Predicting Car Price

AUTO LEARNING

Dongtao Jiang | Springboard Data Science Career Track | 7/22/2020

Table of Contents

Abstract

The cars dataset was originally scraped from thecarconncetion.com with full specs (dimensions, fuel economy, performance specs, safety features, warranty, etc.) and pricing information. Data wrangling and exploratory analysis was conducted subsequently. Car price distribution, buying guide, and price evolution over the years were illustrated. After imputation, one-hot encoding of the dataset, machine learning was implements using algorithms including linear regression, ridge regression, lasso regression, decision tree, random forests. A r2 score close to 0.99 was achieved by Random Forest. Feature importance analysis revealed the torque spec and displacement are the most important factor determining the car prices.

Introduction

A customer is always concerned about if the money paid for the product is worth it. Car is a commodity with sophisticated technological wonders built into it. In any kind of modern car, there exist deep knowledge bases in mechanical engineering, electrical engineering, aerodynamics, software engineering, chemical engineering, automation, etc. Nobody has such full understanding and knowledge to make sound judgement of a new car's price. Car value and reputation are usually based on many years user experience, popularity, quality testing, etc. To the majority of car buyers, it would be a great tool to use a data-driven objective model to find the best quality car with all technical features being taken into account. To a customer that doesn't have an engineering background, the following evaluation metrics that are extracted from thecarconnection.com would help.

- Style: Points can be earned or lost based on above- or below-average interior and exterior style; excellent or poor interior or exterior style; and exceptional (or very poor) style.

- Performance: Points can be earned or lost based on powertrain performance; ride and handling performance. Exceptionally quick (0-60 mph in less than 5 seconds) or exceptionally slow (0-60 mph in more than 10 seconds) can earn or lose an additional point. An additional point can be awarded (or lost) for exceptional circumstances, i.e. off-road prowess, or supercar credentials.

- Comfort: Points can be earned or lost based on comfort in the front seats, back seats, or third-row seats (where applicable); good or bad interior storage and cargo capacity; and good fit and finish.

- Safety: Cars with official crash data gain points for a five-star overall rating by the NHTSA, or Top Safety Pick/Top Safety Pick+ status by the IIHS. An additional point is awarded for cars that come standard with full-speed automatic emergency braking. We award points for excellent outward vision and for abundant safety features and options such as parking assistance, surround-view camera systems, or driver-assistance features. Cars with official crash data lose points for a four-star overall rating by NHTSA, any "Marginal" IIHS or three-star NHTSA ratings, for poor outward vision, and when they lack forward-collision warnings and automatic emergency braking.

- Cars without crash data aren't given a rating at all. Cars with only partial ratings may be scored, generally when it improves their score.

- Features: Cars with excellent base equipment earn a point above average. Extra points can be added for exceptional available features, good value, good infotainment systems with screens larger than 7.0 inches, and good warranty or service programs. Cars may lose points for substandard or expensive features; bad feature packages; poor relative value; or bad warranty or service availability.

- Green: Cars are assigned a rating based on their EPA-estimated highway and combined mileage ratings. Plug-in and battery-electric vehicles start at 9. Electric-only cars with a range of more than 200 miles earn a score of 10. All other vehicles are sorted on a sliding scale based on EPA fuel economy.

Dataset

The new cars dataset was obtained from the following link,

https://www.reddit.com/r/datasets/comments/b6rcwv/i_scraped_32000_cars_including_the_price_and_115/

It was originally scraped from thecarconncetion.com that provide comprehensive care specs and prices. According to its website, the Car Connection is an automotive property of Internet Brands, which owns and operates the largest network of car buying and financing resources in North America, including CarsDirect, Motor Authority, Green Car Reports, and Auto Credit Express.

Listed below are all the specs of an example car that are used as features of our dataset. It is divided into 5 categories:

1. Dimensions
2. Fuel Economy
3. Performance specs
4. Safety Features

5. Warranty

The details of feature names and corresponding full specs can be found in Appendix I.

Software Packages

- Python 3.6
- Pandas 1.0.3
- Numpy 1.15.4
- Sklearn 0.23.1
- Matplotlib 3.2.1
- Scipy 1.10
- Missingno 0.4.1

Data Wrangling

Raw datasets usually have missing values, type errors, mixing data types, irregular format, lacking features and all kinds of other unexpected errors existing in them. Therefore, it is necessary to clean the raw dataset and fix all those problems in order to transform the dataset into the desired form that is suitable for applying all kinds of machine learning algorithms.

For our cars dataset, the following procedures have been carried out

- Rename column names shorter and regularized
- Removing Extraneous Data
    - Columns with 100% missing values
    - Remove duplicate rows
    - Remove columns with only one unique value
- Prepare target column
    - Drop the few rows that have missing values. Remove '$ 'sign and ','.
- Extract Year and Model from one column
- Define function to convert relevant columns to floats using pd.to_numeric()
- Clean up column 'Displacement'
    - Fill missing data with empty string to avoid error during cleaning operations.

