BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC THĂNG LONG

BÀI TẬP LỚN

GIÁ XE Ô TÔ CỦA 2 THƯƠNG HIỆU AUDI VÀ CHEVROLET

GIÁO VIÊN HƯỚNG DẪN

TS.NGUYỄN THỊ HUYỀN CHÂU
CN.ĐOÀN TRUNG PHONG

SINH VIÊN THỰC HIỆN

A42979 PHẠM THỊ HOÀI THU
A41262 ĐỖ QUANG THẮNG

NĂM HỌC: 2022-2023

LỜI GIỚI THIỆU

Tài liệu này cung cấp một cái nhìn tổng quan cho các thành viên tham gia phát triển dự án. Bao gồm mục đích, phạm vi, các định nghĩa, thuật ngữ, từ viết tắt, các tham chiếu của dự án này.

Thực tế trong quá trình nghiên cứu, tại mỗi giai đoạn đều xây dựng một tài liệu khác nhau tương ứng với giai đoạn đó. Để giảm thiểu sự phức tạp của các tài liệu trong quá trình giảng dạy, tài liệu này được xây dựng một cách thống nhất trong suốt quá trình thực hiện.

Mục lục

Chương 1

Giới thiệu chung

1.1 Mô tả bài toán

Ngày nay, xe ô tô là một phương tiện không thể thiếu của mỗi người. Đây là sản phẩm phát triển và giúp ích con cho người rất nhiều. Xe ô tô cho phép người sử dụng di chuyển từ điểm này đến điểm khác một cách thuận tiện và nhanh chóng. Người dùng có thể tự do lựa chọn lộ trình và thời gian di chuyển theo ý muốn của mình.

Sự tiện ích và linh hoạt: Xe ô tô cung cấp sự tiện ích và linh hoạt trong việc vận chuyển hàng hóa và người. Người dùng có thể vận chuyển hàng hóa một cách thuận tiện. Đồng thời, xe ô tô cũng cung cấp khả năng vận chuyển nhiều người cùng một lúc, đáp ứng nhu cầu đi lại của người dùng.

Đảm bảo sự độc lập: Sở hữu một chiếc xe ô tô mang lại sự độc lập và tự chủ trong việc di chuyển. Người dùng không phải phụ thuộc vào lịch trình công cộng hoặc người khác để di chuyển từ nơi này đến nơi khác, giúp tăng cường sự tự do và linh hoạt trong cuộc sống hàng ngày.

Tiết kiệm thời gian: Sử dụng xe ô tô giúp tiết kiệm thời gian di chuyển. Người dùng có thể tránh được những khó khăn và chậm trễ có thể xảy ra khi sử dụng các phương tiện công cộng. Việc có một phương tiện cá nhân giúp tiết kiệm thời gian và tăng cường hiệu suất công việc và hoạt động hàng ngày.

1.2 Đối tượng

Đề tài này hướng đến việc nghiên cứu dự đoán giá xe ô tô, do đó đối tượng nghiên cứu chính là các loại xe ô tô thông dụng có trên thị trường.

Đối tượng người đọc của tài liệu là những người quan tâm đến lĩnh vực công nghệ ô tô, thị trường xe hơi, và các phương pháp ước tính giá trị.

Chương 2

Cơ sở lý thuyết

2.1 Khoa học dữ liệu là gì?

Khoa học dữ liệu là sự kết hợp giữa toán học, thống kê, lập trình chuyên biệt, phân tích nâng cao, trí tuệ nhân tạo (AI) và máy học với kiến thức chuyên môn về chủ đề cụ thể để khám phá những thông tin chi tiết hữu ích ẩn chứa trong các tập dữ liệu. Những hiểu biết sâu này có thể được sử dụng để định hướng việc ra quyết định và lập kế hoạch chiến lược.

