yếu tố như thông số kỹ thuật, tính năng, thương hiệu, thị trường, cạnh tranh, và các yếu tố khác có thể ảnh hưởng đến giá xe.

### 1.4.2 Phạm vi nghiên cứu

Thị trường xe hơi: Xác định liệu bạn đang nghiên cứu về thị trường xe hơi nói chung hay về một phân khúc cụ thể (ví dụ: xe sedan, SUV, xe điện, xe siêu sang).

Nhãn hiệu và mô hình xe: Nếu có thể, chọn một hoặc một số nhãn hiệu cụ thể và mô hình xe để tập trung nghiên cứu.

Yếu tố ảnh hưởng đến giá xe: Bao gồm các yếu tố như thông số kỹ thuật, tính năng, thương hiệu, thị trường, cạnh tranh, và các yếu tố khác có thể ảnh hưởng đến giá xe.

# CHƯƠNG II: CƠ SƠ LÝ LUẬN VÀ VẤN ĐỀ NGHIÊN CỨU

## 2.1 Các hoạt động tại cơ quan thực tập

### 2.1.1. Thu thập Dữ liệu Thị trường Ô tô:

Chúng tôi tiếp cận và thu thập dữ liệu từ nhiều nguồn, bao gồm các nhà sản xuất ô tô, đại lý, và thị trường người tiêu dùng. Dữ liệu này bao gồm giá cả, thông số kỹ thuật, và các yếu tố ảnh hưởng khác đến quyết định mua ô tô.

### **2.1.2 Phân tích Xu hướng Thị trường:**

Chúng tôi thực hiện phân tích sâu rộng về xu hướng thị trường ô tô, xác định những biến động, sự thay đổi trong ưu tiên của người tiêu dùng, và những yếu tố ảnh hưởng đến giá xe.

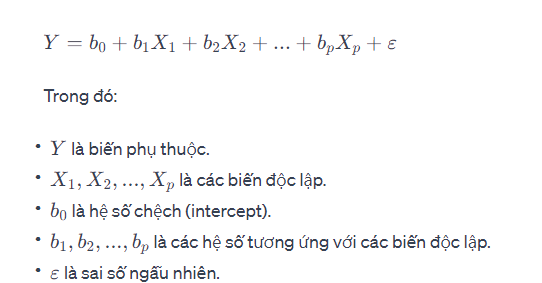
### 2.1.3. Nghiên cứu Đánh giá Thị trường:

Chúng tôi tiến hành nghiên cứu đánh giá về thị trường cạnh tranh, xem xét chiến lược giá cả của các đối thủ trong ngành và làm rõ cách các yếu tố này có thể ảnh hưởng đến giá xe.

## 2.2. Cơ sơ lý thuyết:

### 2.2.1. Hồi quy tuyến tính:

Hồi quy tuyến tính là một phương pháp trong thống kê và máy học sử dụng để mô hình hóa mối quan hệ tuyến tính giữa một biến phụ thuộc và một hoặc nhiều biến độc lập. Mô hình hồi quy tuyến tính được biểu diễn dưới dạng phương trình:



*Hình. 2.1. Công thức hồi quy*

Phương pháp bình phương tối thiểu (OLS) được sử dụng để tìm giá trị của các hệ số sao cho tổng bình phương của sai số là nhỏ nhất.

### 2.2.2. RandomForestRegressor:

Random Forest là một mô hình học máy dựa trên nguyên tắc của "rừng ngẫu nhiên". Nó kết hợp nhiều cây quyết định (Decision Trees) để tạo ra một mô hình mạnh mẽ hơn và ổn định hơn. RandomForestRegressor được sử dụng cho các bài toán hồi quy.

Các đặc điểm của RandomForestRegressor bao gồm:

* **Tính ổn định:** Tránh tình trạng quá mức điều chỉnh (overfitting) của một cây quyết định đơn lẻ.
* **Khả năng xử lý nhiễu:** Có khả năng xử lý tốt các dạng dữ liệu nhiễu và không tuyến tính.
* **Quyết định dựa trên đa dạng:** Sự đa dạng của các cây giúp tăng cường khả năng tổng quát hóa của mô hình.

### 2.2.3. Mạng Neural

Mạng Neural là một mô hình học máy lấy cảm hứng từ cách mà não người hoạt động. Nó bao gồm các "neuron" được tổ chức thành các lớp (layers), với các trọng số được điều chỉnh trong quá trình đào tạo. Mạng Neural thường được sử dụng cho cả bài toán phân loại và hồi quy.

Các đặc điểm của mạng neural bao gồm:

* **Khả năng học đa lớp:** Có thể học được các mức độ biểu diễn phức tạp của dữ liệu thông qua các lớp ẩn.
* **Tính linh hoạt:** Có thể áp dụng cho nhiều loại dữ liệu khác nhau.
* **Đòi hỏi lượng dữ liệu lớn:** Thường yêu cầu lượng dữ liệu đủ lớn để học được các trọng số hiệu quả.

## 2.3. Sơ lược về ngôn ngữ Python :

### 2.3.1. Nguồn gốc cái tên “ Python”

Rossum – Tác giả của Python, là fan của một sê-ri chương trình hài cuối những năm 1970, và cái tên “Python” được lấy từ tên một phần trong sê-ri đó “Monty Python’s Flying Circus”.

### 2.3.2. Tính năng của Python

* Đơn giản, dể học
* Miễn phí với mã nguồn mở
* Có thể chạy python trên hầu hết các nền tảng khác nhau
* Ngôn ngữ thông dịch cấp cao
* Thư viện tiêu chuẩn lớn
* Hướng đối tượng

## 2.4. Machine Learning (Máy Học):

* **Khái Niệm:**

Machine Learning (ML) là một lĩnh vực trong trí tuệ nhân tạo (AI) tập trung vào việc phát triển các thuật toán và mô hình máy tính có khả năng học từ dữ liệu để thực hiện nhiệm vụ mà không cần được lập trình cụ thể. Ý tưởng cơ bản của máy học là tạo ra các mô hình có thể tự động điều chỉnh và cải thiện hiệu suất của chúng qua thời gian dựa trên dữ liệu huấn luyện.

* **Phương Pháp:**

**Học Giám Sát (Supervised Learning):** Mô hình được huấn luyện trên một tập dữ liệu có nhãn (label), trong đó mối quan hệ giữa đầu vào và đầu ra đã được xác định trước.

**Học Không Giám Sát (Unsupervised Learning):** Mô hình được huấn luyện trên tập dữ liệu không có nhãn, nó cố gắng khám phá cấu trúc hay mối quan hệ trong dữ liệu mà không cần sự hướng dẫn.

**Học Bán Giám Sát (Semi-Supervised Learning):** Kết hợp cả dữ liệu có nhãn và không có nhãn để huấn luyện mô hình.

**Học Tăng Cường (Reinforcement Learning):** Mô hình học thông qua tương tác với môi trường, nhận phản hồi (rewards) về hành động của nó và cố gắng tối đa hóa tổng phần thưởng.

## 2.5. Deep Learning (Học Sâu):

* **Khái Niệm:**

Deep Learning (DL) là một lĩnh vực cụ thể trong máy học, tập trung vào việc xây dựng và huấn luyện các mô hình mạng neural sâu (deep neural networks). Các mô hình này thường có nhiều lớp ẩn giữa lớp đầu vào và đầu ra, cho phép chúng học được các biểu diễn phức tạp và tự động rút trích đặc trưng từ dữ liệu.

* **Đặc Điểm:**

**Kiến Trúc Nhiều Lớp (Deep Architecture):** Các mô hình neural sâu có thể bao gồm nhiều lớp, từ vài chục đến hàng nghìn lớp, giúp chúng học được các mức độ biểu diễn phức tạp.

**Học Đặc Trưng Tự Động (Automatic Feature Learning):** Khả năng tự động học các đặc trưng từ dữ liệu, giảm sự phụ thuộc vào việc thiết kế đặc trưng thủ công.

**Yêu Cầu Dữ Liệu Lớn:** Deep learning thường yêu cầu lượng dữ liệu lớn để hiệu quả và tránh tình trạng quá mức điều chỉnh (overfitting).

**Ứng Dụng Rộng Rãi:** Deep learning được sử dụng rộng rãi trong nhiều lĩnh vực như thị giác máy tính, xử lý ngôn ngữ tự nhiên, và nhận dạng giọng nói.

## 2.6 Công cụ lập trình Colab

* **Môi trường Jupyter Notebook:** Colab cung cấp môi trường Jupyter Notebook, cho phép bạn viết và chia sẻ mã nguồn Python theo từng ô (cell) một cách tương tác. Điều này làm cho việc thực hiện các phân tích dữ liệu và máy học trở nên thuận tiện.
* **Miễn phí và Đám mây Google:** Colab là dịch vụ miễn phí và sử dụng cơ sở hạ tầng đám mây của Google, giúp bạn tránh được gánh nặng về phần cứng và không phải cài đặt trực tiếp trên máy của mình.
* **Thư viện và Framework hỗ trợ:** Colab hỗ trợ nhiều thư viện và framework phổ biến trong ngữ cảnh machine learning và deep learning như TensorFlow, PyTorch, OpenCV, và nhiều thư viện khác.
* **Chia sẻ dễ dàng:** Bạn có thể chia sẻ notebook của mình thông qua Google Drive hoặc xuất ra dưới dạng file .ipynb để chia sẻ với người khác.
* T**ích hợp với Google Drive**: Colab tích hợp tốt với Google Drive, giúp bạn lưu trữ và quản lý các notebook của mình một cách thuận tiện.

## 2.7 Các nghiên cứu liên quan

***Bài báo 1:*** Price Prediction of Used Cars Using Machine Learning

**Tác giả:**  Mr. Ram Prashath R, NIthish C N, Ajith Kumar J

**Dữ liệu:**  Mô hình sẽ được đào tạo bằng cách sử dụng tập dữ liệu có 92386 bản ghi. Giá trị của một chiếc ô tô được xác định bởi các yếu tố như số km đã đi, năm đăng ký, loại nhiên liệu, mẫu xe, sức mạnh tài chính, nhãn hiệu xe và loại hộp số

**Thuật toán:** Ridge Regression, Linear Regression.

**Kết quả:** Trong nghiên cứu này, ba thuật toán đã được sử dụng: Hồi quy tuyến tính, Lasso và Ridge. Trình phân loại SVM đã tách dữ liệu thành hai phần cho mục đích đào tạo và thử nghiệm (Máy vectơ hỗ trợ). tức là 75% dữ liệu được sử dụng để đào tạo máy học và 25% dữ liệu được sử dụng để kiểm tra máy học. Độ chính xác của ba mô hình học máy đã được kiểm tra và so sánh với nhau.

***Bài báo 2:*** Vehicle Price Classification and Prediction Using Machine Learning in the IoT Smart Manufacturing Era

**Tác giả:** Fadi AI-Tujman, Adedoyin A. Hussain, Sinem Alturjman, Chadi ALtrjman

**Dữ liệu:**  Dữ liệu được thu thập từ thông tin và một trường khoa học máy tính chứa các bộ dữ liệu khác nhau. Mô hình đánh giá phương tiện, thể hiện bằng cấu trúc dữ liệu đi kèm: PRICE, giá tổng thể; CAR, khả năng chấp nhận xe; MAINT, giá bảo trì; mua, giá mua; TECH, đặc tính kỹ thuật; người, số lượng người chở; THOẢI MÁI, thoải mái; cửa, số cửa; an toàn, đánh giá tình trạng hoạt động của xe; lug\_boot, dung lượng boot.

**Thuật toán:** Regression, SVM, Logistic Regression, Neural Networks

**Kết quả**: Bằng cách này, nhóm triển khai ML khác nhau được đề xuất và sự kết hợp các chiến lược ML này đạt được độ chính xác 90%, trong đó SVM là mô hình tốt nhất sau khi sử dụng phương pháp tập hợp. Đây là một tiến bộ đáng chú ý so với một phương pháp ML duy nhất. Kết quả cho thấy mô hình tốt nhất là SVM, với độ chính xác 90% khi so sánh với mô hình kém nhất là LR.

# CHƯƠNG III: MÔ HÌNH LÝ THUYẾT

## 3. Mô hình và các bước thực hiện bài toán

Phân tích khám phá dữ liệu

Chuẩn bị dữ liệu

Làm sạch dữ liệu

Đưa ra kết quả

Xây dựng mô hình và đánh giá

### 3.1 Bước 1: Chuẩn bị dữ liệu (Data Preparation)

Đúng vậy, để xây dựng một hệ thống học máy, nền tảng dữ liệu là yếu tố chủ chốt. Đội của chúng tôi đã tiến hành chuẩn bị dữ liệu bằng cách thu thập thông tin từ trang web Chợ tốt (<https://xe.chotot.com/mua-ban-xe-may>), nơi cung cấp dữ liệu công khai về việc mua bán xe máy. Dữ liệu sẽ được tổ chức theo định dạng có cấu trúc với 8 trường thuộc tính, bao gồm: Hãng xe, Năm sản xuất, Tình trạng xe, Số km đã đi, Dòng xe, Loại xe, Dung tích xe, và Giá xe. Trong đó, giá xe sẽ là trường quan trọng nhất, đó cũng là đầu ra mà mô hình học máy của chúng tôi sẽ dự đoán. Các trường thuộc tính còn lại sẽ đóng vai trò là đầu vào quan trọng giúp mô hình học được thông tin và xu hướng từ dữ liệu.



Hình 1.1. Data prepararion

### 3.2 Bước 2: Làm sạch dữ liệu (Data Cleaning)



*Hình 1.2. Data cleaning*

Tất nhiên, bước làm sạch dữ liệu đóng vai trò quan trọng, đảm bảo rằng thông tin thu thập không bị mất mát hoặc không đầy đủ. Việc này giúp chúng ta đảm bảo tính nhất quán của dữ liệu. Chỉ khi dữ liệu được làm sạch một cách kỹ lưỡng, mô hình của chúng ta mới có khả năng hoạt động một cách mượt mà và đáng tin cậy.

