Project 2: Project Report Capstone

Introduction:

Customers, as well as the used car industry, often evaluate used car prices based on the odometer reading and year. However, these features alone are not enough to predict the car's price.

The Goal:

The aim of the project is to predict the depreciation per year of used cars to determine the car's value more accurately. This model would be valuable for value-conscious customers looking to buy new vehicles that will retain their value, for customers buying used cars who want to make sure they are getting a good deal, and for used car dealerships trying to accurately price their vehicles to sell for maximal profit.

Section 1: Data Collection and Wrangling Phase

The final resource for this project was gathered from two sources

1-) Kaggle Used Car Dataset: A dataset of used vehicles containing their age, price, manufacturer, model, drive, type, cylinders, title status, transmission, odometer, condition, and many other features.

2-) Kelly Blue Book (KBB) car data: I scraped kbb.com to get more information about the vehicles to supplement the Kaggle dataset, such as consumer/expert rating, MPG, price when new (necessary to calculate depreciation), and year.

Combining these two datasets wasn't simple, as cars often had different names, and the years didn't necessarily match. I addressed feature name differences manually to merge the dataset, as I only had the price of the newest car models from KBB. I calculated "price when new" by removing the expected inflation over the years from the year the car in the Kaggle dataset was purchased to the year in the KBB dataset.

Section 2: Depreciation Per Year

Used cars with the same age, odometer reading, and clean title status may not have the same depreciation value because cars differ in price retention. The parameter depreciation per year of each vehicle minimizes the errors coming from predicted prices based on the car's age and mileage.

The four features retrieved from the datasets; "depreciation", "depreciation per year", "new_price" and "year_difference" from the formulas below:

df[year_difference] = df[year_kbb] - df[year]

df['new_price] = round(df[price_kbb]/df1[year_difference].apply(lambda x: 1.02 ** x), 3)

df[depreciation] = (df[new_price] - df['price'])/ df[new_price]

df[depreciation per year'] = -1*((-df['depreciation']+1)**(1/df[year_difference])-1)

Where:

- depreciation: The percentage of the car has depreciated since “price when new.”
- new_price: un estimated price derived from removing the expected inflation from the latest price over the years since the car was new.
- year_difference: The number of years elapsed between the latest year and cars’ age in the Kaggle dataset.
- depreciation per year: The average depreciation compounded yearly basis.

Year vs. Depreciation Per Year

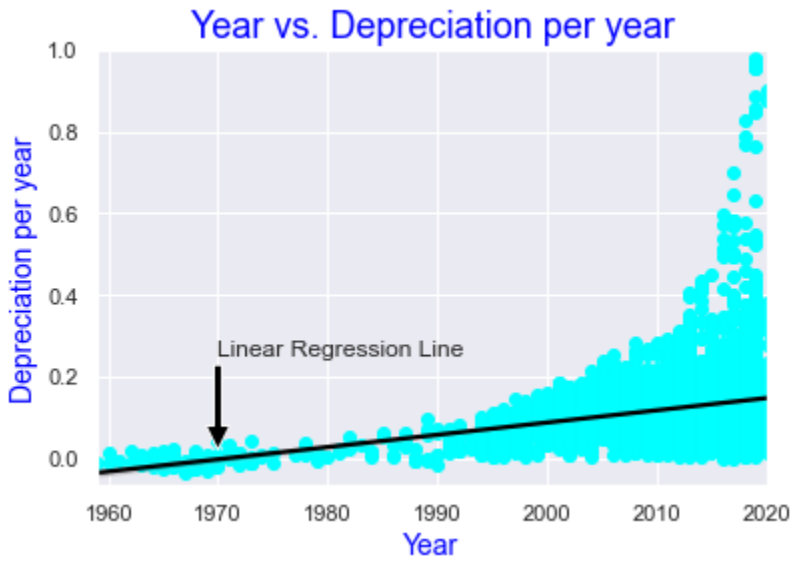

Figure 1.1: year vs. depreciation per year

The scatter plot distribution covers used cars made from 1960 to today. The slope of the regression line shows the negative slope towards the oldest cars indicating that depreciation per year is high in new cars. There is an exponential trend that reveals depreciation is faster in new cars.