2.2 Các gói thư viện hỗ trợ

2.2.1 Pandas

Pandas là một mô-đun mạnh mẽ được tối ưu hóa trên Numpy và cung cấp một tập hợp các cấu trúc dữ liệu đặc biệt phù hợp với chuỗi thời gian và phân tích dữ liệu kiểu bảng tính (giống bảng tổng hợp trong Excel).

- Pandas là một thư viện của Python được sử dụng để làm việc với các tập dữ liệu. Nó có các chức năng phân tích, làm sạch, khám phá và khai thác dữ liệu.
- Cho phép phân tích dữ liệu lớn và đưa ra kết luận dựa trên lý thuyết về thống kê. Pandas có thể dọn dẹp các tập dữ liệu lộn xộn, làm cho chúng dễ đọc và trở nên phù hợp. Dữ liệu chuẩn rất quan trọng trong khoa học dữ liệu.

- Gửi các yêu cầu GET, POST, PUT, DELETE và các loại yêu cầu HTTP khác đến một URL cụ thể.
- Truy cập dữ liệu từ API web và trang web bằng cách gửi yêu cầu và nhận lại các phản hồi.
- Quản lý các thông tin liên quan đến yêu cầu như tiêu đề, tham số truy vấn, dữ liệu POST, và cookie.
- Xử lý các phản hồi từ các yêu cầu như truy xuất nội dung HTML hoặc dữ liệu JSON từ trang web hoặc API.

2.2.2 BeautifulSoup (bs4)

Thư viện BeautifulSoup là một công cụ phân tích HTML và XML trong Python. Nó cho phép bạn trích xuất dữ liệu từ các tài liệu HTML/XML một cách dễ dàng và thuận tiện.

Một số tính năng chính của BeautifulSoup bao gồm:

- Phân tích cú pháp HTML/XML và xây dựng cây cấu trúc dữ liệu phù hợp.
- Tìm kiếm và trích xuất dữ liệu từ các phần tử HTML/XML dựa trên các tiêu chí như tên thẻ, lớp, id, v.v.
- Trích xuất nội dung văn bản, thuộc tính, và các phần tử con từ các thẻ HTML/XML.
- Cung cấp các phương pháp hỗ trợ để điều hướng và duyệt qua cây cấu trúc dữ liệu.

2.2.3 HTML

HTML (HyperText Markup Language) là một ngôn ngữ đánh dấu sử dụng để tạo và định dạng các trang web. Nó sử dụng các thẻ và các yếu tố khác để mô tả cấu trúc và nội dung của trang web.

Một số khái niệm và yếu tố quan trọng trong HTML bao gồm:

- Thẻ HTML: Được sử dụng để đánh dấu các phần tử trong trang web và xác định vai trò và tính chất của chúng.

- Thuộc tính: Là thông tin bổ sung được cung cấp cho các thẻ HTML để mô tả và tùy chỉnh các phần tử.
- Cấu trúc và lồng ghép: HTML cho phép bạn tạo cấu trúc phân cấp bằng cách lồng ghép các thẻ HTML bên trong nhau.
- Văn bản và hình ảnh: HTML cho phép bạn chèn và định dạng văn bản, hình ảnh và các phương tiện truyền thông khác vào trang web.

2.2.4 Matplotlib

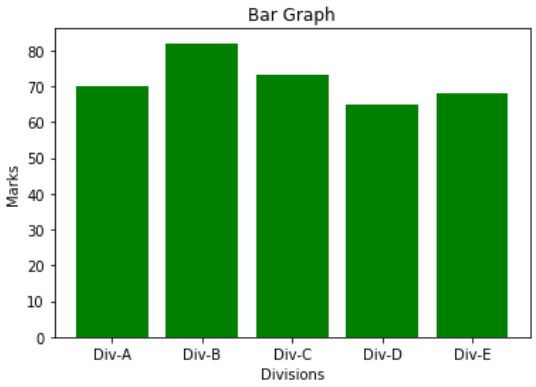
Matplotlib là một thư viện trực quan hóa dữ liệu trong Python. Nó cung cấp các công cụ mạnh mẽ để tạo ra các biểu đồ đa dạng và phong phú để trực quan hóa dữ liệu.