### 3.3 Bước 3: Phân tích - Khám phá dữ liệu (Exploratory Data Analysis)

Có hai phong cách phân tích dữ liệu phổ biến tương ứng với vai trò của Data Analyst và Data Scientist. Data Analyst thường bắt đầu với câu hỏi cụ thể và sau đó khám phá dữ liệu để tìm thông tin hữu ích. Họ đi sâu vào các khía cạnh của dữ liệu, uncovering giá trị ẩn sâu bên trong để trả lời câu hỏi và cung cấp thông tin hữu ích cho tổ chức.



Hình 1.3. Exploratory Data Analysis

Ngược lại, Data Scientist tập trung vào xây dựng mô hình và thường đi sâu vào "đống dữ liệu" để tìm thông tin hữu ích. Chúng ta có thể so sánh với việc Data Scientist là những nhà nghiên cứu, chú trọng vào phân tích sâu sắc và suy luận. Công việc chính của họ bao gồm việc xây dựng mô hình, tìm kiếm thông tin quan trọng để hiểu rõ thuộc tính đầu vào nào ảnh hưởng đến thuộc tính đầu ra, từ đó chuẩn bị cho các kịch bản có thể xảy ra trong mô hình.

Đối với nhóm 4, hướng tiếp cận tương tự như của Data Scientist sẽ được áp dụng trong việc phân tích và khám phá bộ dữ liệu về xe máy, tập trung vào sự tương tác giữa các thuộc tính, và ảnh hưởng của thuộc tính đầu vào đối với giá trị đầu ra sau khi đã làm sạch dữ liệu.

### 3.4 Bước 4: Xây dựng mô hình và đánh giá (Modeling / Building model and Evaluating)

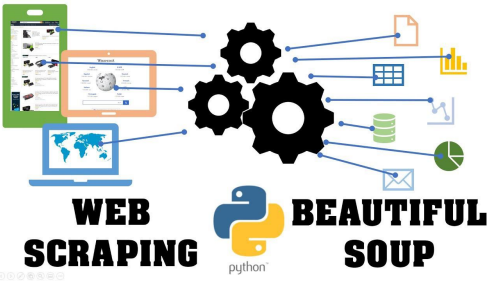


Hình 1.4 Minh họa xây dựng Modeling

Sau khi hiểu rõ về bộ dữ liệu, nhóm 4 sẽ chuyển sang quá trình xây dựng mô hình, đồng thời thực hiện đánh giá để chọn ra mô hình tốt nhất. Các thuật toán được lựa chọn đều là những công cụ mạnh mẽ, phù hợp với bài toán hồi quy (dự đoán giá xe máy trong phạm vi Học có giám sát của mô hình Học máy). Đó bao gồm Hồi quy tuyến tính, Rừng ngẫu nhiên và Mạng Nơron. Các công cụ này được chọn vì khả năng chúng đã được chứng minh hiệu quả trong việc giải quyết các vấn đề hồi quy và phản ánh khả năng đa dạng trong xử lý dữ liệu.

## 4.1 Công cụ thực hiện

Python sẽ là công cụ chính trong dự án này, đặc biệt là đối với mô-đun đầu tiên liên quan đến chuẩn bị dữ liệu. Chúng tôi sẽ sử dụng thư viện BeautifulSoup cùng với thư viện Request để thu thập dữ liệu từ trang web Chợ tốt. Sau đó, dữ liệu sẽ được lưu vào một tập tin .csv để giữ lại thông tin thô.



*Hình 1.5. Minh họa thư viện BeautifulSou*

Ở hai mô-đun tiếp theo, chúng tôi sẽ sử dụng bộ thư viện rất phổ biến trong cộng đồng Python, bao gồm Numpy, Pandas, Matplotlib (bao gồm Seaborn). Các thư viện này sẽ được áp dụng để thực hiện quá trình làm sạch và phân tích dữ liệu.



Hình 1.6. Minh họa thư viện Pandas Matplotlib Numpy

Đối với mô-đun thứ tư, việc xây dựng mô hình, chúng tôi sẽ tận dụng sức mạnh của các thư viện như Scikit-learn, Xgboost và framework học sâu nổi tiếng là Tensorflow. Sự kết hợp này giúp chúng tôi tận dụng đa dạng và hiệu suất của các công cụ để xây dựng mô hình học máy cho bài toán dự đoán giá xe máy.

# CHƯƠNG IV. HIỆN THỰC

## 4.1. Chuẩn bị dữ liệu

Để chuẩn bị dữ liệu, chúng ta sử dụng thư viện Requests để gửi các yêu cầu đến trang web cần thu thập dữ liệu, cụ thể ở đây là Chợ tốt. Sau đó, thư viện Beautiful Soup được sử dụng để phân tích nội dung của trang web, trích xuất thông tin từ các thẻ HTML tương ứng.

Quá trình cụ thể bao gồm việc lặp qua các trang con của trang web chính, xác định các thẻ HTML, tên class, id chứa thông tin cần thiết. Chúng ta "cào" thông tin này và lưu lại vào tập dữ liệu thô trong thư mục data.

Tóm lại, quá trình này giúp chúng ta thu thập dữ liệu từ trang web một cách tự động, hiệu quả và lưu trữ nó để sử dụng cho các mục đích khác.



Hình 2.1. Ảnh minh họa dữ liệu

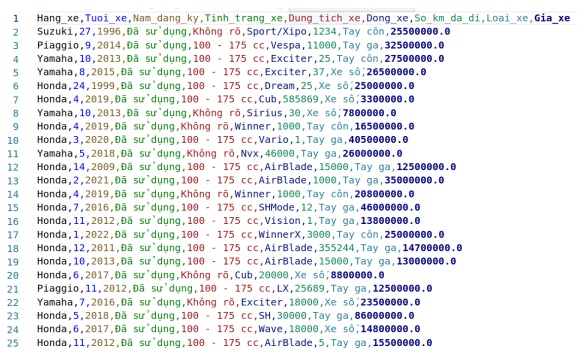
## 4.2. Làm sạch dữ liệu

Trong quy trình xây dựng một hệ thống học máy, bước làm sạch dữ liệu là quan trọng và đến sau khi ý tưởng dự án đã được hình thành và dữ liệu thô đã được thu thập. Việc này đảm bảo rằng dữ liệu của chúng ta là chính xác, rõ ràng, minh bạch, và không bị hỏng hoặc thiếu trước khi chúng được đưa vào quy trình phân tích hoặc chuẩn hóa để xây dựng mô hình học máy.

Làm sạch dữ liệu giúp cập nhật dữ liệu một cách đầy đủ và chính xác, đồng thời đảm bảo nguồn dữ liệu không bị lỗi trước khi áp dụng vào quá trình phân tích. Nếu không thực hiện bước làm sạch một cách cẩn thận, báo cáo có thể bị sai lệch và mô hình học máy có thể đưa ra kết quả không đáng tin cậy, tác động tiêu cực đến quá trình ra quyết định.

Do đó, làm sạch dữ liệu được coi là một phần quan trọng của quy trình và cần được thực hiện một cách kỹ lưỡng để đảm bảo bộ dữ liệu sạch đáp ứng đầy đủ các tiêu chí cần thiết.

Trong mô-đun 2, chúng ta tiến hành quá trình làm sạch dữ liệu bằng Python trong một notebook. Toàn bộ quá trình này được mô tả chi tiết trong tập tin của mô-đun 2. Kết quả sau khi làm sạch dữ liệu là bộ dữ liệu có các tính chất cần thiết để tiếp tục vào các bước phân tích và xây dựng mô hình học máy.

****

*Hình 2.1. Ảnh minh họa dữ liệu*

### 4.2.1 Tạo dataframe từ dữ liệu thô với các header tương ứng.

Trước tiên, ta cần phải xóa các cột có dữ liệu trùng

*# Số chiều của dữ liệu ban đầu*

print(df**.**shape)

df **=** df**.**drop\_duplicates(keep**=**'first')

*# Số chiều của dữ liệu sau khi loại bỏ các dữ liệu trùng*

print(df**.**shape)

Kết quả **:** (43473, 8)

(12484, 8)

Khi cào bộ dữ liệu này, các dữ liệu trống đang được gán giá trị là Null, tuy nhiên ta cần chuyển đổi chúng sang NaN để có thể làm việc được với thư viên pandas

*# Thay đổi các value từ "Null" sang NaN*

df **=** df**.**replace("Null", np**.**nan)

df**.**head(10)

****

### 4.2.2 "Làm sạch" dữ liệu với cột hãng xe

Trong số các hãng xe, có xuất hiện giá trị rỗng, nên ta sẽ loại đi các hàng có hãng xe là rỗng.

Việc bỏ các giá trị rỗng của hãng xe là hết sức cần thiết, bởi đây có thể xem là thuộc tính rất quan trọng, cần phải có khi bán xe máy cũ

df[df['Hang\_xe']**.**isnull()]**.**shape

Kết quả : (11, 8)

Có 11 dòng không có giá trị về hãng xe Rõ ràng, khi xây dựng mô hình dự đoán giá của xe máy, hãng xe là thuộc tính bắt buộc phải có, nên ta sẽ loại những hàng mà hãng xe của nó có giá trị rỗng

*# Xóa đi những dòng mã hãng xe có giá trị rỗng*

df **=** df[df['Hang\_xe']**.**notna()]

*# Kiểm tra xem đã xóa hết chưa*

df[df['Hang\_xe']**.**isnull()]

Trong các giá trị của hãng xe, ta nhận thấy có giá trị "hãngkhác" Ta sẽ xem qua giá trị hãng khác

****

Hình 2.2. Ảnh minh họa các hãng xe

Các giá trị hãng xe khác đều có dòng không rõ, giá trị dao động của chúng có thể tùy thuộc vào các trường dữ liệu khác, và cũng chiếm một số lượng đáng kể, do đó ta sẽ không loại bỏ chúng, mà chỉ thay thế lại tên thành "Hãng khác"

df['Hang\_xe'] **=** df['Hang\_xe']**.**replace(["Hãngkhác"], "Hãng khác")

### 4.2.3. "Làm sạch" dữ liệu với cột năm đăng ký

Với cột năm đăng ký xe, ta có thể tạo thêm một cột mới chứa thông tin về tuổi của xe tính đến thời điểm hiện tại bằng cách lấy năm hiện tại (2023) trừ đi năm đăng ký xe

Nhưng trước đó, ta sẽ kiểm tra dữ liệu của cột năm đăng ký xe

print("Các năm đăng ký: ")

years **=** list(set(df['Nam\_dang\_ky']))

**for** i, year **in** enumerate(years):

print(f'{i **+** 1}. {year}')

Kết quả : Các năm đăng ký:

1. 1987

2. 1994

3. 2015

4. 2016 24. 2004

5. 2008 25. 1999

6. 2000 26. 1981

7. 1997 27. 2020

8. 2022 28. 2007

9. 2010 29. 1989

10. 1992 30. 1995

11. 2011 31. 2006

12. 1983 32. 1988

13. 1986 33. 1984

14. 2014 34. 2009

15. 1990 35. 2012

16. 2018 36. 2003

17. 1993 37. 2021

18. 2019 38. 1991

19. 2001 39. 2005

20. 2013 40. Trước năm 1980

21. 2002 41. 2017

22. 1982 42. 1985

23. 1996 43. 1998

Để quy về tuổi của xe, ca cần chuyển kiểu dữ liệu của cột năm đăng ký xe về kiểu dữ liệu số nguyên int, do đó, ta cần chuẩn hóa dữ liệu 'trướcnăm1980' về thành 1980

df['Nam\_dang\_ky'] **=** df['Nam\_dang\_ky']**.**replace(['trướcnăm1980'], '1980')

years **=** list(set(df['Nam\_dang\_ky']))

**for** i, year **in** enumerate(years):

print(f'{i **+** 1}. {year}')

Kết quả :

1. 1987

2. 1994

3. 2015

4. 2016 24. 2004

5. 2008 25. 1999

6. 2000 26. 1981

7. 1997 27. 2020

8. 2022 28. 2007

9. 2010 29. 1989

10. 1992 30. 1995

11. 2011 31. 2006

12. 1983 32. 1988

13. 1986 33. 1984

14. 2014 34. 2009

15. 1990 35. 2012

16. 2018 36. 2003

17. 1993 37. 2021

18. 2019 38. 1991

19. 2001 39. 2005

20. 2013 40. 1980

21. 2002 41. 2017

22. 1982 42. 1985

23. 1996 43. 1998

Bây giờ, ta sẽ thêm cột tuổi của xe vào ngay sau cột năm đăng ký

THIS\_YEAR **=** 2023

df["Nam\_dang\_ky"] **=** df["Nam\_dang\_ky"]**.**astype(int)

df**.**insert(1,'Tuoi\_xe', THIS\_YEAR **-** df["Nam\_dang\_ky"])

df**.**head(10)

****

Hình 2.3. Ảnh mô tả năm đăng ký xe

### 4.2.4. "Làm sạch" dữ liệu với cột tình trạng xe

Trước tiên, ta sẽ xem xem cột tình trạng xe có bao nhiêu loại dữ liệu

df["Tinh\_trang\_xe"]**.**value\_counts()

Kết quả :

Đãsửdụng 12378

Mới 95

Name: Tinh\_trang\_xe, dtype: int64

Ta sẽ xem qua về các dữ liệu những xe có tình trạng sử dụng là mới

****

Hình 2.4. Ảnh mô tả tình trang xe mới

Ta sẽ thay đổi dữ liệu từ 'Đãsửdụng' về thành 'Đã sử dụng'

df['Tinh\_trang\_xe'] **=** df['Tinh\_trang\_xe']**.**replace(['Đãsửdụng' ], 'Đã sử dụng')

df**.**head(10)

****

Hình 2.5. Ảnh mô tả tình trang xe đã qua sử dụng

### 4.2.5. "Làm sạch" dữ liệu với cột dung tích xe

Tương tự như tình trạng xe, trước tiên, ta cũng sẽ xem xem cột dung tích xe có bao nhiêu loại dữ liệu

motor\_caps **=** list(set(df['Dung\_tich\_xe']))

print("Phân loại dung tích xe:")