The Distribution Manufacturer vs. Depreciation per Year

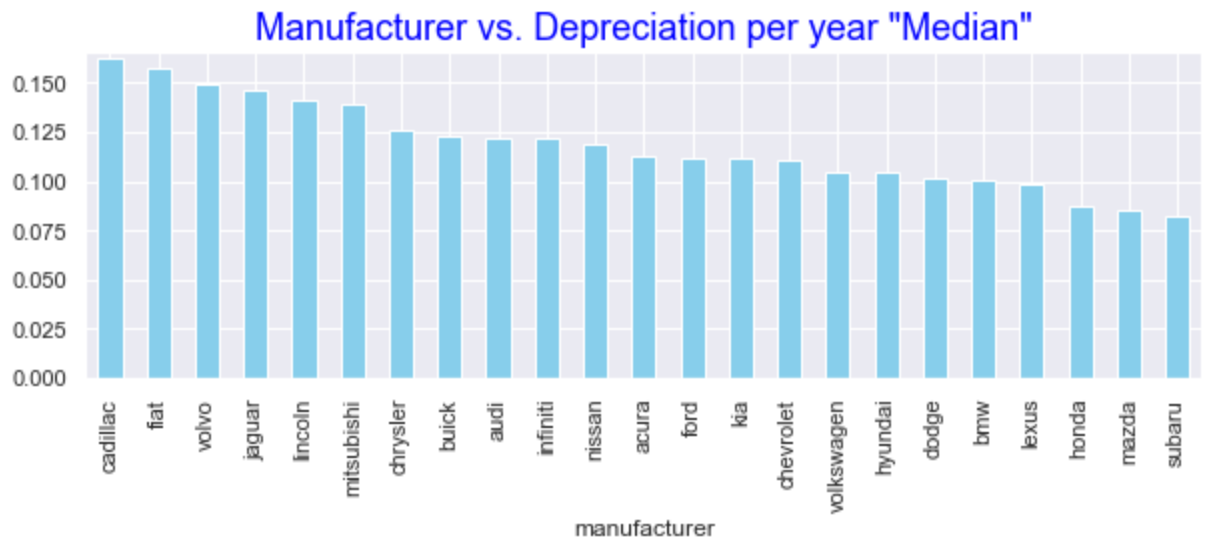

Figure 2.1: manufacture vs. depreciation per year

Figure 2.1 shows that the median distribution of depreciation per year varies among manufacturers. This is potentially due to the importance of customers' trust in a brand to make durable cars or could be due to the actual difference in the durability of the cars produced by these manufacturers. As we see Cadillac, Fiat, Volvo, Jaguar, and Lincoln manufacturers are the top five in depreciation per year respectively, on the other hand, Subaru retains the value better than any other cars. Mazda, Honda, Lexus, and BMW manufacturers are sorted next to Subaru in terms of value retention.