Một số tính năng chính của Matplotlib bao gồm:

- Cung cấp các kiểu biểu đồ phổ biến như biểu đồ đường, biểu đồ cột, biểu đồ hộp, biểu đồ phân tán, biểu đồ vòng tròn, v.v.
- Cho phép tùy chỉnh các thành phần của biểu đồ như tiêu đề, nhãn trục, màu sắc, kích thước, v.v.
- Hỗ trợ tạo ra các biểu đồ phức tạp hơn với nhiều lớp và phần tử trên cùng một trục.
- Tích hợp tốt với NumPy và Pandas, cho phép làm việc dễ dàng với dữ liệu được tổ chức thành mảng hoặc DataFrame.
- Cho phép lưu biểu đồ vào các định dạng hình ảnh như PNG, JPEG, PDF, v.v.

```
divisions = ["Div-A", "Div-B", "Div-C", "Div-D", "Div-E"]
division_average_marks = [70, 82, 73, 65, 68]

plt.bar(divisions, division_average_marks, color='green')
plt.title("Bar Graph")
plt.xlabel("Divisions")
plt.ylabel("Marks")
plt.show()
```



Hình 2.1: Biểu đồ sử dụng Matplotlib

2.2.5 Seaborn

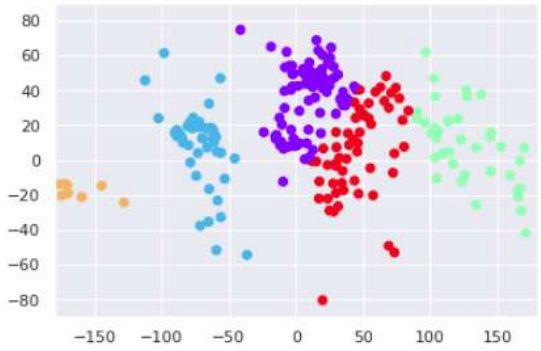
Seaborn là một thư viện trực quan hóa dữ liệu trong Python, được xây dựng dựa trên thư viện Matplotlib. Nó cung cấp giao diện cao cấp và dễ sử dụng để tạo ra các biểu đồ đẹp và chuyên nghiệp.

Một số tính năng chính của Seaborn bao gồm:

- Tích hợp sẵn với Pandas và NumPy, giúp làm việc dễ dàng với dữ liệu được tổ chức thành DataFrame.

- Cung cấp các biểu đồ thống kê trực quan như biểu đồ đường, biểu đồ cột, biểu đồ hộp, biểu đồ phân phối, biểu đồ phân loại, v.v.
- Hỗ trợ việc tạo ra các biểu đồ phức tạp hơn với sự tinh chỉnh và tùy chỉnh màu sắc, kiểu đường, đánh dấu, v.v.
- Tự động điều chỉnh các tham số mặc định để tạo ra các biểu đồ hấp dẫn và chuyên nghiệp.
- Cung cấp chức năng tạo ra các biểu đồ phân tích đa biến và phân tích sự tương quan giữa các biến.

```
In [30]: plt.scatter(data['Longitude'],
                     data['Latitude'],
                     c=data_with_clusters['Cluster'], cmap = 'rainbow')
         plt.xlim(-180,180)
         plt.ylim(-90, 90)
         plt.show()
```



Hình 2.2: Biểu đồ sử dụng Seaborn

Chương 3

Thu thập dữ liệu

3.1 Nguồn dữ liệu

Đường dẫn tới trang web Car.com: "*https://www.cars.com/*"

Khi thu thập dữ liệu, thông tin cần quan tâm là các giá trị có thể ảnh hưởng đến giá ô tô, thương hiệu, tình trạng còn mới hay cũ.
Do lượng dữ liệu tương đối lớn và rộng nên để dữ liệu thu thập có ích cần đặt ra một số tiêu chí:

- Phụ thuộc vào việc các sản phẩm có chứa đầy đủ thông tin hay không, lựa chọn các thông tin mà đa phần các sản phẩm trên web đều có. Chẳng hạn chỉ có một số ít ô tô có thông tin nếu thu thập thì sẽ có nhiều bản ghi bị thiếu thuộc tính này.
- Dữ liệu về thông số cần viết có quy tắc để dễ dàng lấy được giá trị cần thiết.