**for** i, cap **in** enumerate(motor\_caps):

print(f"Loại {i **+** 1}: {cap}")  
Kết quả :

Phân loại dung tích xe:

Loại 1: 100-175cc

Loại 2: Dưới50cc

Loại 3: nan

Loại 4: Khôngbiếtrõ

Loại 5: 50-100cc

Loại 6: Trên175cc

df["Dung\_tich\_xe"]**.**value\_counts()

Kết quả :

100-175cc 7607

50-100cc 628

Trên175cc 400

Dưới50cc 209

Khôngbiếtrõ 30

Name: Dung\_tich\_xe, dtype: int64

Ta sẽ tách các cụm từ ra thành các từ rõ ràng như 100-175 cc, Trên 175 cc, 50 - 100 cc, Dưới 50 cc, Không biết rõ

df['Dung\_tich\_xe'] **=** df['Dung\_tich\_xe']**.**replace(['100-175cc'], '100 - 175 cc')

df['Dung\_tich\_xe'] **=** df['Dung\_tich\_xe']**.**replace(['Trên175cc'], 'Trên 175 cc')

df['Dung\_tich\_xe'] **=** df['Dung\_tich\_xe']**.**replace(['50-100cc'], '50 - 100 cc')

df['Dung\_tich\_xe'] **=** df['Dung\_tich\_xe']**.**replace(['Dưới50cc'], 'Dưới 50 cc')

df['Dung\_tich\_xe'] **=** df['Dung\_tich\_xe']**.**replace(['Khôngbiếtrõ'], 'Không rõ')

df**.**head(10)

****

Hình 2.6. Ảnh mô tả dung tích xe 100-175cc

Đếm dữ liệu rỗng của dung tích xe

df[df['Dung\_tich\_xe']**.**isnull()]**.**shape

Kết quả : (3599, 9)

Do số lượng dữ liệu rỗng của dung tích xe là quá lớn, nên ta sẽ thay thế giá trị rỗng bằng "Không rõ" đối với các giá trị rỗng của dung tích xe

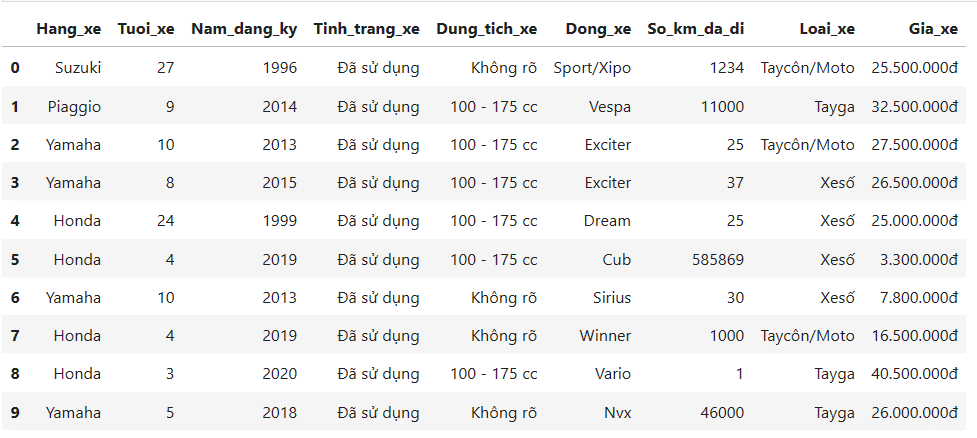
df['Dung\_tich\_xe'] **=** df['Dung\_tich\_xe']**.**replace([np**.**nan], 'Không rõ')

*# Kiểm tra lại số lượng dung tích xe có giá trị rỗng sau khi đã được*

*# thay bằng giá trị "Không rõ"*

df[df['Dung\_tich\_xe']**.**isnull()]**.**shape

df**.**head(10)



Hình 2.7. Ảnh mô tả dung tích xe không rõ

### 4.2.6. "Làm sạch" cột dữ liệu dòng xe

Cũng hoàn toàn giống với các trường dữ liệu khác, ta sẽ xem xem có bao nhiêu dòng xe khác nhau trong bộ dữ liệu này

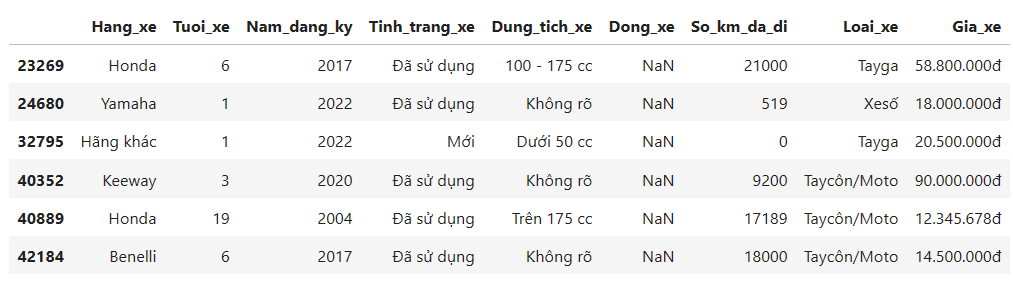
models **=** list(set(df["Dong\_xe"]))

print("Các dòng xe:")

**for** i, model **in** enumerate(models):

print(f"Dòng xe {i **+** 1}: {model}")

Do số lượng dòng xe khá lớn (196 dòng xe), nên ta sẽ chạy lệnh để kiểm tra có giá trị rỗng trong dòng xe không

****

Hình 2.8. Ảnh mô tả dòng xe

Cũng tương tự với hãng xe, dòng xe là một trong những thuộc tính quan trọng cần phải biết, cùng với đó, số lượng dữ liệu rỗng cũng không phải quá lớn (Quan sát bằng mắt thường ta đếm được chỉ có lượng nhỏ) nên ta sẽ loại bỏ đi những hàng này

df **=** df[df["Dong\_xe"]**.**notna()]

*# Đồng thời cũng thay đổi lại khoảng cách từ*

df['Dong\_xe'] **=** df['Dong\_xe']**.**replace(['Dòngkhác'], "Dòng khác")

df**.**shape

Kết quả : (12467, 9)

### 4.2.7. "Làm sạch" cột dữ liệu về số km đã đi

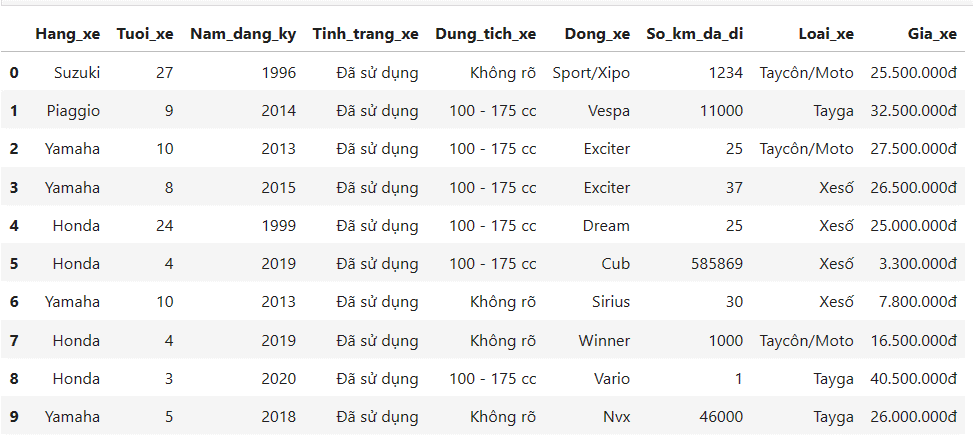
Ta sẽ ép kiểu các giá trị của số km đã đi sang kiểu int, đồng thời, loại đi các giá trị rỗng, do số km là giá trị định lượng nên không thể tìm được giá trị thay thế nên cần loại bỏ.

df **=** df[df['So\_km\_da\_di']**.**notna()]

df['So\_km\_da\_di'] **=** df['So\_km\_da\_di']**.**astype(int)

df['So\_km\_da\_di']**.**dtypes

df**.**head(10)



Hình 2.9. Ảnh mô tả số Km đã đi

### 4.2.8. "Làm sạch" dữ liệu về loại xe

Với dữ liệu này, ta sẽ xem các loại xe và tìm các giá trị rỗng, tùy thuộc vào giá trị các loại xe mà ta sẽ quyết định cách thức xử lý giá trị rỗng

*# Ta đếm số loại xe*

df["Loai\_xe"]

kết quả :

0 Taycôn/Moto

1 Tayga

2 Taycôn/Moto

3 Xesố

4 Xesố

...

43319 Xesố

43370 Xesố

43386 Xesố

43471 Taycôn/Moto

43472 Tayga

Name: Loai\_xe, Length: 12226, dtype: object

May mắn rằng không tồn tại giá trị rỗng tại Loại xe, cuối cùng ta sẽ phân tách các từ của loại xe

df['Loai\_xe'] **=** df['Loai\_xe']**.**replace(['Tayga'], 'Tay ga')

df['Loai\_xe'] **=** df['Loai\_xe']**.**replace(['Xesố'], 'Xe số')

df['Loai\_xe'] **=** df['Loai\_xe']**.**replace(['Taycôn/Moto'], 'Tay côn')

df**.**head(10)

****

*Hình 2.10. Ảnh mô tả các loại xe*

### 4.2.9. "Làm sạch" dữ liệu về giá

Giá trị của giá xe phải đảm bảo không được phép rỗng, bởi đó là yếu tố quan trọng nhất, là output của mô hình học máy.

Ta sẽ tiến hành kiểm tra xem có giá trị rỗng trong giá xe hay không và sẽ loại bỏ nếu có, đồng thời thay đổi thay đổi về format của các giá trị trong giá xe (Cụ thể, bỏ ký tự đ ở cuối, bỏ các dấu .) sau đó chuyển đổi kiểu dữ liệu cho chúng sang float.

*# Loại bỏ giá trị rỗng ngay nếu có*

df **=** df[df['Gia\_xe']**.**notna()]

*# Format lại và chuyển kiểu dữ liệu*

df["Gia\_xe"] **=** df['Gia\_xe']**.**apply(**lambda** x: x[:**-**1])

df["Gia\_xe"] **=** df['Gia\_xe']**.**apply(**lambda** x: x**.**replace('.', ''))

df['Gia\_xe'] **=** df['Gia\_xe']**.**astype(float)

df**.**head(10)

****

*Hình 2.11. Ảnh mô tả giá xe*

Ta xem lại số chiều của bộ dữ liệu sau khi làm sạch

df**.**shape

Kết quả : (12216, 9)

## 4.3. Phân tích, khám phá dữ liệu

Một trong những bước cần thiết trước khi chúng ta bước vào quá trình xây dựng mô hình học máy dự đoán giá xe máy đó là phân tích - khám phá dữ liệu của chúng ta, đưa ra được những thông tin hữu ích cơ bản từ tập giá trị này. Đồng thời, có thể kết hợp thêm việc làm sạch dữ liệu nếu thấy cần thiết.

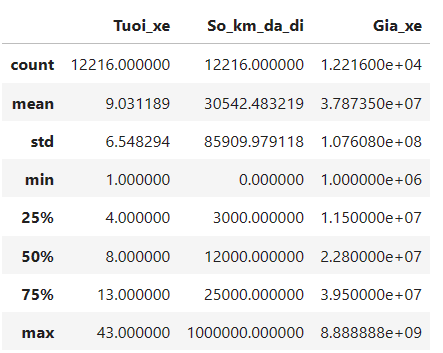
Việc có cái nhìn sơ bộ, đánh giá tổng quan ban đầu về mô hình là hết sức cần thiết và quan trọng, từ đó, giúp chúng ta định hướng ban đầu về các nhân tố tác động đến giá cả từ tập thuộc tính của dữ liệu, giúp đưa ra quyết định xây dựng mô hình một cách tốt hơn.

### 4.3.1 Phân tích dữ liệu định lượng

Dữ liệu định lượng là dữ liệu có trường giá trị là các số thực, thường mang tính liên tục. Các trường dữ liệu định lượng này có thể được đánh giá thông qua các tham số thống kê như giá trị trung bình, giá trị trung vị, tứ phân vị,...

Các trường dữ liệu định lượng có thể thấy trong tập dữ liệu này bao gồm

* Tuoi\_xe
* So\_km\_da\_di
* Gia\_xe



*Hình 2.12. Phân tích và tóm tắt đặc điểm*

Có thể thấy có độ lệch "kha khá" về tham số giá trị trung bình trong trường thuộc tính "so\_km\_da\_di" so với điểm trung vị, có thể điều này do sự xuất hiện của các điểm đột biến của thuộc tính so\_km\_da\_di trong bộ dữ liệu, điều tương tự cũng xuất hiện với "Gia\_xe" (Cần phải lưu ý các điểm này)

Tuy nhiên việc "so\_km\_da\_di" bị tác động nhiều hơn có thể thấy rõ hoàn toàn do tham số về độ lệch chuẩn (std) cũng "lớn một cách bất thường".

