DSCI510 Final Project: What Combination of Characteristics of a Ferrari will Allow me to Purchase at the Best Price?

In this project, I was hoping to solve a fun question using data and see if the results made sense. Using a linear regression, could I determine what aspects of a Ferrari devalued the car and which aspects added value in order to "build the best value car"?

My Data:
I gathered my own data by scraping the website: https://autotrader.com and also by downloading a comparison dataset from https://www.kaggle.com/datasets/hellbuoy/car-price-prediction. My goal was to see how my data compared to a known usable dataset that many people have used successfully for a similar usage goal. I used the BeautifulSoup4 library in Python to scrape the autotrader website for about 2000 cars before cleaning and converting the data to a csv, which was very easy to compare to the downloadable csv from Kaggle. Initially, I had wanted to scrape more car samples and a wider variety, but the web scraping process is very time consuming and even scraping 2000 could take over 30 minutes. Another issue I came across was that not every element on a website is scrapable. Some items were not scrapable using the standard techniques and were locked behind JavaScript objects. Because of this challenge, in my timeframe, I scraped fewer unique values from the web pages than I hoped for. I had to try and be creative using my outside knowledge to create a few extra variables. For example, I know that the vast majority of Ferraris are rear wheel drive (rwd), but there are a few models that are all wheel drive (awd). Using this knowledge, if I could scrape the model name, then I also knew the drivetrain of the car as well and could use it as another feature.

My Analysis:
To determine which characteristics of a car would be costly or cheap, I decided to try using a linear regression using the price as the dependent variable and other characteristics as independent variables. My hope was that if I can break down each individual contribution of a characteristic, then I may be able to determine what I want to search for when I filter my results. For example, I just need any Ferrari, the cheapest possible: what combination of paint, seat color, model, etc will best match my needs?

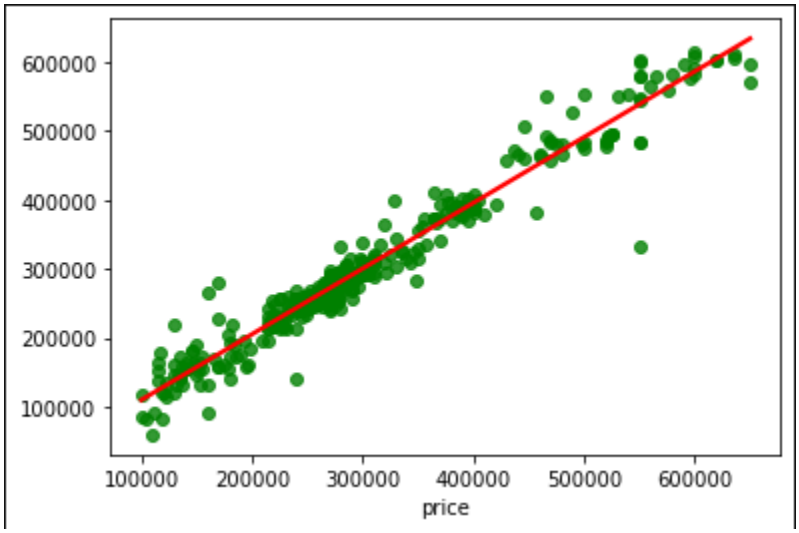
My findings were that the model name is the most important variable. Typically, some older models such as a 360 Modena lowered the value in the model's eyes. However, some "rare" models, such as a 360 Modena Challenge Stradale, with only about 1200 models made, were very expensive and this model name increases value heavily. However, when comparing my data for a brand such as Ferrari to the more regular data for standard car brands from Kaggle, I saw that the linear model values different independent variables more heavily. In the normal car data, it appears that a high value independent variable is actually the brand, which isn't included in the Ferrari dataset because they all have the same brand. For example, 'isBMW == True' contributes a high gain to value. This leads me to believe that the categorical variable of brand or model name is very important to price. Another potentially interesting analysis could be to split up each dataset by brand, and even more by model as well if a person cares more about

more mechanical things instead of name branding. My conclusion is that the most important features from my Ferrari dataset were typically model names, which makes sense because from my outside knowledge of Ferraris, there are some models that are extremely rare and highly exclusive which drive up a high price and there are some models that are produced more heavily as an "entry level Ferrari" with a lower exclusivity and value. Compared to the standard car dataset that contains more "mundane " important variables such as curb weight, fuel type, and number of cylinders.