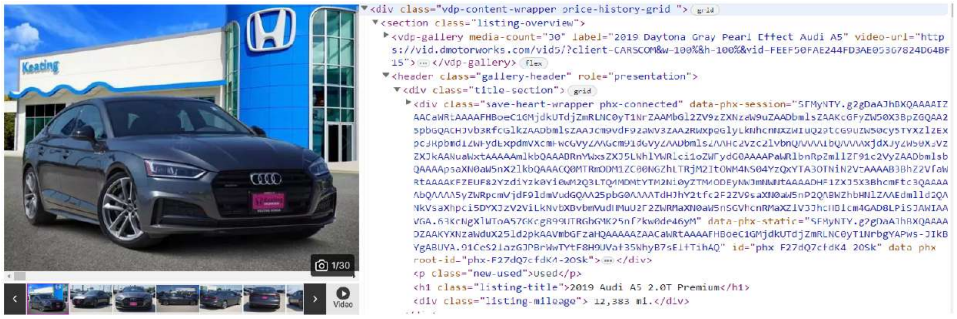
Khi thu thập dữ liệu thô thì thường có nhiều chữ xuất hiện kèm trong dữ liệu cần quan tâm nên kiểu dữ liệu đa phần ở dạng chuỗi. Dữ liệu sau khi làm sạch sẽ được thay đổi kiểu dữ liệu.

Một số trang web chứa thông tin về điện thoại uy tín ở Việt Nam: bonbanh.com, muaban.net,.. Tuy nhiên dữ liệu đa phần chỉ gồm các sản phẩm gần đây và trên thị trường Việt Nam. Sau khi tìm hiểu thì đa phần chỉ khoảng dưới 100 sản phẩm. Vì vậy, tìm kiếm thêm các trang web nước ngoài.

Trong đó, trang web cars.com là một trang web nổi tiếng và được tin cậy trong lĩnh vực đánh giá, so sánh và cung cấp thông tin về sản phẩm ô tô. cars.com được đánh giá là một trang web có uy tín trong ngành công nghiệp ô tô. Web chứa thông tin chi tiết và đầy đủ về ô tô của từng hãng, bao gồm cả sản phẩm đầu tiên của hãng. Vậy nên sử dụng trang web này để thu thập thông tin ô tô. Sau khi thu thập có thể thấy số lượng sản phẩm có khoảng 18000 sản phẩm

3.2 Thực nghiệm thu thập dữ liệu

Phương pháp thu thập thông qua HTML, sử dụng thư viện selenium để truy cập vào trang web và tương tác với các yếu tố cần thu thập.



Hình 3.1: Lấy API thông qua web

Chương 4

Xử lý dữ liệu

4.1 Kiểm tra và đánh giá chất lượng dữ liệu

Số cột: 4

Số hàng: 800

Thông tin về các cột dữ liệu:
Vấn đề của tập dữ liệu: Thông tin về các cột dữ liệu:

- Brand: Thương hiệu
- Name: Tên mẫu sản phẩm
- Status: Tình trạng sản phẩm trên thị trường

Giá xe cao nhất là "$81,075"