Do đó ta sẽ sử dụng biểu đồ hộp để quan sát các điểm đột biến trong bộ dữ liệu đối với cả 3 thuộc tính định tính

# Thể hiện biểu đồ hộp

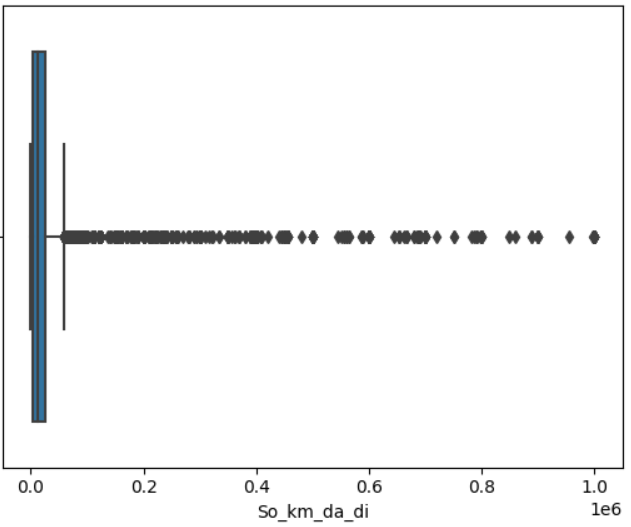
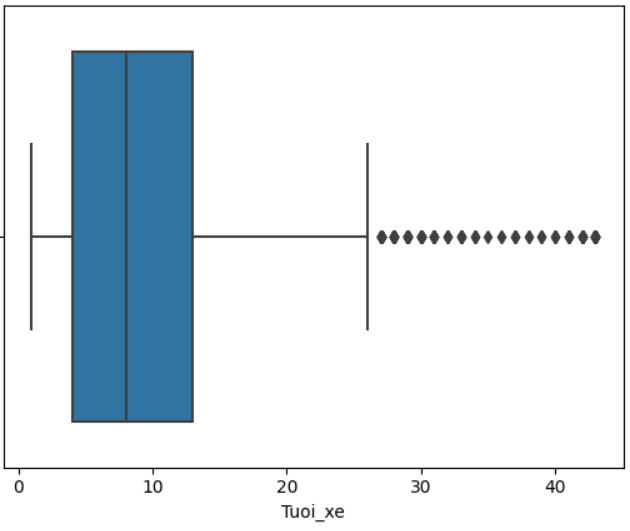
cot\_dinh\_lg = ["Tuoi\_xe", "So\_km\_da\_di", "Gia\_xe"]

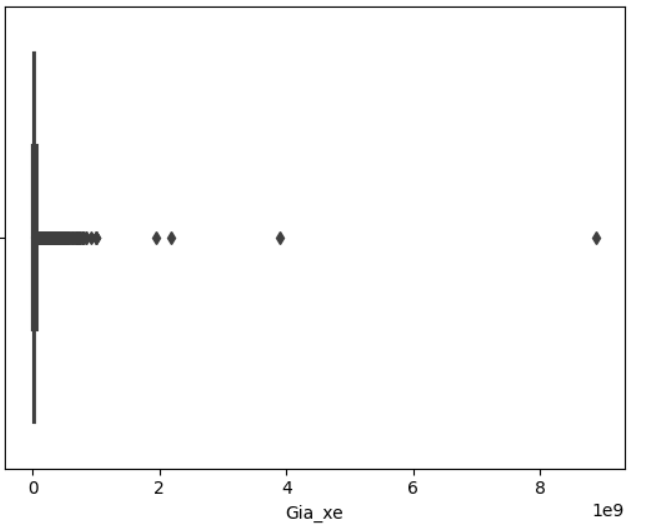
for cot in cot\_dinh\_lg:

    print(f'Biểu đồ hộp của cột "{cot}":')

    sns.boxplot(x=df1[cot])

    plt.show()

****

****

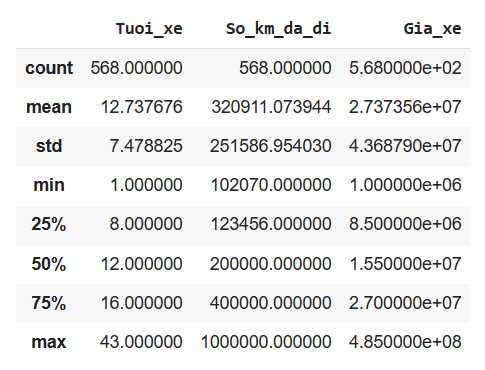
*Hình 2.13. Ảnh thể hiện biểu đồ*

Đúng như chúng ta dự đoán, trong hai thuộc tính "Gia\_xe" và "So\_km\_da\_di", các điểm đột biến quá khác biệt khiến cho chiếc hộp dường như ợ trạng thái "bẹp lại"

Theo [Bài báo này](https://kenh14.vn/moi-nam-nguoi-viet-di-xe-may-trung-binh-7800-km-20200122195837562.chn), số km trung bình xe máy đi một năm là 7800 km, tức để đi được 100000km cần mất khoảng gần 14 năm, chúng ta sẽ xem xét về số tuổi xe trong các quan sát quá bất thường về số km (Số km đã đi lớn hơn 100000km), để quyết định xem có nên loại bỏ chúng không

so\_km\_bat\_thuong = df1[df1["So\_km\_da\_di"] > 1e5]

so\_km\_bat\_thuong[["Tuoi\_xe", "So\_km\_da\_di", "Gia\_xe"]].describe()

****

Có tới 75% xe dưới 16 năm, có nghĩa rằng để đi được con số 100000km có phần phi lý với những dữ liệu này, điều này xảy ra có thể do sự nhập "tự do" dữ liệu của người dùng khi đem bán

Đồng thời, các điểm bất thường cũng không tốt cho mô hình học máy, chúng có thể khiến các tham số của mô hình bị lệch đi đáng kể.

Còn đối với những chiếc xe có giá bất thường (Trên 200 triệu đồng), qua tìm hiểu trên [Chợ tốt](https://xe.chotot.com/mua-ban-xe-may-thanh-pho-thu-duc-tp-ho-chi-minh/99109024.htm#px=SR-similarad-%5BPO-5%5D%5BPL-bottom%5D), các thông tin này thường là các xe bán sỉ, không phải bán lẻ, do đó, đây là dữ liệu bị sai lệch về mặt thông tin.

Do vậy ta sẽ quyết định loại bỏ đi các điểm bất thường này.

GIA\_XE\_MAX = 2e8

SO\_KM\_MAX = 0.7e5

df1 = df1[df1["Gia\_xe"] < GIA\_XE\_MAX]

df1 = df1[df1["So\_km\_da\_di"] < SO\_KM\_MAX]

df1.shape

Kết quả : (11119, 9)

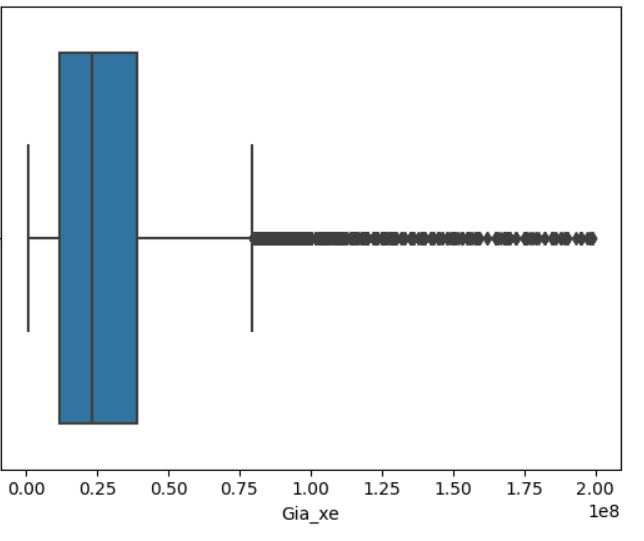
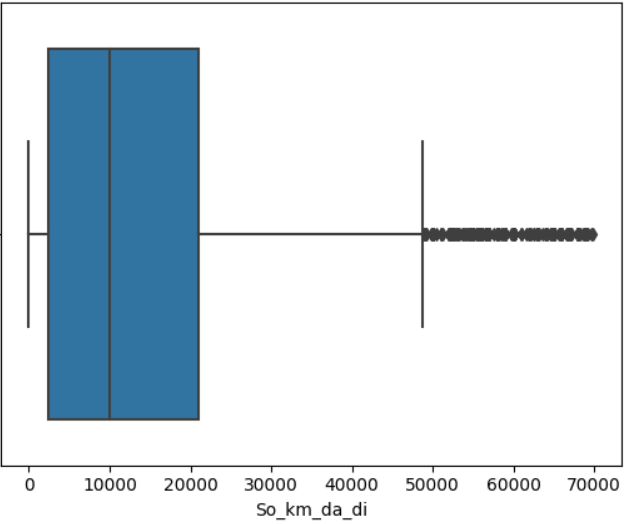
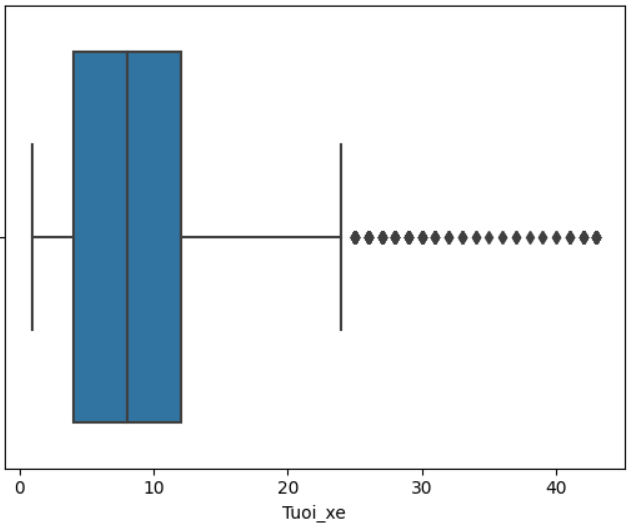
Ta thể hiện lại biểu đồ cột sau khi đã loại bỏ các điểm "siêu đột biến"

for cot in cot\_dinh\_lg:

    print(f'Biểu đồ hộp của cột "{cot}":')

    sns.boxplot(x=df1[cot])

    plt.show()



*Hình 2.14. Thể hiện biểu đồ cột*

Để thể hiện cho phương thức describe() về mức độ phân bố của các dữ liệu định lượng, ta có thể dùng tổ chức đồ tần số

# Thể hiện tổ chức đồ tần số

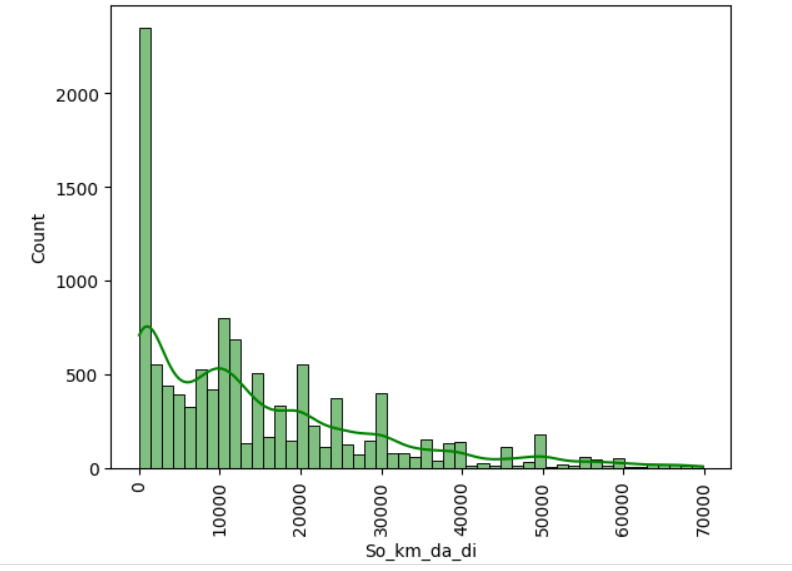
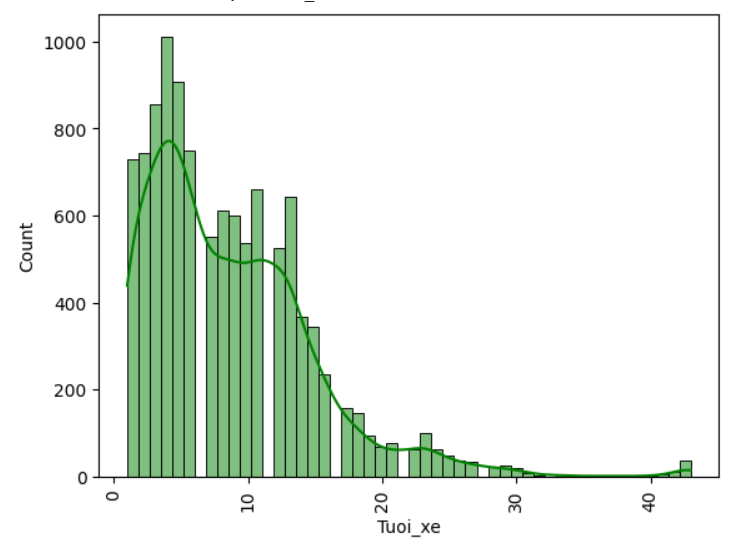
for cot in cot\_dinh\_lg:

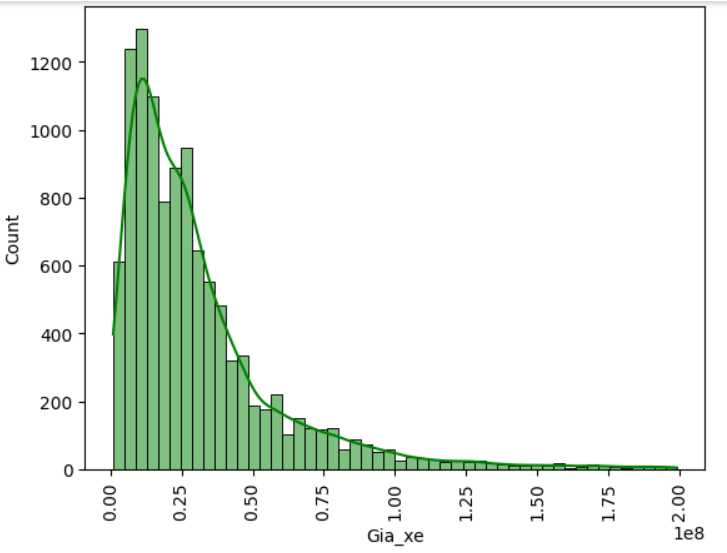
    print(f'Tổ chức đồ tần số của cột "{cot}":')

    sns.histplot(data=df1, x=cot, kde=True, bins=50, color='g')

    plt.xticks(rotation=90)

    plt.show()





*Hình 2.15. Thể hiện tổ chức đồ tần số*

* Cả ba trường dữ liệu đều phân bố chủ yếu ở những mức giá trị bên trái góc nhìn đối xứng của trục tung, như giá xe phân bố chủ yếu từ mức 75 triệu đồng trở xuống, số km đã đi phân bố nhiều ở mức dưới 30000km, còn tuổi xe đa phần ở mức từ 15, 16 tuổi trở xuống (Tức hầu hết các xe được sản xuất từ giai đoạn khoảng năm 2008 trở đi)
* Điều đó đồng nghĩa với việc xu hướng bán các chiếc xe lâu đời, đi được nhiều km là không nhiều, điều này cũng là một điều có thể lưu ý khi chúng ta lựa chọn giao bán chiếc xe(Chú ý rằng xu hướng đăng bán không nhiều không đồng nghĩa với tỉ lệ mua ít).