One other thing to note for the analysis is that initially the model didn't make much sense. I believe that there were too many variables and the model was unable to fit the linear equation well. I used a bucketing technique to reduce the amount of variables involved by grouping some of the rare variables together. For example, if there were some rare color types that were only present in one-2 cars then I would bucket those colors together in a "rare color" category. By doing this, I was able to reduce the number of columns/variables by about 175 and after that, I started obtaining reasonable results.

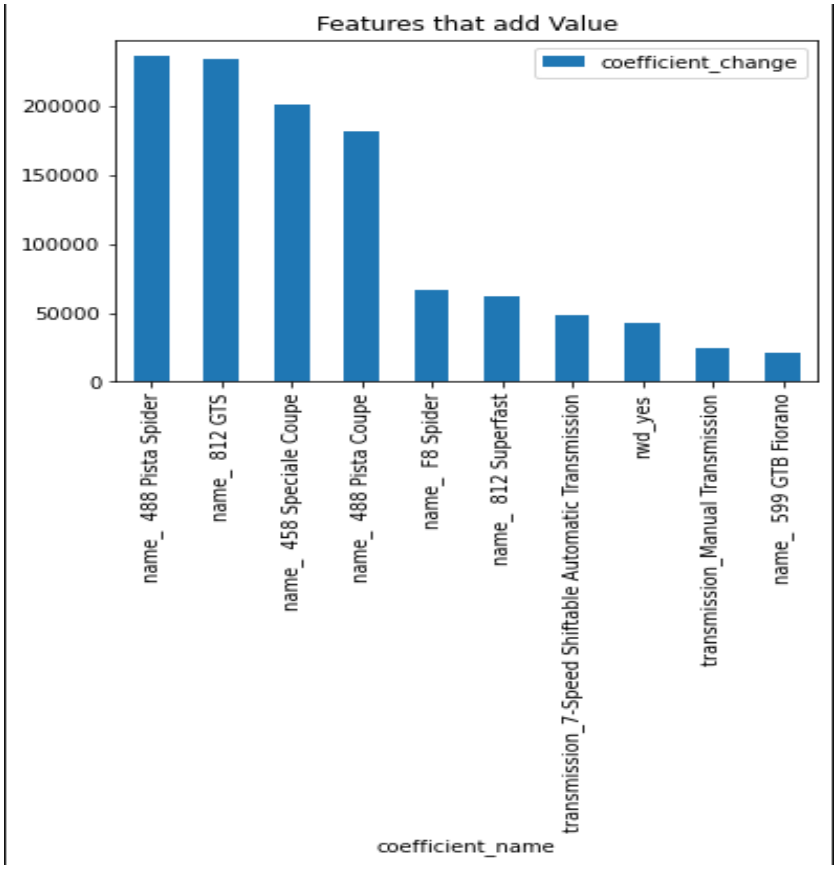
My Visualizations:

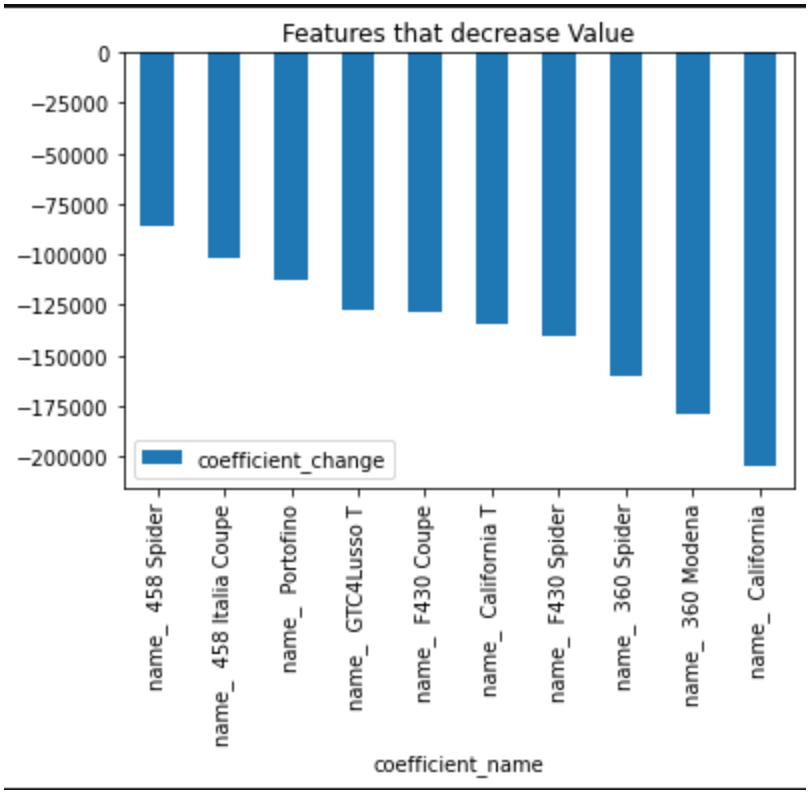
I made a few easy to interpret visualizations displaying the top 10 features that contribute to low value Ferraris and as well for the high value Ferraris. The bar plots are very straight to the point with the x axis as the linear regression coefficient names and the y axis as the value that each of those coefficients has and when combined with the intercept will sum to the car's predicted price. I also included the regression line using Matplotlibs regplot function which shows the predicted test points and the regression line that best follows those plot points. These visuals helped guide my future research into my project's question and I started to see what the key features are in a car relative to its price. This RegPlot simply shows how the computer fits a straight line to existing data points the best it can. The model's predictions should lie on that line, so it's clear that some error is expected, it's a decent estimate though.



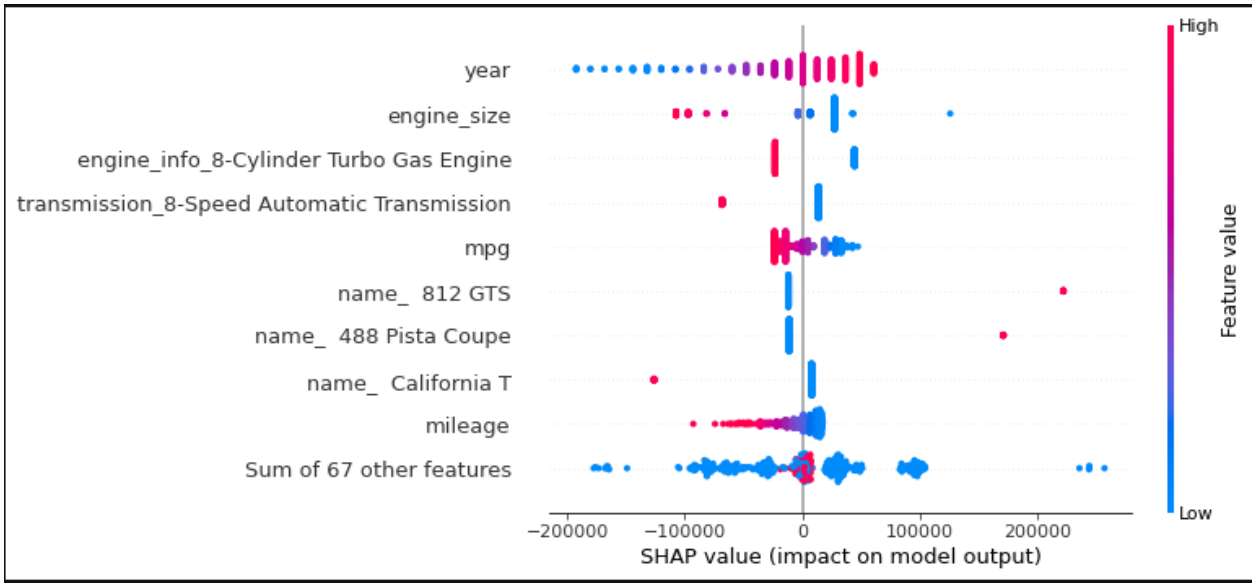
Below are the two bar plots for the Ferrari dataset. Displayed are the top ten features that add and reduce value according to the linear model. 7/10 of the features that add value are model