The Number Cylinders vs. Depreciation Per Year

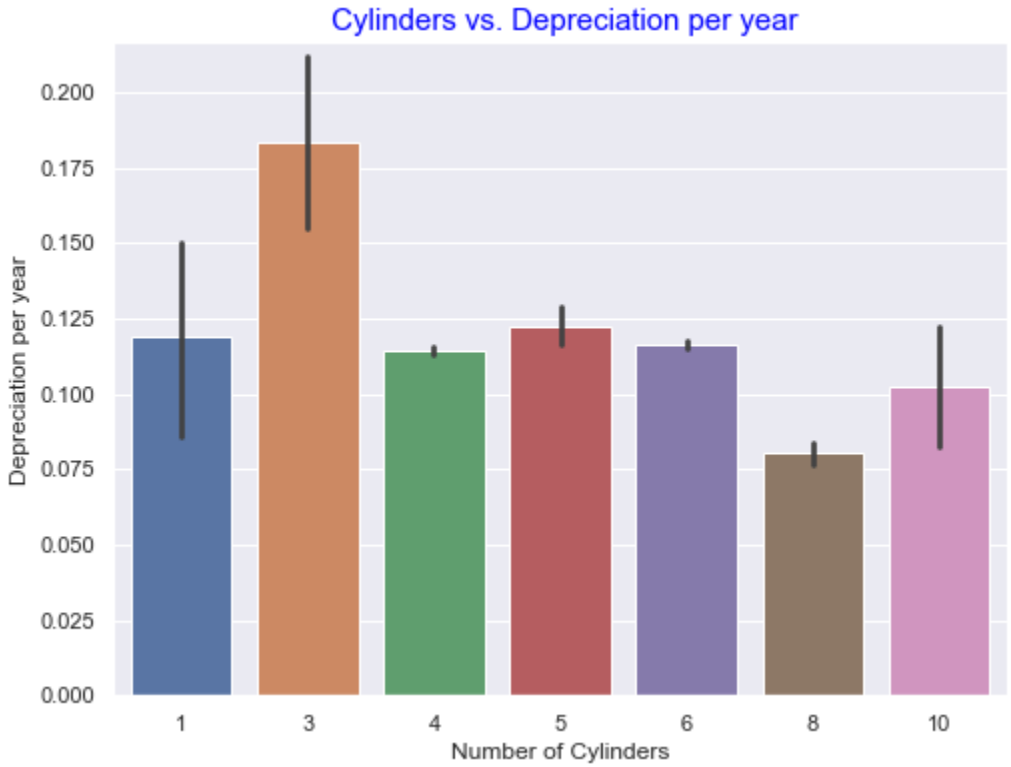

Figure 3.1: cylinders vs. depreciation per year

The number of cylinders a car has affected the overall power of the engine. Figure 3.1 shows depreciation per year rate significantly varies from 8% to 18%. Cars with 3-cylinders show over 18% of the maximum depreciation per year; on the other hand, vehicles with 8-cylinders maintain their value better than most others.

Odometer vs. Depreciation Per Year

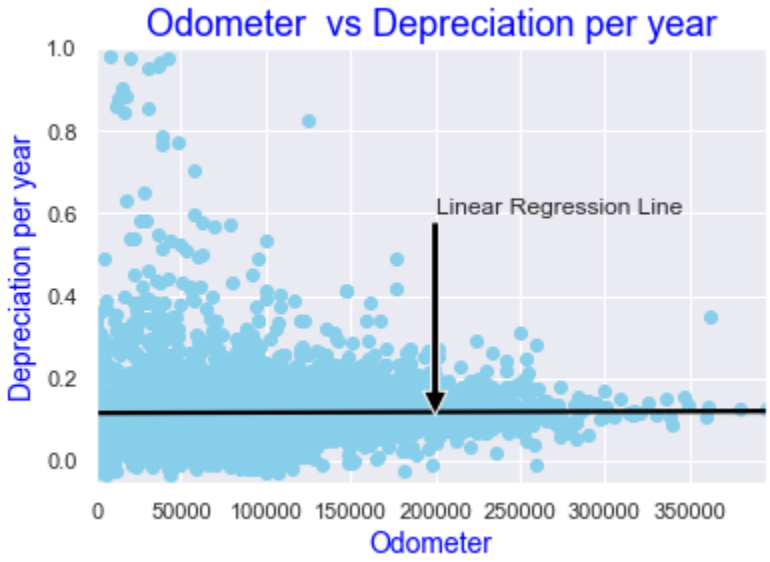

Figure 4.1: scatter plot - odometer vs. depreciation per year, and best fit line.

The regression line shows as the odometer increases the depreciation per year remains constant for the entire period of cars' service. The scatter data points distribution indicated that depreciation faster over the new cars. Moreover, The data points seen below zero reveal the negative depreciation per year or simply interpret as car's values are appreciated such as Antique cars.