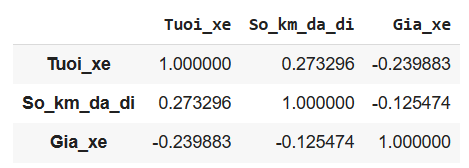
Một điều rất được quan tâm tới các dữ liệu định lượng, là mối tương quan tuyến tính giữa các trường, đặc biệt là giữa trường output "Gia\_xe" với hai trường còn lại

Ta sẽ thử đánh giá về mức độ tương quan tuyến tính

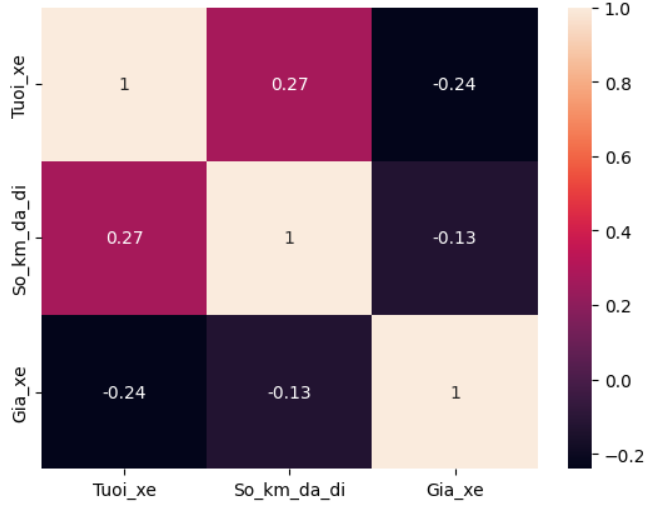
#đánh giá về mức độ tương quan tuyến tính

he\_so\_tg\_quan = df1[["Tuoi\_xe", "So\_km\_da\_di", "Gia\_xe"]].corr()

he\_so\_tg\_quan

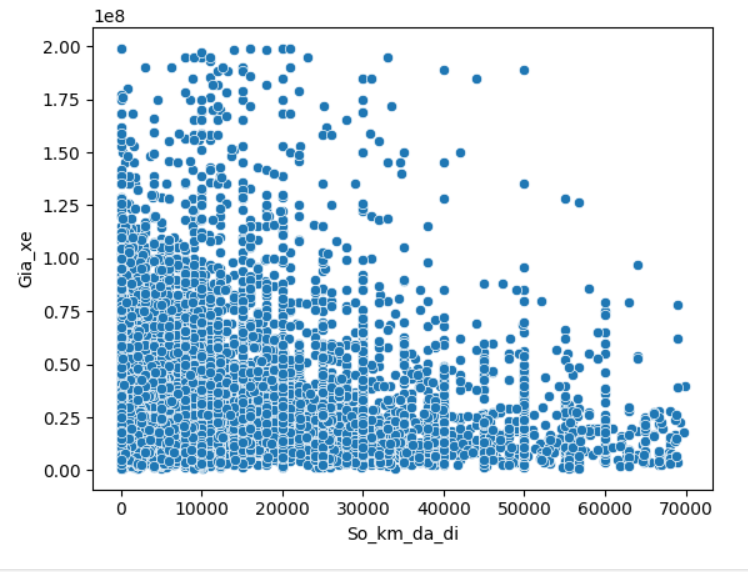
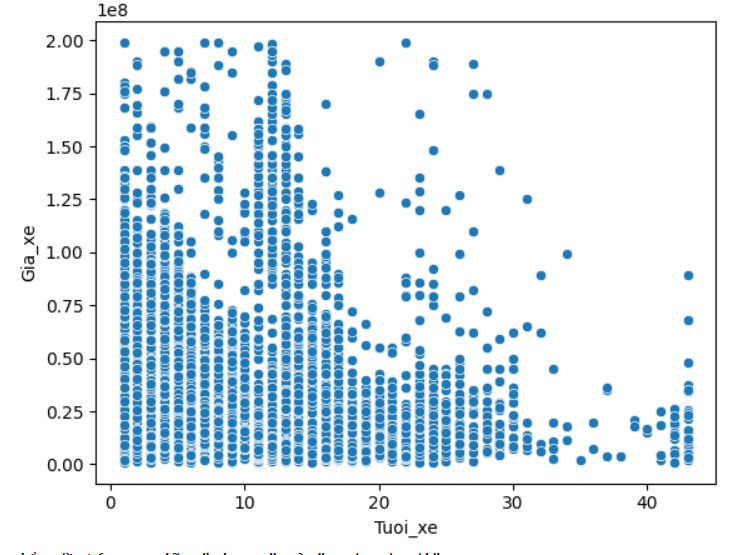
****

Hệ số tương quan giữa "Gia\_xe"với hai trường thuộc tính trên có giá trị tuyệt đối không cao, thể hiện rằng rất rất ít mối quan hệ tuyến tính giữa các đại lượng này.

****

*Hình 2.16. Thể hiện biểu đồ hệ số tương quan*

Hệ số tương quan thể hiện qua biểu đồ phân tán



*Hình 2.17. Thể hiện biểu đồ tán xạ giữa*

Hệ số tương quan giữa "Gia\_xe" với hai trường thuộc tính trên có giá trị tuyệt đối không cao, thể hiện rằng rất rất ít mối quan hệ tuyến tính giữa các đại lượng này.

### 4.3.2. Phân tích dữ liệu định tính

Chúng ta sẽ phân tích - khám phá tiếp tới nhóm dữ liệu còn lại, là nhóm dữ liệu định tính, với các trường thuộc tính của chúng có chứa các giá trị phân loại hữu hạn, là các đối tượng khác nhau

Đối với các hãng xe, ta sẽ đếm số lượng xe của mỗi hãng

# Đổi các hãng xe có số lượng xe dưới 50 thành hãng khác

df1['Hang\_xe'] = df1['Hang\_xe'].apply(\

                lambda x : 'Hãng khác' if len(df1[df1['Hang\_xe']==x]) < 50 else x)

df1["Hang\_xe"].value\_counts()

Kết quả :

Honda 6753

Yamaha 2296

Piaggio 805

Suzuki 438

SYM 306

Hãng khác 191

Kawasaki 61

GPX 54

Benelli 32

Kymco 29

Ducati 26

Detech 16

Brixton 14

Daelim 13

RebelUSA 11

Visitor 9

MVAgusta 9

KTM 9

Halim 8

Bazan 5

Taya 5

Peugeot 5

BMW 4

Nioshima 4

RoyalEnfield 3

Kengo 3

Sachs 3

Victory 2

Hyosung 1

RegalRaptor 1

Sanda 1

Vento 1

HarleyDavidson 1

Name: Hang\_xe, dtype: int64

Các hãng xe có số lượng thấp thường sẽ rất khó đánh giá chính xác khi xây dựng mô hình, nên ta sẽ gom các hãng xe có số lượng ít, cụ thể ta có thể đặt ra ngưỡng số lượng dưới 50 thành giá trị "Hãng khác".

# Đổi các hãng xe có số lượng xe dưới 50 thành hãng khác

df1['Hang\_xe'] = df1['Hang\_xe'].apply(\

                lambda x : 'Hãng khác' if len(df1[df1['Hang\_xe']==x]) < 50 else x)

df1["Hang\_xe"].value\_counts()

Kết quả :

Honda 6753

Yamaha 2296

Piaggio 805

Suzuki 438

Hãng khác 406

SYM 306

Kawasaki 61

GPX 54

Name: Hang\_xe, dtype: int64

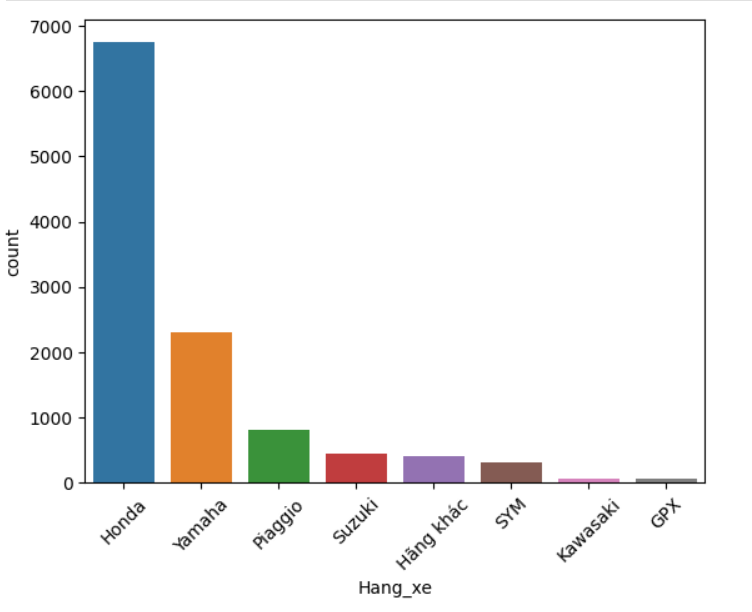
def plot\_bar(column):

  sns.countplot(data=df1, x=column,order=df1[column].value\_counts().index)

  plt.xticks(rotation=45)

  plt.show()

plot\_bar('Hang\_xe')

****

*Hình 2.18. Thể hiện biểu đồ cột hãng xe*

* Biểu đồ cột cho ta thấy sự vượt trội về số lượng của các xe đước bán đến từ Honda, có lẽ đây cũng sẽ là hãng xe được giao bán nhiều nhất trên thị trường.

Cần lưu ý rằng, khi người đăng bán bán xe, nếu trường "Hãng xe" của họ có giá trị Hãng khác, còn Dòng xe của họ lại có giá trị cụ thể, thì rất có thể các dòng xe này đã được có giá trị hãng xe tương ứng ở các dòng dữ liệu này.

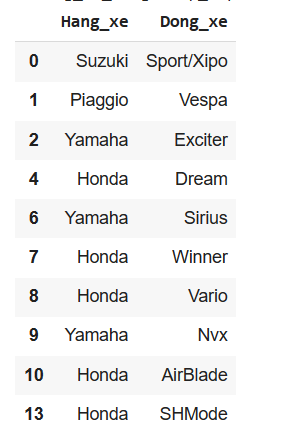
Do đó, tiếp theo đây, ta sẽ thử phân tích xem, liệu có dòng xe nào có dữ liệu hãng xe là "Hãng khác", trong khi đó, nó đã có giá trị cụ thể về hãng ở một dòng dữ liệu khác hay chưa

#phân tích xem, liệu có dòng xe nào có dữ liệu hãng xe là "Hãng khác"

hang\_va\_dong =  df1[['Hang\_xe', 'Dong\_xe']]

hang\_va\_dong.drop\_duplicates(inplace=True)

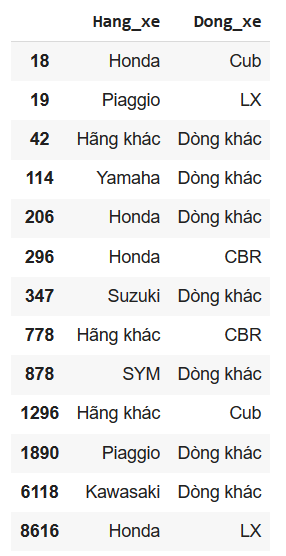
hang\_va\_dong.head(10)



# Xem xem hãng khác có dòng nào của mục "Hãng khác" trùng với các hãng đã biết không

pd.set\_option('display.max\_rows', None)

hang\_va\_dong[hang\_va\_dong['Dong\_xe'].duplicated(keep=False)]



Phát hiện ra rằng dòng xe "Cub", "CBR" là của Honda nhưng bị gán hãng xe là "Hãng khác", do đó ta có thể trả về hãng xe đúng giá trị.