name variables, along with all of the features that subtract value. This shows how important the model is when it comes to selecting a Ferrari to purchase.

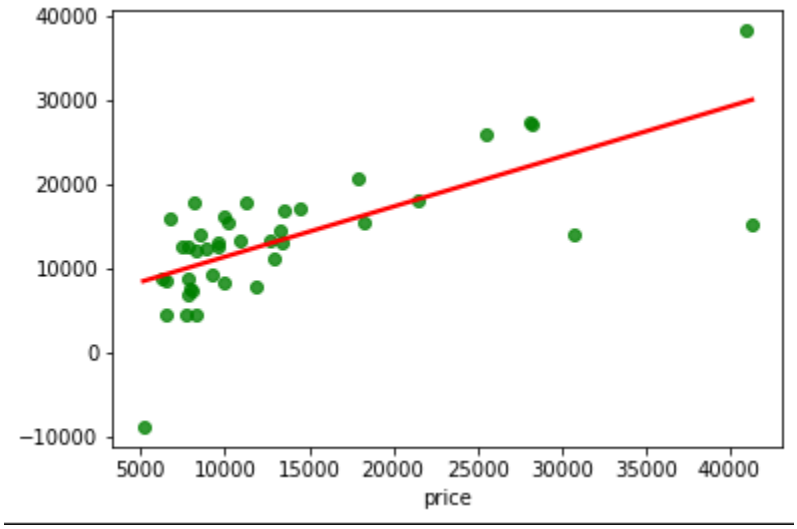




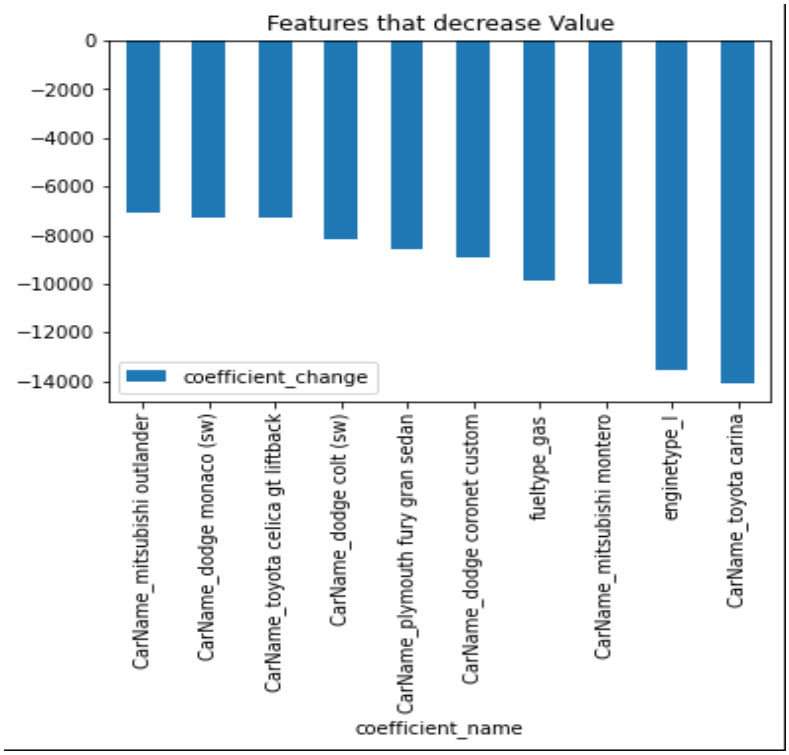
To further explain the linear model, I used SHAP values which try to explain each variable's individual contribution to the model's prediction by emphasizing the importance of each variable. For example, a higher SHAP value for the year variable is a high value indicator of increasing car value.

