# MVP: What drives the price of a used car?

## Background:

The goal of this project is to build a linear regression model that would predict the price of used cars in a 20 miles radius of the following ZIP Code: 11221 in New York. The model can be used to understand what drives the price of a used or certified car by looking at features including but not limited to: mileage,average mpg, accidents reported, model,make,year,engine specifications,color,etc..

## Data Gathering, Cleaning and Exploration:

Data was gathered from the website Cars.com using Beautiful Soup.

Initial Data Cleaning consisted of breaking down categorical and numerical variables into further categories so they are interpretable, the examples listed below are non-exhaustive:

- Extracting Make,Year, Model from Name
- Counting the number of safety,convenience and entertainment features by cars
- Further breaking down categorical features to be relevant to the problem, for example engine was broken down into valvetrain and engine displacement

## Baseline Model and Initial Observations:

### a. First Try:

The baseline model included the following numerical features: Year, Mileage, Average MPG, Rating, Safety Count, Convenience Count, Entertainment Count.

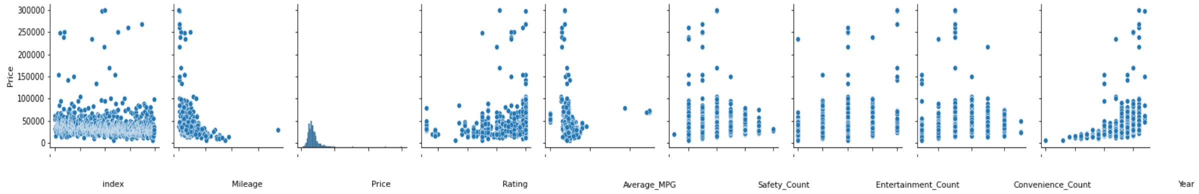
The initial results below showed that there was a lot of room for improvement:

*Linear Reg Mean Val Score :* 0.14069263134941934.
*Linear Reg Train Score:* 0.1887299509435162.
*Linear Reg Test Score:* 0.17623785245289392.

## b. **Second Try:**

In order to improve on the model I will walk through a number of questions I tried to answer.

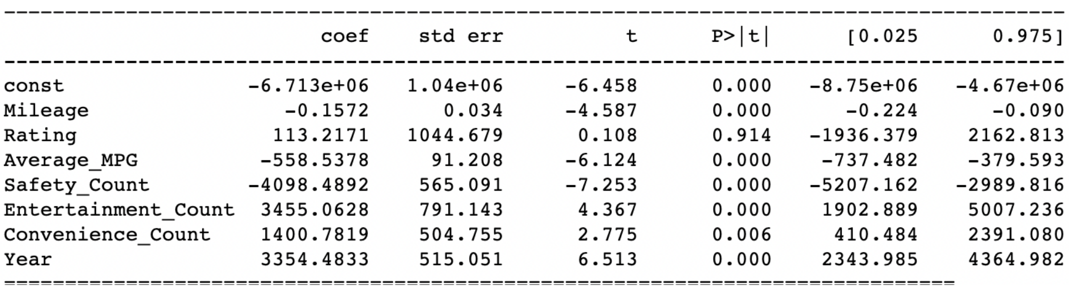
*Question 1: Is there a better way to describe the relationship between dependent and independent variables?*



A few non linear relationships standout:

- Year and Price could benefit from a polynomial (x^2) relationship
- Mileage seems to have an inverse (1/x) relationship with Price