The Distribution of Year vs. Price

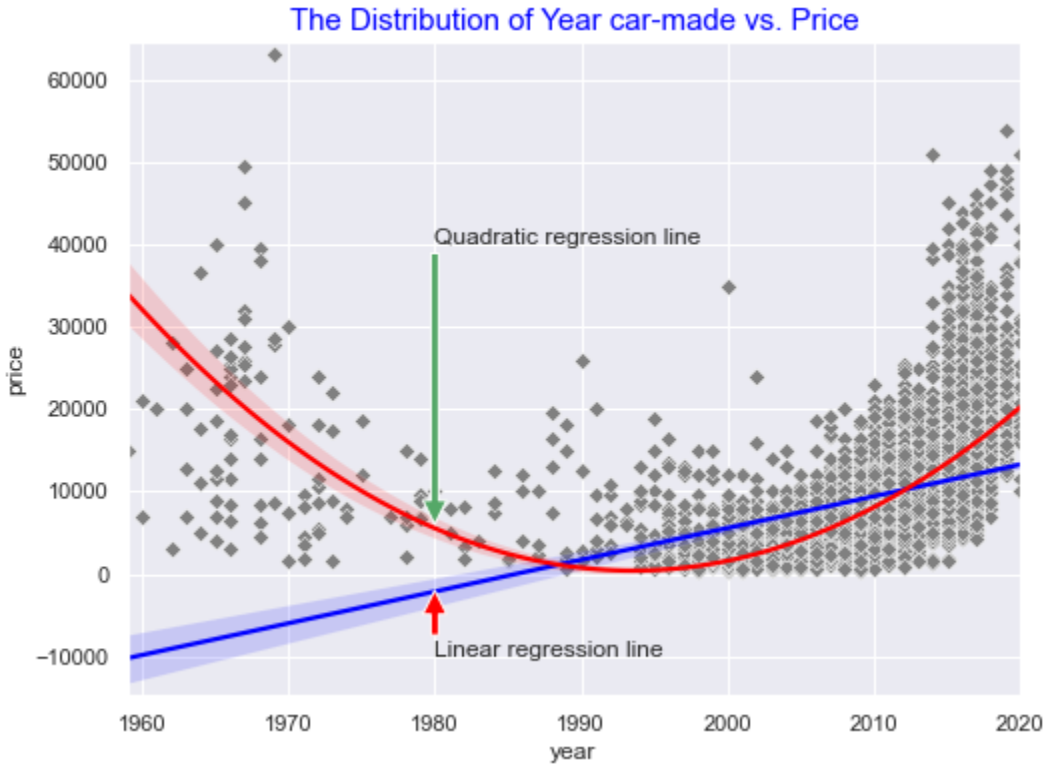

Figure 5.1: scatter plot - Year versus Price, linear and quadratic model regression lines.

Figure 5.1 provides used car prices sold in the Kaggle dataset. The depreciation was retrieved from prices sold at the Kaggle dataset minus the estimated price when new. Prices sharply dropped between 2015 and 2020 and declined steadily until 1990; however, cars made before 1990 could recover the depreciated values. For example, cars made between 1960 and 1970 sold as equally as today's most expensive, revealed that antique cars contributed to price retention. The quadratic regression line fits the model better to predict the used cars price distribution.