In [254]:
```
# Tìm giá trị lớn nhất và nhỏ nhất trong tập dữ liệu
max_price = df['Price'].max()
print("Max price:", max_price)
```

```
Max price: $81,075
```

có 780 hàng trong cột Price được lặp lại trong tổng 800 hàng

In [246]:
```
# đếm giá trị lặp lại
df['Price'].duplicated().sum()
```

Out[246]:
```
780
```

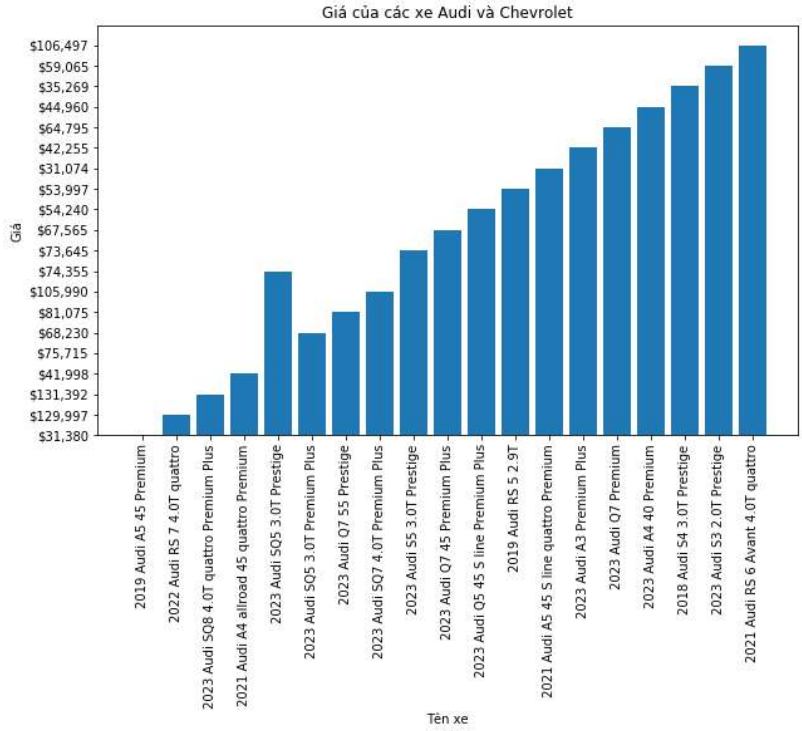
Không có giá trị nào bị missing value

In [236]:
```
# Kiểm tra dữ liệu bị thiếu
df.isnull().sum()
```

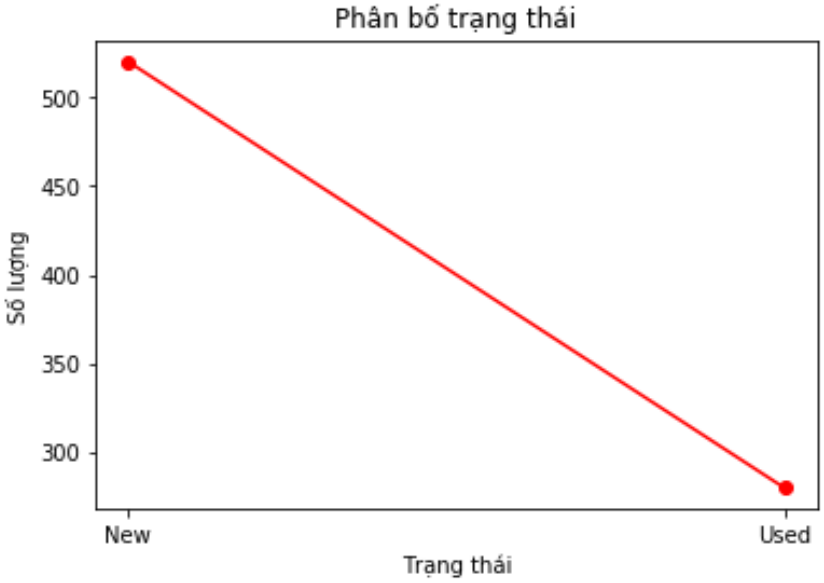
Out[236]:
```
Brand     0
Name      0
Price     0
Status    0
dtype: int64
```