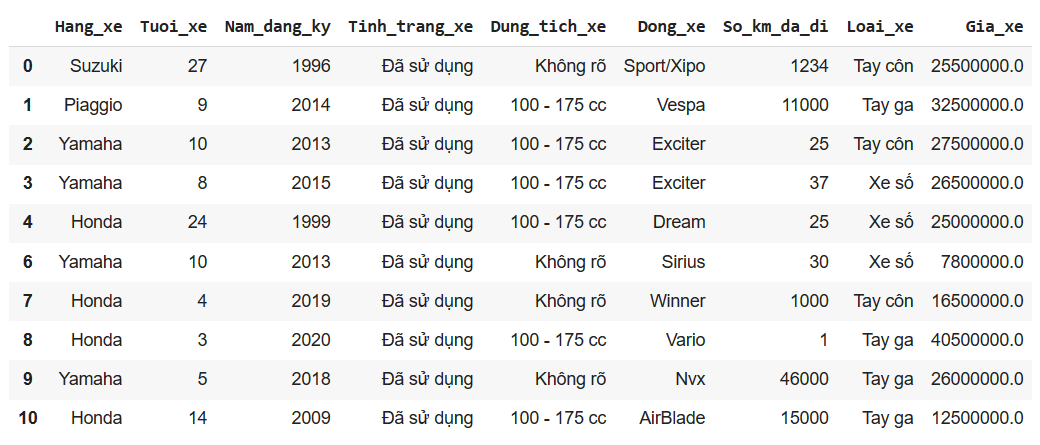
Còn với Vespa LX, giá trị hãng xe chính xác của nó là Piaggio ([Thông tin](https://xe.chotot.com/kinh-nghiem/xe-vespa-cua-hang-nao.html)) nên ta cũng sẽ trả về đúng giá trị.

df1["Hang\_xe"]=np.where(df1["Dong\_xe"]=="Cub",'Honda', df1['Hang\_xe'])

df1["Hang\_xe"]=np.where(df1["Dong\_xe"]=="CBR",'Honda', df1['Hang\_xe'])

df1["Hang\_xe"]=np.where(df1["Dong\_xe"]=="LX",'Piaggio', df1['Hang\_xe'])

df1.head(10)

****

Tiếp theo, ta xem xét giá trung bình của các dòng trong từng hãng qua các biểu đồ

#xem xét giá trung bình của các dòng trong từng hãng qua các biểu đồ

# Dòng xe của các hãng và giá trung bình

gia\_TB=df1.groupby(['Hang\_xe','Dong\_xe'],as\_index=False)['Gia\_xe'].mean)cac\_hang = list(set(df1['Hang\_xe']))

for hang in cac\_hang:

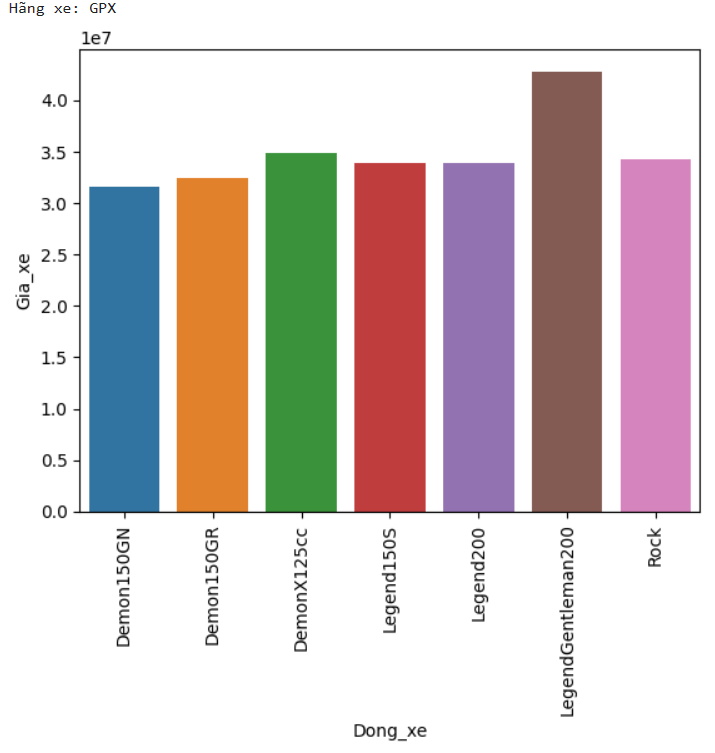
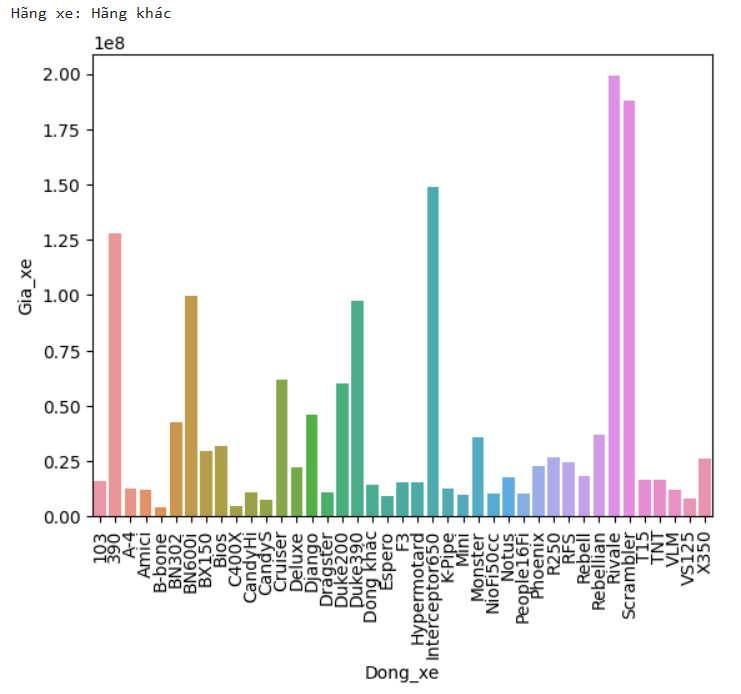
    print(f"Hãng xe: {hang}")

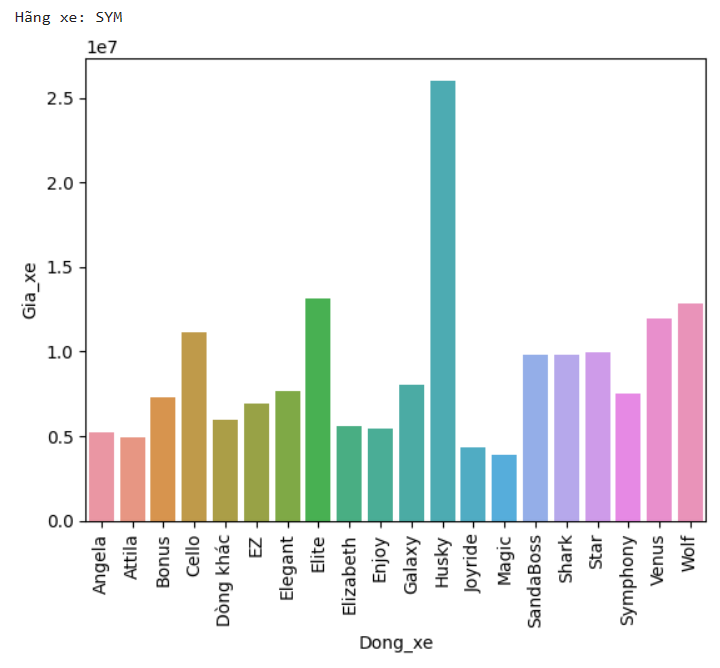
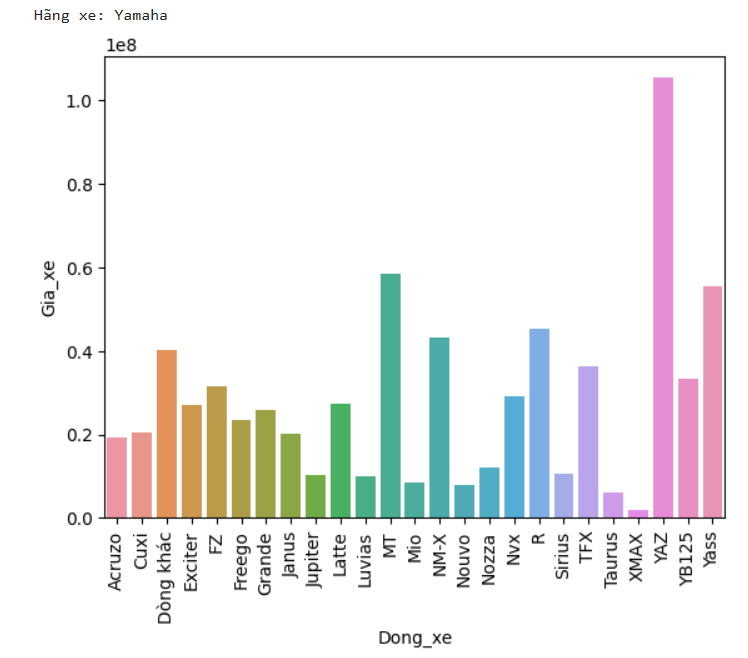
    temp\_df = gia\_TB[['Dong\_xe', 'Gia\_xe']][gia\_TB['Hang\_xe'] == hang]

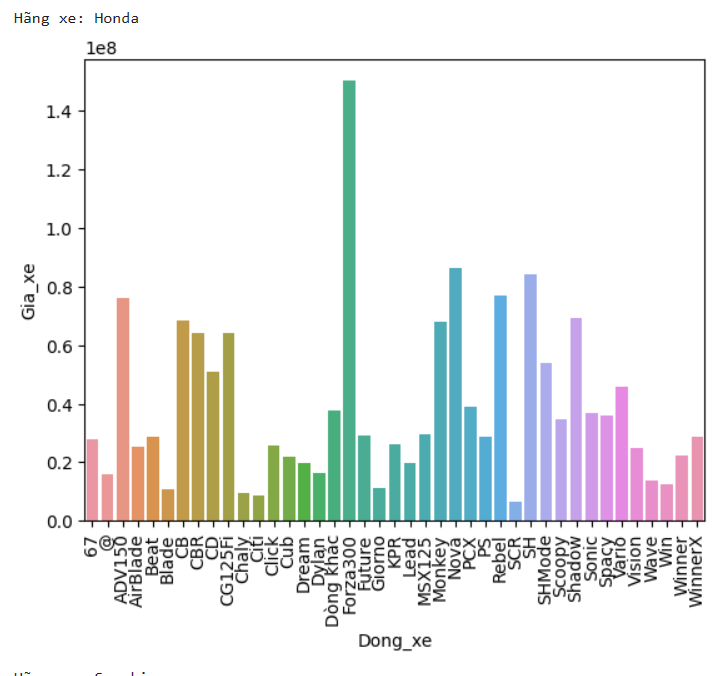
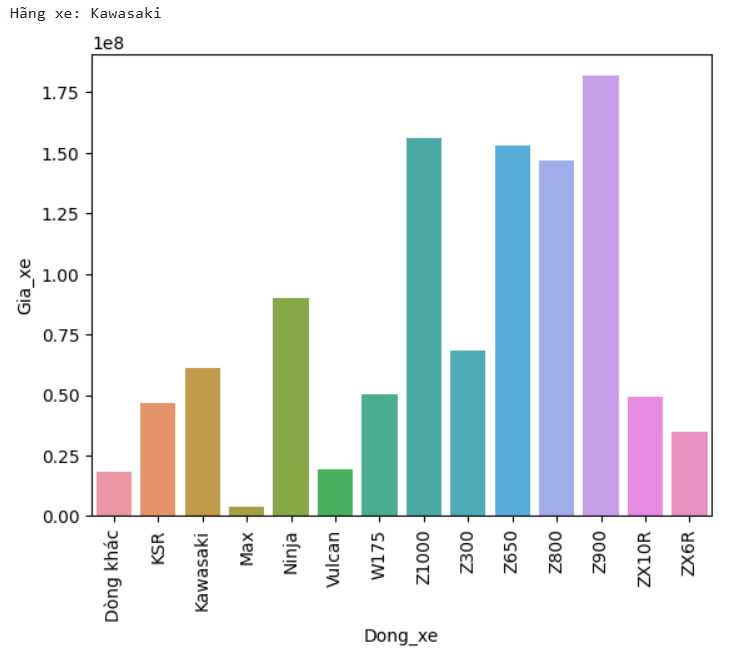
    sns.barplot(data=temp\_df, x="Dong\_xe", y="Gia\_xe")

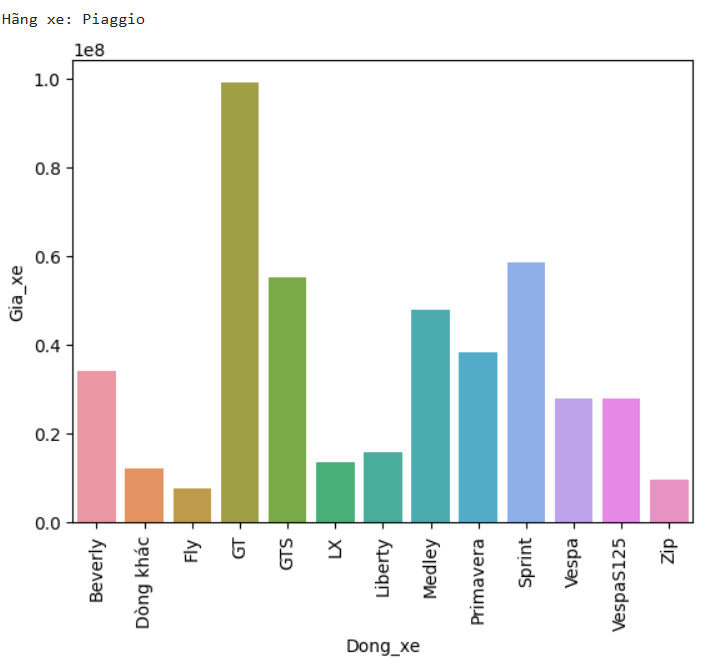
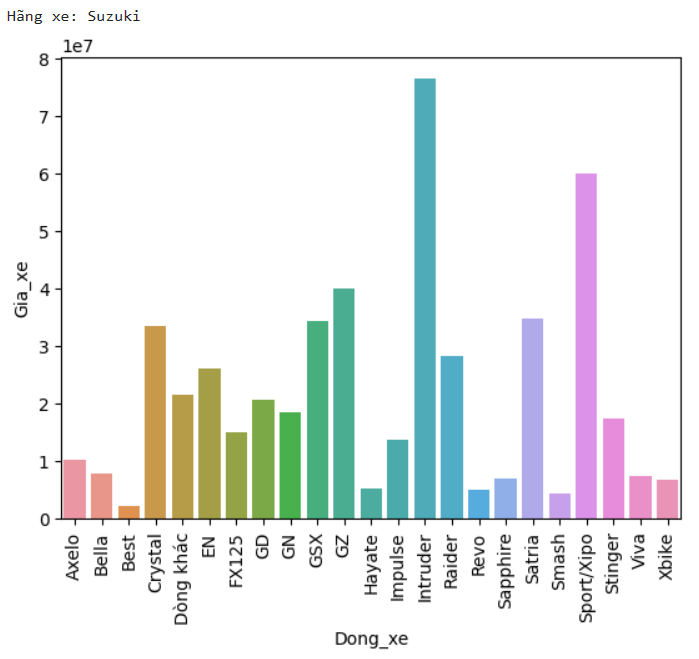
    plt.xticks(rotation=90)

    plt.show()









*Hình 2..19. Biểu đồ giá trị trung bình các dòng xe*

* Hầu hết ở các hãng, các dòng đều có sự chênh lệch khá nhiều về giá trung bình, đồng thời, dòng xe luôn thuộc về một hãng xe, do đó khi xây dựng mô hình dự báo về giá xe, ta có thể loại đi thuộc tính về hãng xe.