Title status vs. Depreciation per year

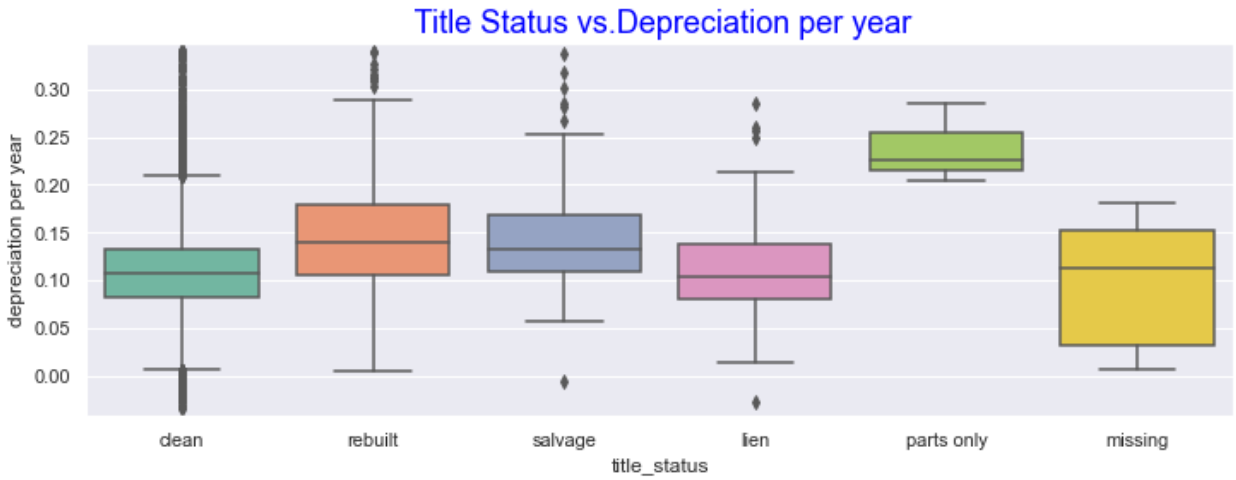

Figure 6.1: box plot - title status vs. depreciation per year

In the dataset, the vast majority of cars have a “clean” Title Status which unsurprisingly has a median depreciation per year close to the median for the dataset as a whole. However, other Title Statuses do seem to affect depreciation per year. In particular the “Parts only“ title status - attained the highest median in depreciation per year compared to the other title status. The most likely reason might be, the above percentage, “part only” title status has the least insufficient sample size which inevitably leads to statistical bias. The second reason could be the title status deal may have a regular interest rate inflating the car’s price until the debt is clear, which is associated with raising the depreciation per year.

Mpg vs. Depreciation per Year

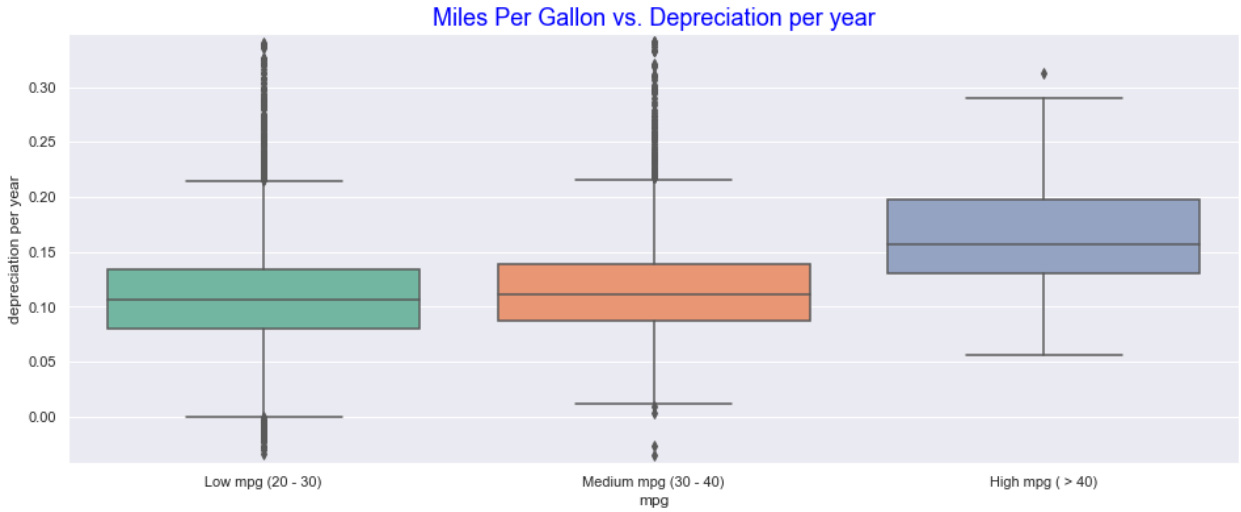

Fig 7.1: box plot - mpg vs. depreciation per year

One can assume that cars consuming less fuel may retain their value better. But Fig 7.1 shows the depreciation per year median is high on used cars traveled at a high miles-per-gallon rate. Most of the time cars travel over 40 miles per gallon either the cars are electric or hybrid. The likely reason for high depreciation per year could be the initial cost of these cars is high or clients' demand is very low due to maintenance service requiring expertise and so does the cost.