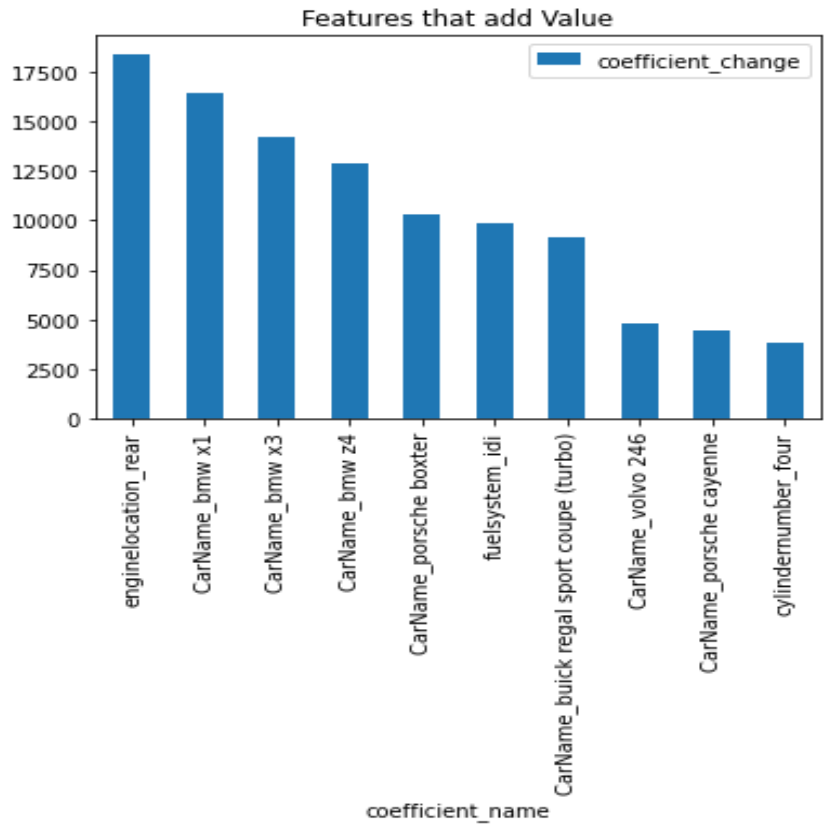


Additionally, I created these plots for my downloaded comparison dataset for standard cars as well. This is another RegPlot, plotting the price prediction for the comparison dataset.