Ta xem xét đến các thuộc tính định tính còn lại:

#xem xét đến các thuộc tính định tính còn lại:

# Loại xe, tinh trang xe, dung tich xe

con\_lai = ['Loai\_xe', 'Tinh\_trang\_xe', 'Dung\_tich\_xe']

def show\_value\_counts(col):

    plot\_bar(col)

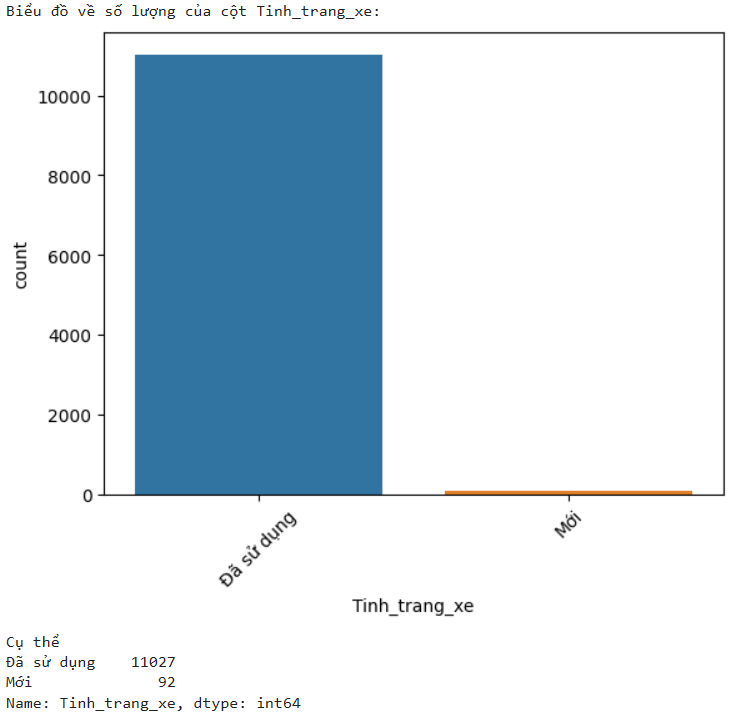
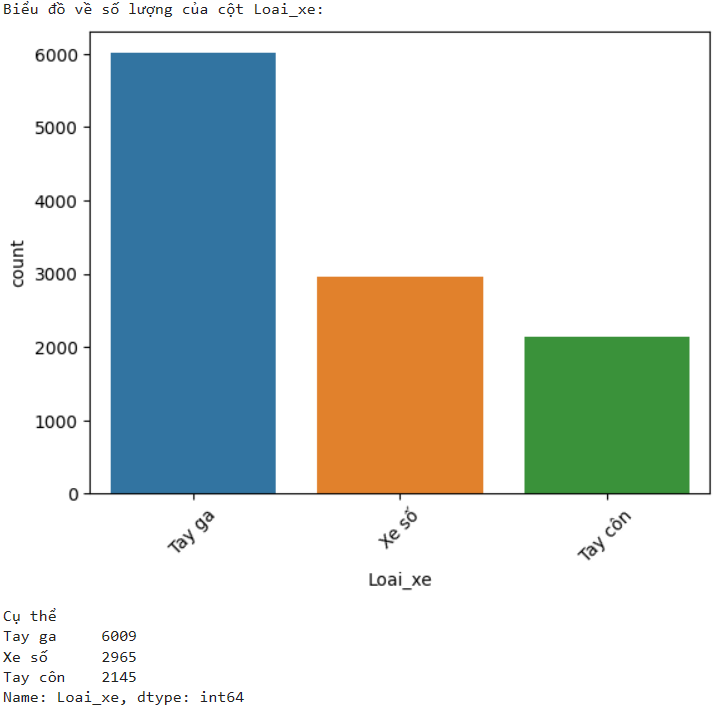
    print('Cụ thể :')

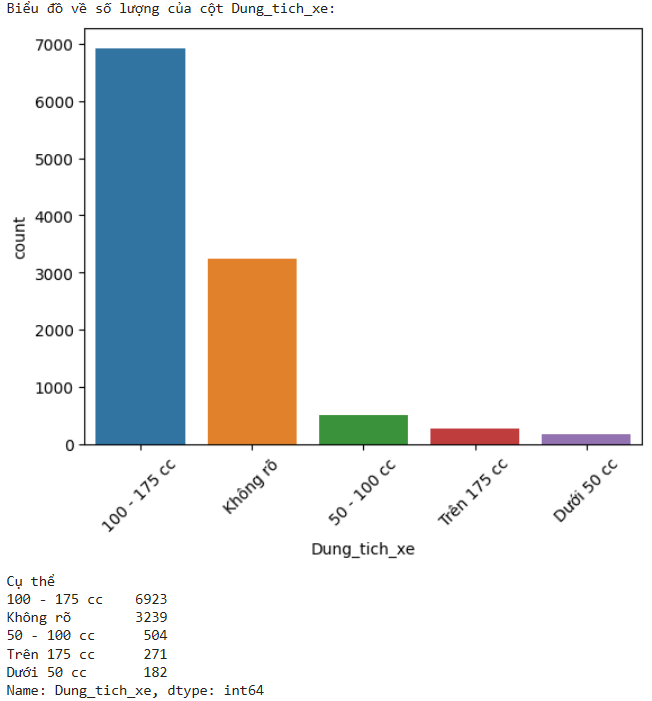
    print(df1[col].value\_counts())

for col in con\_lai:

    print(f"Biểu đồ về số lượng của cột {col}:")

    show\_value\_counts(col)





*Hình 2.20. Biểu đồ về số lượng các thuộc tính*

Về số lượng:

* Xe tay ga được bán nhiều hơn so với xe tay côn hay xe số
* Các xe hầu hết là đã sử dụng, tuy các xe mới là không nhiều, nhưng về số lượng, nó có thể đóng góp vào mô hình do cũng có tới 92 dữ liệu về các xe mới
* Các xe được bán chủ yếu là xe có phân khối từ 100-175cc

# Giá xe trung bình theo dung tích và loại xe:

def show\_price(col):

gia\_TB = df1.groupby([col], as\_index=False)['Gia\_xe'].mean()

print(f"Giá xe trung bình theo thuộc tính: {col}")

sns.barplot(data=gia\_TB, x=col, y="Gia\_xe")

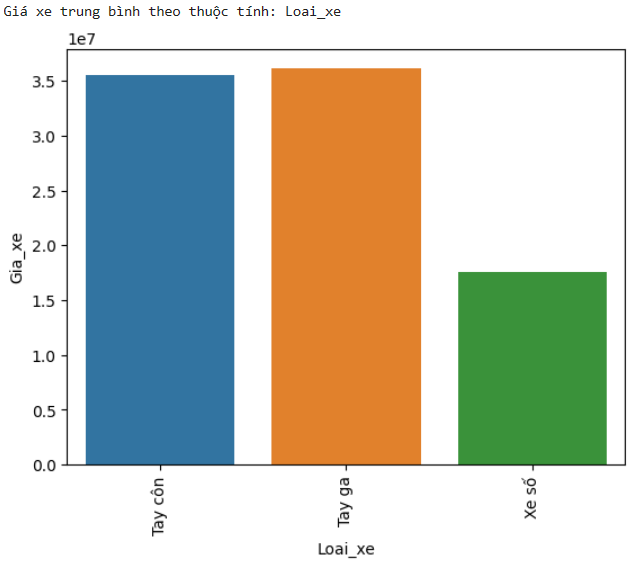
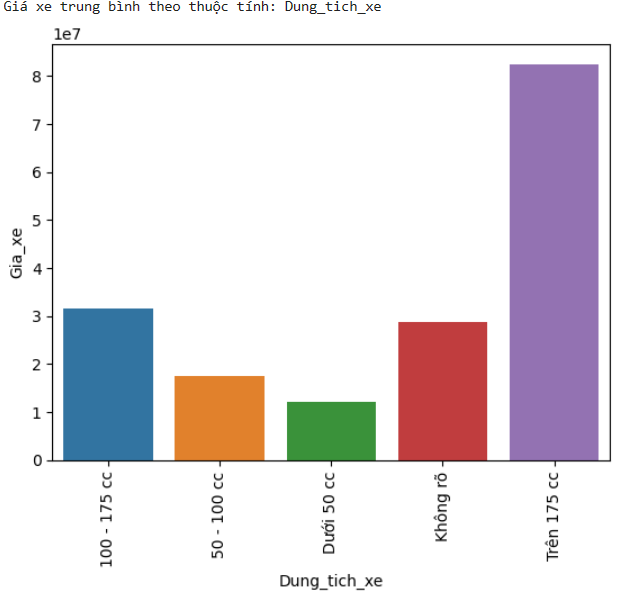
plt.xticks(rotation=90)

plt.show()

show\_price('Dung\_tich\_xe')

show\_price('Loai\_xe')

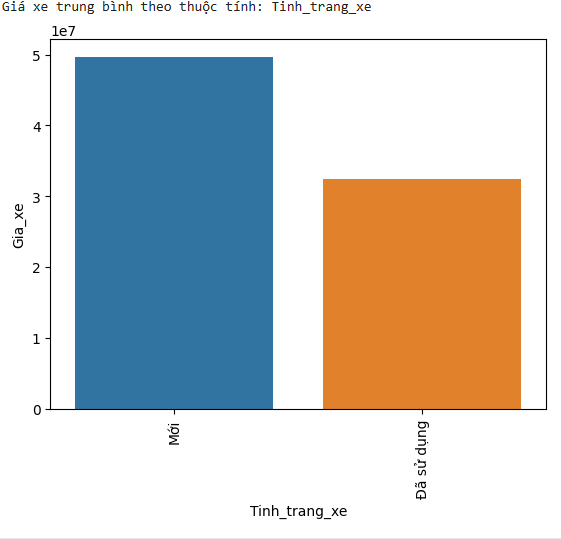
* 1. **Phân tích và khám phá dữ liệu**



*Hình 2.21. Biểu đồ giá xe trung bình theo cột*

Về giá trung bình:

* Xe tay ga và xe tay côn có giá trung bình cao hơn so với xe số
* Xe mới có giá cao hơn so với xe cũ cùng loại
* Xe phân khối càng cao thì giá càng cao (Tỷ lệ thuận)

****

*Hình 2.22. Biểu đồ thể hiện tình trạng xe*

## 4.4. Xây dựng mô hình học máy dự đoán giá xe máy

Sau khi đã trải qua các bước, từ việc chuẩn bị dữ liệu, làm sạch, đưa ra những sự phân tích, đánh giá cơ bản, thì bây giờ chúng ta sẽ đến với bước quan trọng nhất, cũng chính là bước chính thức của dự án này, đó là chúng ta sẽ xây dựng một mô hình dự đoán giá xe máy, có sự đánh giá, lựa chọn ra mô hình tốt nhất trước khi chúng ta đưa vào hệ thống sản phẩm (có giao diện website).

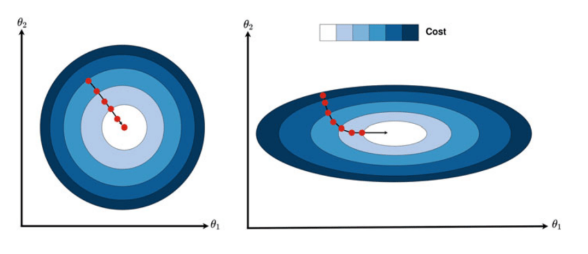
Bài toán lần này là một bài toán về hồi quy, dựa đoán giá trị thực của một thuộc tính, do đó chúng ta cần chọn ra những công cụ, những thuật toán phù hợp để xây dựng tốt mô hình.

Trong dự án này, sẽ sử dụng các công cụ hồi quy đang được hướng dẫn căn bản từ khóa học Machine Learning Specialization của công ty DeepLearning.AI, bao gồm Hồi quy tuyến tính (Linear Regression) (Không chỉ gồm hồi quy tuyến tính mà còn có thể sử dụng tới hồi quy đa thức (Polynomial Regression) - một biến thể của hồi quy tuyến tính, Kỹ thuật hiệu chỉnh thông qua Ridge Regression), Rừng ngẫu nhiên (Random Forest) và Mạng Neural với hàm kích hoạt ở lớp (layer) cuối cùng là hàm tuyến tính (linear-activation).

### 4.4.1 Chuẩn hóa dữ liệu

Trước tiên, để xây dựng được mô hình dữ liệu, chúng ta cần chuẩn hóa dữ liệu. ta sẽ chuẩn hóa các dữ liệu của các thuộc tính đầu vào về khoảng [-1, 1] bằng cách chuẩn hóa theo giá trị trung bình (Mean Normalization) để đảm bảo cho các thuật toán tối ưu được hiệu quả .