- Số lượng dòng (hàng) của DataFrame là 800, đồng nghĩa với việc chúng ta có 800 mẫu xe được thu thập.
- Cột "Brand" chỉ có 2 giá trị duy nhất là "Audi" và "Chevrolet". Có 400 mẫu xe thuộc thương hiệu "Audi" và 400 mẫu xe thuộc thương hiệu "Chevrolet". Chúng ta có thể suy ra rằng dữ liệu trong DataFrame là phân phối đều giữa hai thương hiệu này.
- Cột "Name" có 19 giá trị duy nhất, đại diện cho 19 mẫu xe khác nhau. Mẫu xe "2023 Audi SQ5 3.0T Prestige" xuất hiện nhiều nhất trong cột này, với 80 lần xuất hiện.
- Cột "Price" có 20 giá trị duy nhất, đại diện cho 20 mức giá khác nhau. Giá "75,715" xuất hiện nhiều nhất trong cột này, với 40 lần xuất hiện.
- Cột "Status" chỉ có 2 giá trị duy nhất là "New" và "Used". Giá trị "New" xuất hiện nhiều nhất trong cột này, với 520 lần xuất hiện.

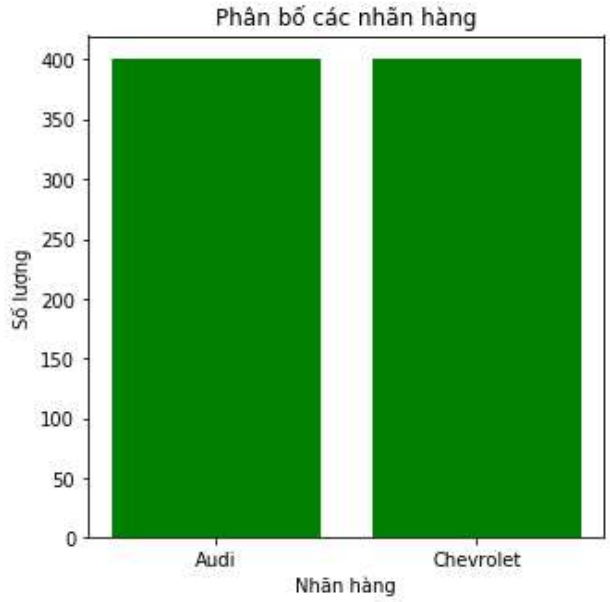
```
]: ## Tạo biểu đồ cột cho giá
   plt.figure(figsize=(10, 6))
   plt.bar(df['Name'], df['Price'])
   plt.xlabel('Tên xe')
   plt.ylabel('Giá')
   plt.title('Giá của các xe Audi và Chevrolet')
   plt.xticks(rotation=90)
   plt.show()
```



```
]: # Biểu đồ đường cho trạng thái
status_counts = df['Status'].value_counts()
plt.plot(status_counts.index, status_counts.values, marker='o',color = 'r')
plt.xlabel('Trạng thái')
plt.ylabel('Số lượng')
plt.title('Phân bố trạng thái')
plt.show()
```



```
# Biểu đồ cột cho nhãn hàng
brand_counts = df['Brand'].value_counts()
plt.figure(figsize=(5,5))
plt.bar(brand_counts.index, brand_counts.values,color = 'g')
plt.xlabel('Nhãn hàng')
plt.ylabel('Số lượng')
plt.title('Phân bố các nhãn hàng')
plt.show()
```



```python
# Tạo một mảng các điểm dữ liệu cho từng thuộc tính
x = df['Name']
y_price = df['Price']
y_brand = df['Brand']
y_status = df['Status']
# Vẽ biểu đồ vùng cho giá

plt.figure(figsize=(19, 6))
plt.stackplot(x, y_price, labels=['Giá'])
plt.xlabel('Tên')
plt.ylabel('Giá')
plt.title('Biểu đồ vùng giá')
plt.xticks(rotation=90)
plt.show()
```