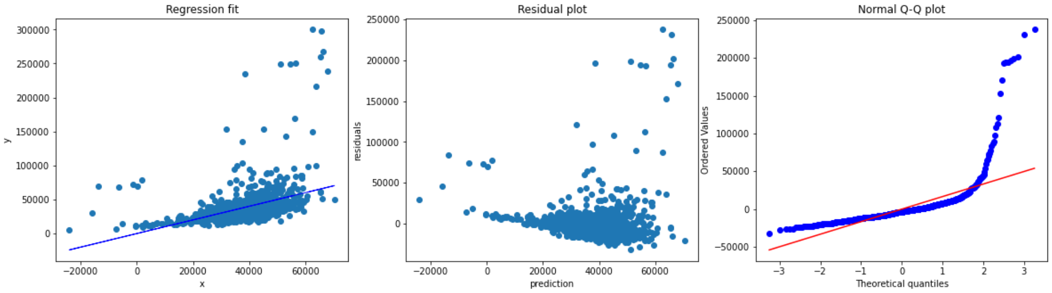
*Question 2: Are there any variables that are irrelevant to the problem at hand - they show no relationship with the target?*



```
==========================================================================================
                         coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------
const              -6.713e+06   1.04e+06     -6.458      0.000   -8.75e+06   -4.67e+06
Mileage               -0.1572      0.034     -4.587      0.000      -0.224      -0.090
Rating               113.2171   1044.679      0.108      0.914   -1936.379    2162.813
Average_MPG         -558.5378     91.208     -6.124      0.000    -737.482    -379.593
Safety_Count       -4098.4892    565.091     -7.253      0.000   -5207.162   -2989.816
Entertainment_Count 3455.0628    791.143      4.367      0.000    1902.889    5007.236
Convenience_Count   1400.7819    504.755      2.775      0.006     410.484    2391.080
Year                3354.4833    515.051      6.513      0.000    2343.985    4364.982
==============================================================================
```

It looks like Rating doesn’t have a very strong relationship with the target, it will be dropped

Question 3: *What do the diagnostic plots tell us about distributions and residuals?*



- Negative Prices are being Predicted, which doesn’t make sense.
- It looks like the distribution of the target is skewed given the cone shape of residuals and could use a log transformation.

***Applying the changes:***

***With the 1/x Mileage to Price Relationship:***

After applying all of the changes mentioned above, the new results are the following:

A great improvement in validation scores, however train vs  test score is not  great and likely overfitting.

Linear Reg Mean Val Score : 0.40651690862792866
Linear Reg Train Score': 0.4708379767256461
Linear Reg Test Score: 0.1559599052951104

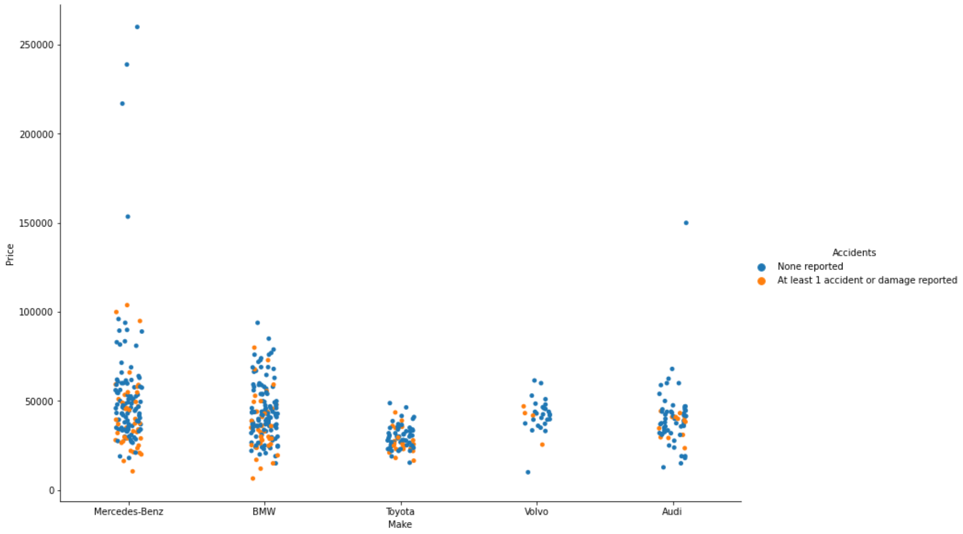
***<u>Excluding the 1/x Mileage to Price Relationship:</u>***

We see a lesser mean validation score, however train and test do better in terms of the difference between scores,meaning the model is less overfit.This baseline model will be used onwards.

Linear Reg Mean Val Score : 0.3584938890455887
Linear Reg Train Score: 0.406908204072436
Linear Reg Test Score: 0.3749749523098348

**<u>Next Steps:</u>**

- Take a closer look into outliers and multicollinearity issues (VIF Scores)
- Layer in categorical variables: make,model,engine,zip code…etc
- Ridge and Lasso to remedy overfitting
- Look at potential interaction variables, the plot below shows potential interaction between make and accidents reported:


Price
250000
200000
150000
100000
50000
0
Mercedes-Benz
BMW
Toyota
Volvo
Audi
Make
Accidents
● None reported
● At least 1 accident or damage reported