Ví dụ minh họa về tác dụng của việc thực hiện chuẩn hóa dữ liệu khi thực hiện thuật toán Gradient descent:

****

*Hình 3.1. Chuẩn hóa dữ liệu*

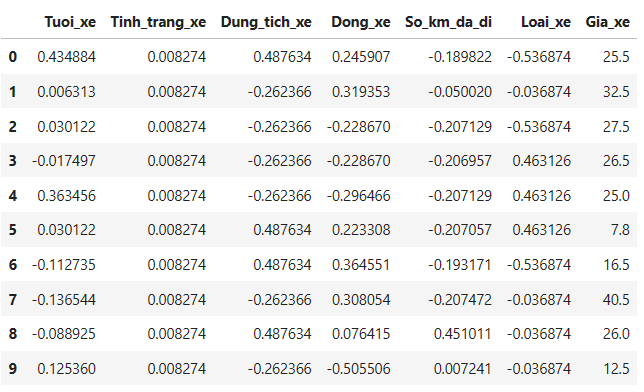
Sau khi thực hiện quá trình chuyển hóa, dữ liệu ban đầu chuyển thành:

Do giá trị giá xe khá lớn, nên ta cũng sẽ thu nhỏ các giá trị lại bằng cách chia chúng cho một triệu (Giá xe tương ứng sẽ chỉ cần nhân lại với một triệu)

# thu nhỏ các giá trị lại bằng cách chia chúng cho một triệu

df2['Gia\_xe'] = df2['Gia\_xe'] / 1e6

df2.head(10)

****

*Hình 3.2 Ảnh mô tả sau khi chuẩn hóa dữ liệu*

### 4.4.2 Tách thành các tập huấn luyện (training) và kiểm thử (testing)

Tiếp theo ta sẽ tách bộ dữ liệu ra thành các tập huấn luyện và kiểm thử mô hình, trong mỗi tập đó sẽ bao gồm các thành phần input X - chứa các thuộc tính ngoại trừ giá cả, output y chứa thuộc tính về giá cả.

Trong tập kiểm thử chúng ta sẽ tách ra thêm thành hai tập "tập validation" và tập kiểm thử "thực sự", tập này có ý nghĩa để chúng ta kiểm thử mô hình và cố gắng tối ưu hóa các tham số, điều này sẽ giúp cho tập kiểm thử "thực sự"của chúng ta là đại diện tốt nhất, mang tính ngẫu nhiên nhất để đánh giá mô hình.

X **=** df**.**drop(['Gia\_xe'], axis**=**1)

y **=** df['Gia\_xe']

X\_train, X\_test, y\_train, y\_test **=** train\_test\_split(X, y, test\_size**=**0.2, random\_state**=**28)

X\_test, X\_val, y\_test, y\_val **=** train\_test\_split(X\_test, y\_test, test\_size**=**0.5, random\_state**=**21)

# Kiểm tra qua số chiều của các tập train test

X\_train.shape, X\_test.shape, X\_val.shape, y\_train.shape, y\_test.shape, y\_val.shape

Kết quả : output

((8895, 6), (1112, 6), (1112, 6), (8895,), (1112,), (1112,))

Bước sau đó sẽ là bước quan trọng nhất, có thể nói như là trái tim của cả dự án: Xây dựng mô hình và đánh giá, lựa chọn ra mô hình tốt nhất có thể. Như đã nói ở phần giới thiệu, trong phần xây dựng mô hình lần này, chúng ta sẽ sử dụng ba phương pháp:

* Hồi quy tuyến tính
* Rừng ngẫu nhiên
* Mạng Neural

Để đánh giá tất cả các mô hình, ta sẽ sử dụng trung một tham số đó là MSE (Mean Squared Error), tham số đo trung bình về độ lệch bình phương giữa các giá trị dự đoán với giá trị thực tế của chúng.

Với phương pháp hồi quy tuyến tính, chúng ta sẽ đưa ra các mô hình sau:

Mô hình đầu tiên của hồi quy tuyến tính:

* Chúng ta sẽ áp dụng phương pháp hồi quy tuyến tính truyền thống,

sử dụng thuật toán Gradient Descent để tìm ra tham số tối ưu cho việc dự đoán giá xe

Mô hình thứ hai của hồi quy tuyến tính:

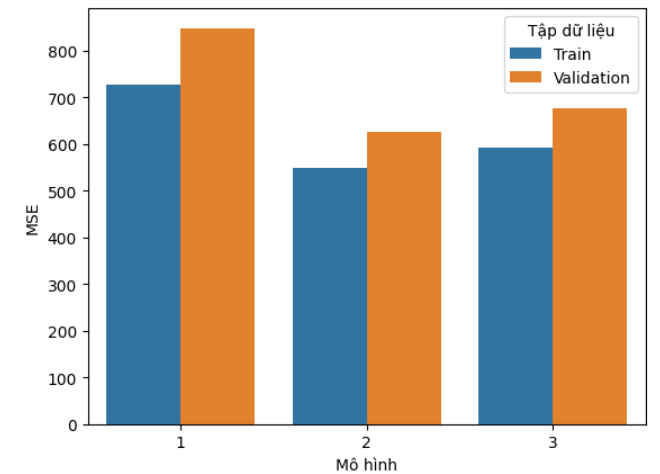
* Chúng ta sẽ thay hồi quy tuyến tính bằng hồi quy đa thức,

với một đa thức bậc cao bất kỳ, việc lựa chọn này sẽ chọn đa thức bậc cao, để phục vụ cho mô hình thứ 4

Mô hình thứ ba của hồi quy tuyến tính:

* Từ mô hình thứ hai, chúng ta sẽ thêm kỹ thuật hiệu chỉnh, hay còn được gọi là hồi quy Ridge để tránh bị Overfitting dữ liệu

Kết quả :



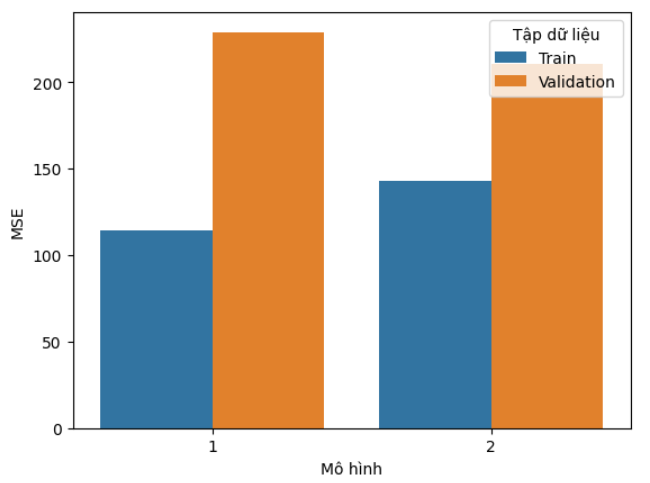
Hình 3.3. Biểu đồ đánh giá thuật toán hồi quy tuyến tính

Với kết quả này, đối với phương pháp hồi quy tuyến tính truyền thống, ta sẽ chọn mô hình thứ hai.

Với phương pháp rừng ngẫu nhiên, chúng ta sẽ sử dụng hai phương pháp:

* Phương pháp rừng ngẫu nhiên truyền thống (Mô hình 1)
* XGboost - một biến thể của rừng ngẫu nhiên (Mô hình 2)

Cách chúng ta đánh giá cũng hoàn toàn tương tự như Hồi quy tuyến tính phía trên.

****

*Hình 3.4. Biểu đồ so sánh mô hình thuật toán*

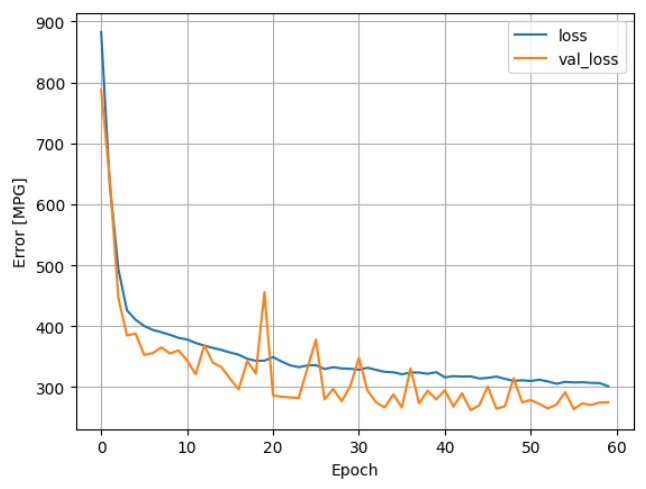
Kết quả đưa ta sự lựa chọn với XGboost (mô hình thứ 2)

Cuối cùng, ta sẽ sử dụng một mạng Neural được xây dựng từ nền tảng Keras của TensorFlow, để tạo ra một mô hình so sánh với hai mô hình tốt nhất của hai phương pháp trên.

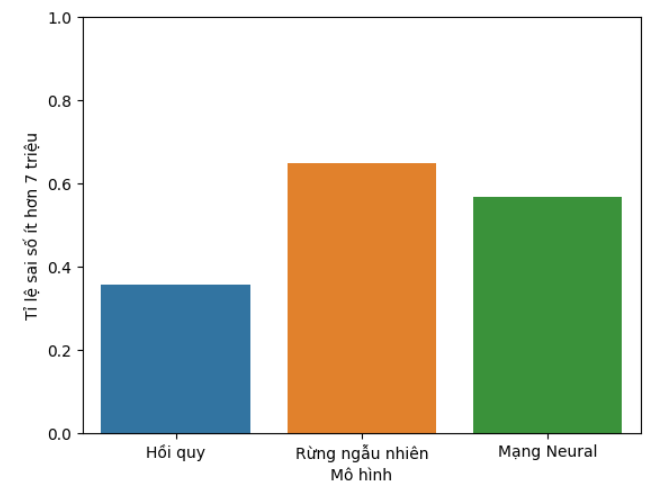
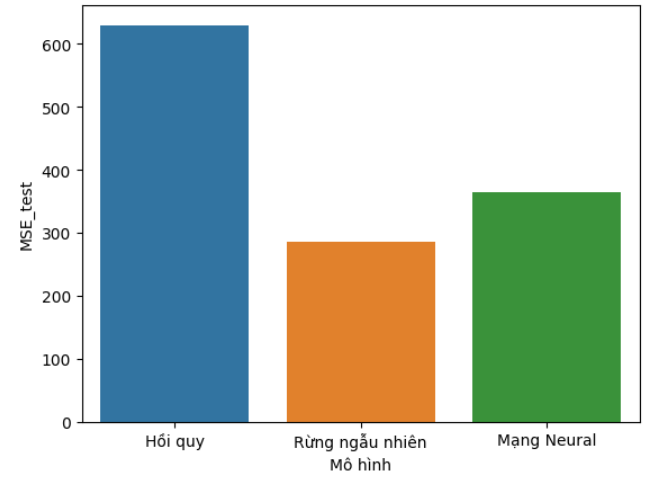
Đây là dữ liệu về MSE của mạng Neural đối với hai tập huấn luyện và validation.

Sau khi đã có 3 "đối thủ nặng ký"nhất, cuối cùng ta sẽ so sánh chúng với tập dữ liệu test thực sự, tập mà chúng ta đã tách ra hoàn toàn riêng biệt với tập validation, là tập dữ liệu "chuẩn"đại diện tốt cho dữ liệu test trong thực tế, do chúng không bị ta tác động như tập validation.

Từ các kết quả trên, so sánh về tỉ lệ giá xe dao động thấp hơn 7 triệu và MSE test, ta quyết định lựa chọn mô hình XGboost là mô hình cuối cùng để

****

*Hình 3.5. Biểu đồ so sánh tỉ lệ giá xe*

****

*Hình 3.6. Biểu đồ so sánh của các thuật toán*

# CHƯƠNG V. KẾT LUẬN

## 1. Kết quả đạt được

Sau một quá trình thực hiện dự án, nhóm chúng tôi đã hoàn thành dự án. Mặc dù đã hoàn thành, nhưng nhận thấy dự án vẫn còn những điểm hạn chế cần phải cải thiện như:

* Giá trị MSE vẫn không thực sự quá nhỏ ở mô hình XGboost (Mô hình tốt nhất đã được chọn).
* Chưa thể phân tích hết được nhiều mặt khác trong quá trình Phân tích - Khám phá dữ liệu (Mô-đun 3).
* Chưa lấy được trường thuộc tính về thành phố (Nơi xe được bán) - một thuộc tính có thể cũng rất hữu ích.

Tuy vậy, hệ thống hoạt động cũng khá tốt, với nhiều trường hợp đối với các dòng xe phổ biến hiện nay như Dream, Wave, Sirius, Vespa, Vision, SH, Air-Blade,... hệ thống đưa ra được một mức giá cũng rất hợp lý theo mặt bằng chung, giúp cho người dùng có thể dựa vào đó để tham khảo về mức giá tùy cho mục đích của mình.

## 2. Hướng phát triển:

* Thêm mới nguồn dữ liệu, có thể bổ sung thêm nguồn dữ liệu đa dạng hơn về các loại xe (Nguồn dữ liệu ban đầu thể hiện sự chiếm ưu thế của Honda).
* Phân tích sâu hơn về các khía cạnh mới (Ví dụ như sự phụ thuộc về giá giữa các tổ hợp thuộc tính thay vì chỉ một loại thuộc tính).
* Phát triển thêm về mặt sản phẩm, có thể sau khi đưa ra dự đoán, sẽ cho ra một số mẫu xe và mức giá có độ tương đồng với xe mà người dùng nhập vào, để họ có thêm sự tham khảo (Điều này có thể được hỗ trợ bởi thuật toán KNN)

# TÀI LIỆU THAM KHẢO

Link mã nguồn: <https://drive.google.com/drive/folders/1LZ4AdvqjaDWiraXDOs13tpHftVH-N5Gx?usp=sharing>

[1] Thư viện pandas, <https://pandas.pydata.org/>

[2] Thư viện numpy, <https://numpy.org/>

[3] Thư viện matplotlib, <https://matplotlib.org/>

[4] Thư viện scikit-learn, <https://scikit-learn.org/stable/>

[5] Framework tensorflow, <https://www.tensorflow.org/>

[6] Thư viện streamlit, <https://streamlit.io/>

[7] Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, Aurélien Géron

[8] Deep Learning with Python, Fran¸cois Chollet

[9] Practical Statistics for Data Scientists: 50+ Essential Concepts Using R and Python, Andrew Bruce, Peter C. Bruce, Peter Gedeck