New Price and Depreciation Per Year

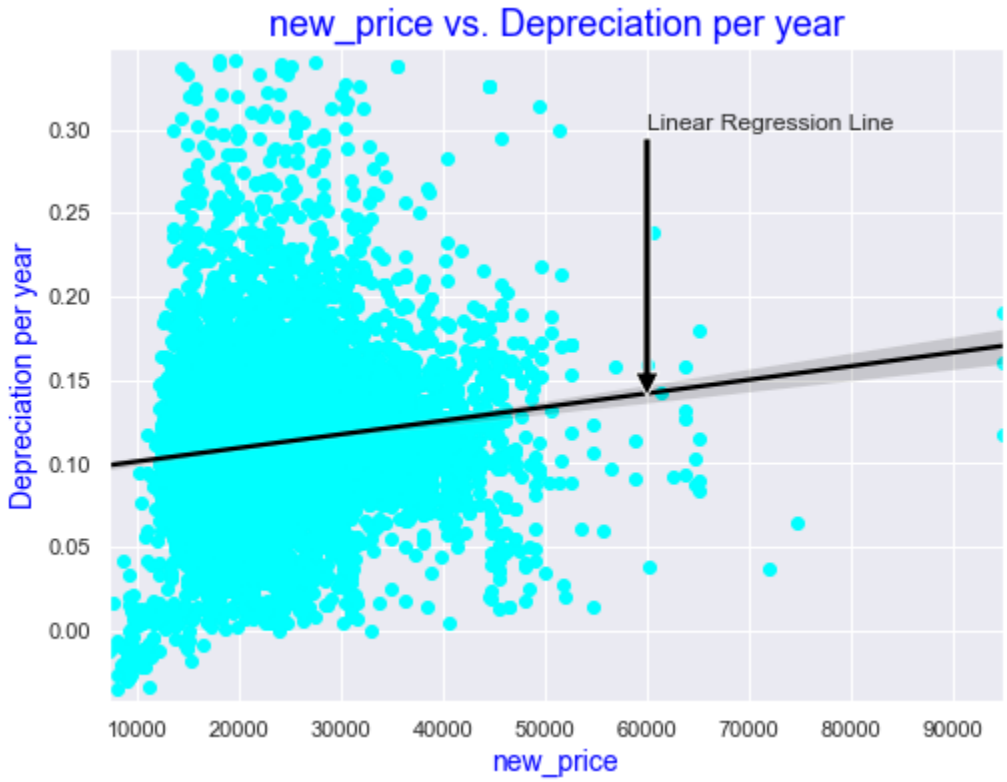

Fig 8.1 scatter plot on new_price vs. depreciation per year

The regression line shows the depreciation per year increases as the price of the car's values become more expensive.

Random Forest Feature importance:

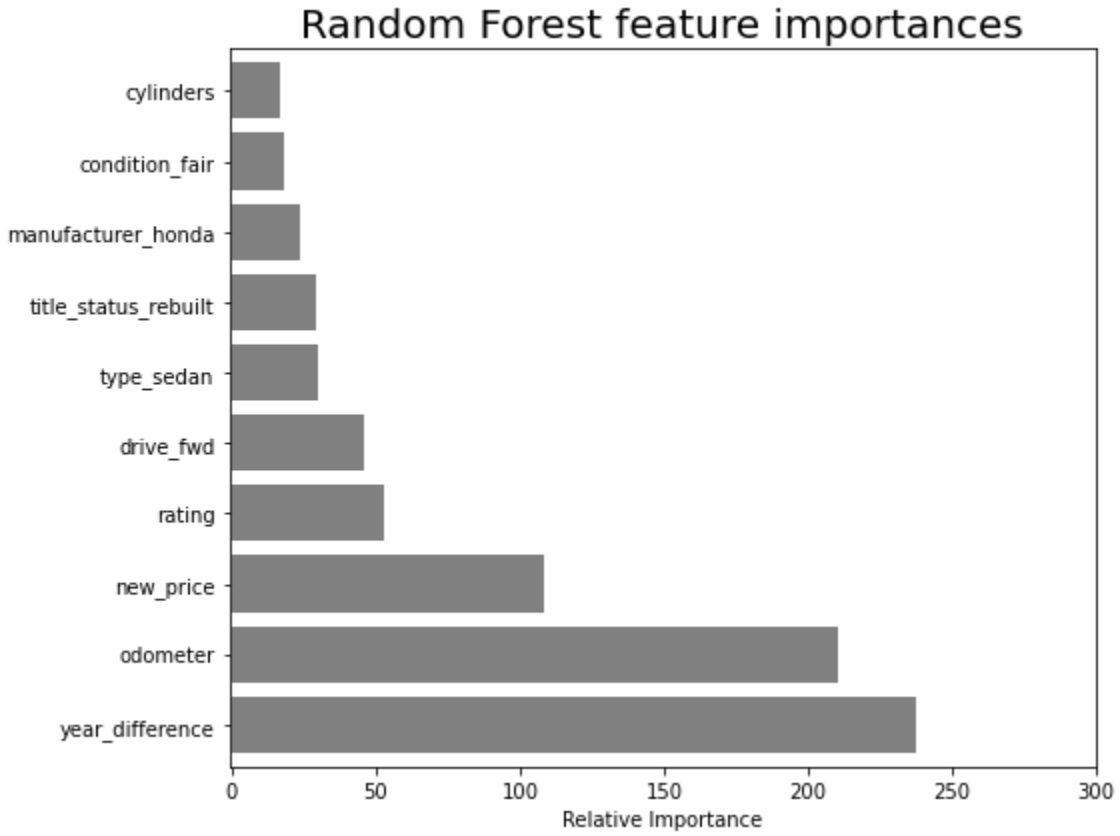

Fig 9.1: scatter plot - depreciation per year versus odometer.

Random forest regression algorithm yields the most important features. The six most important features to predict vehicles depreciation per year are as follow:

- year-difference
- odometers
- new-price, estimated Used cars price while it was a new brand.
- rating
- drive_fwd

year_difference and odometer are the prominent features attributed 45% to score, and new_price , rating and drive_fwd are the second most relevant features. Generally, the top five relevant features are responsible for 65.47% of the R-square score. The remaining 34.53% of R-square score retrieved from the other sixty-seven least relevant features

Modeling:

I chose five regression algorithms to predict depreciation per year, planning to compare the results to see which worked best after grid searching to optimize the hyperparameters for each.

1. Linear Regression
2. Ridge Regression
3. XGBoost Regressor
4. Decision Tree Regressor
5. Random Forest Regressor

Model Metrics

Model	Parameter	Metric	Metric
		R-squared	RMSE
Linear Regression	None	0.47458	0.0327
Ridge Regression	Alpha = 0.1	0.4555	0.0333
XGBoost Regressor	**Alpha = 0.1 n_estimators = 200**	**0.6212**	**0.02720**
Decision Tree Regression	'max_depth': 11, 'min_samples_split': 18	0.41356	0.03964
Random Forest Regression	Max_depth = 30 n_estimatores=200	0.596646	0.02842

Figure 10.1 Model scores

Recommendation:

After testing, and hyperparameter tuning I was able to achieve an R-squared score of 62.12% using a XGBoost regressor. This model could be used by used car dealerships to better price their vehicles, and could also be used by buyers to ascertain whether a vehicle is over or undervalued.

More generally, the first piece of advice I'd have for anyone looking to buy a used car, that is interested in having that car retain its value, would be to avoid cars with low odometer readings. Cars with greater odometer values tend to have a low depreciation per year. I'd also recommend buying cars that have a lower value when new, as expensive cars tend to depreciate more quickly over time.

Recommendation for further improvement:

The limited number of features and small sample size preclude the predictive power of the models, hence for further improvement, I could add more features and use a large sample size to achieve a better score. In my data, the new price of a vehicle was inferred rather than directly measured and this may also have increased error - so getting the actual new price data might be helpful. Furthermore, if I had a larger dataset I might be able to build individual models for each manufacturer which might be more accurate as the way features interact with depreciation may vary by manufacturer.