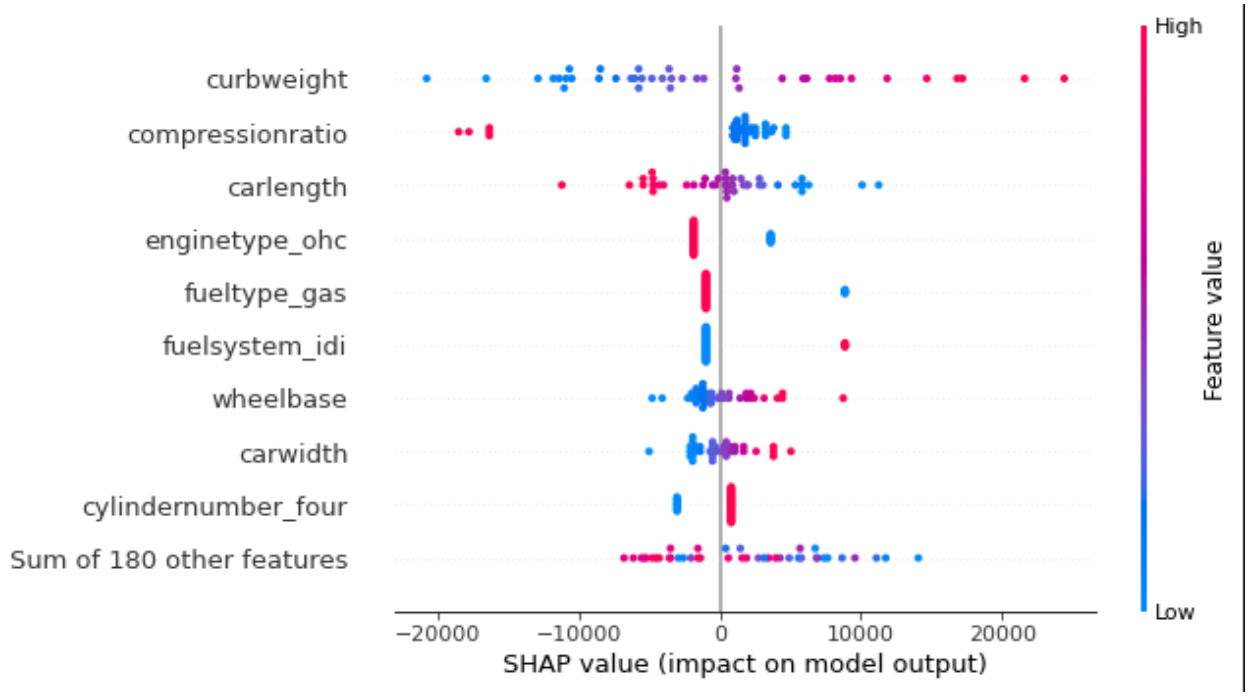


Below are the bar plots for the comparison dataset. Compared to the Ferrari dataset, CarName is also extremely important, but there are a few features that matter significantly as well, such as fuel type and engine location. The addition of these details make me believe that when shopping for a car for day to day life, distinguishing more mechanical features is considered important.





And a SHAP plot for the comparison dataset. It is nice to compare these visuals side by side so we can see what features are important in each model and compare similarities and differences. For example, according to the SHAP importance, some values here seem to behave like the Ferrari model names. Enginetype_ohc, fueltype_gas and fuelsystem_idi all have a small vertical line, so it is probably relevant to those types of vehicles only, in the same way that in the above SHAP plot, name_California_T and name_488_Pista_Coupe behave. This is probably also because they are categorical variables and I created binary dummy variables for them in order to have “numeric” types for the regression to run on.



My Conclusion:

My conclusion after comparing the two independent datasets is that there are some common themes in these very different products. One dataset contains high value luxury commodities and the other has a mix of regular middle value commodities. Keeping in mind the goal of my project was to try and get a good deal on a car, my strategy would follow some of those themes in each of the datasets. Based on my findings in the expensive Ferrari dataset, if I was searching for the best Ferrari for my needs, then I would first narrow my search by model name since that is the most important feature, then down the list of features until I am left with a small subset of choices to buy. In the standard dataset, I would choose a less luxurious car brand, then a lesser known model from there. It looks like the crucial aspect to choosing an economic car in either dataset is selecting the correct model first. The impact of my project is hopefully to teach and inform about what to look for in cars to be a savvy shopper. The cheapest Ferrari I could buy going off my model appears to follow this structure:

Ferrari California
Year as low as possible
Mileage as high as possible
Color
8-Speed Automatic Transmission
8-Cylinder Turbo Gas Engine
Bordeaux Leather Seats
Used
Nero Exterior

I don't know if this car will exist exactly the way it appears on the list and truly be the cheapest Ferrari I can buy, but it gives me a path to follow to find that car in real life.

Future Work:

Given more time, I would love to enhance my web scraping ability to find a way to gather those bits of data hidden behind JavaScript objects. I think there may be ways of doing it with Selenium or other techniques that mimic the way humans interact with websites. Doing this would allow me more unique characteristics that I believe may be valued such as luxury sound systems, additional interior features that need to be specifically noted, and others. I would also enjoy using what I learned from this dataset to pivot to another project like a recommender or something like that. Maybe if I prompt some user input about their preferences or finances, I could recommend a specific car to buy. Sort of like how you would ask an informed friend that is knowledgeable about the topic.