- replace 'L/...' and strip white spaces right and left
- Strip leading and trailing white space
- Use len() to list out other data to be cleaned.
- Romove '/xx'
- Remove ' (152)'
- transform '39.5 Cu.in. Range Extender' to liter
- calculate from cubit inch to liter:

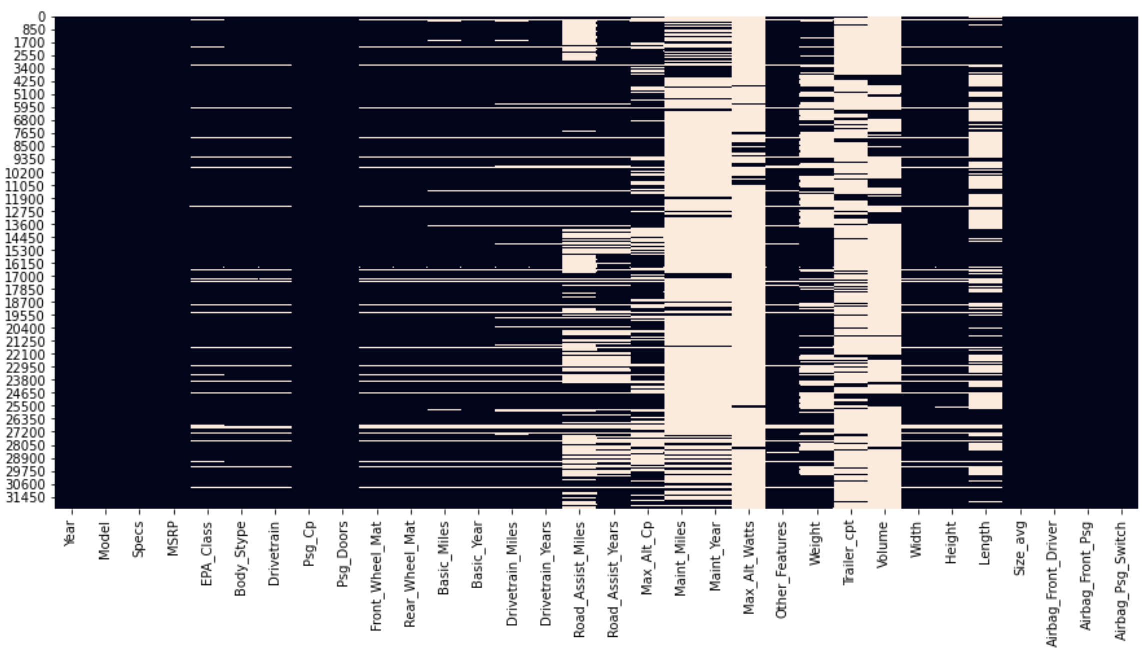
– Two columns ['EPA Class', 'EPA Classification'] are identical. Remove one.

Exploratory Data Analysis

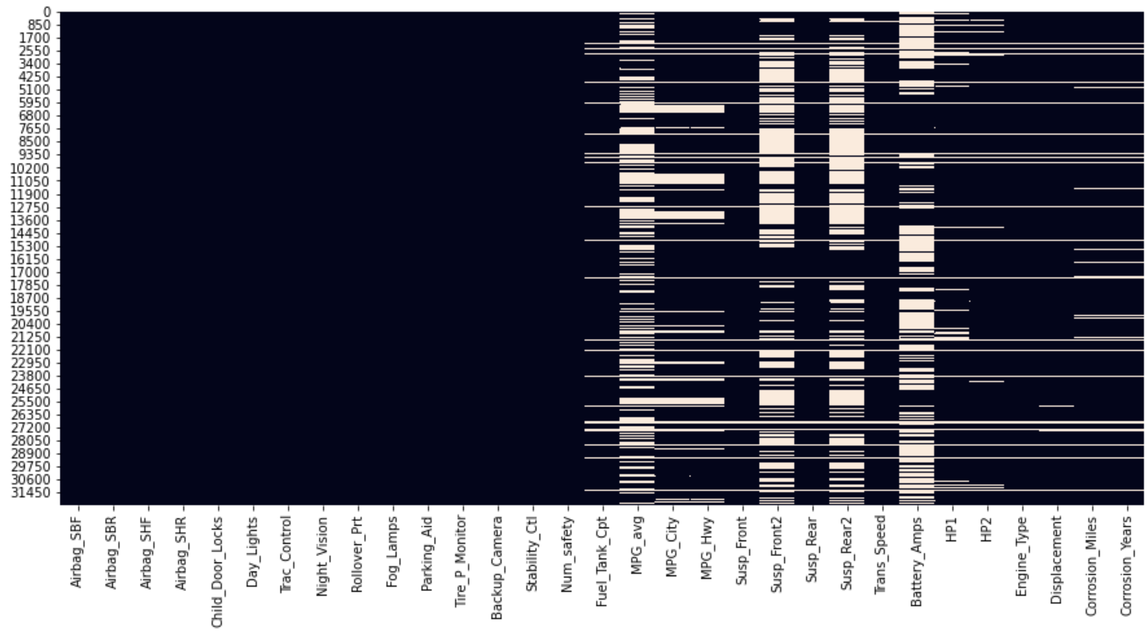
Missing data

How the missing values are distributed in the dataframe can be best visualized by the heatmap provided by seaborn package. The white area in the figure below represents missing entries. The following figure give us an Overall idea how much missing values are present in each individual feature. Columns like Volume and Trailer_cpt are mostly missing. There are no missing values in features like Year, Model, Specs, Safety related.

Missing Data Visualization 1/2



Missing Data Visualization 2/2



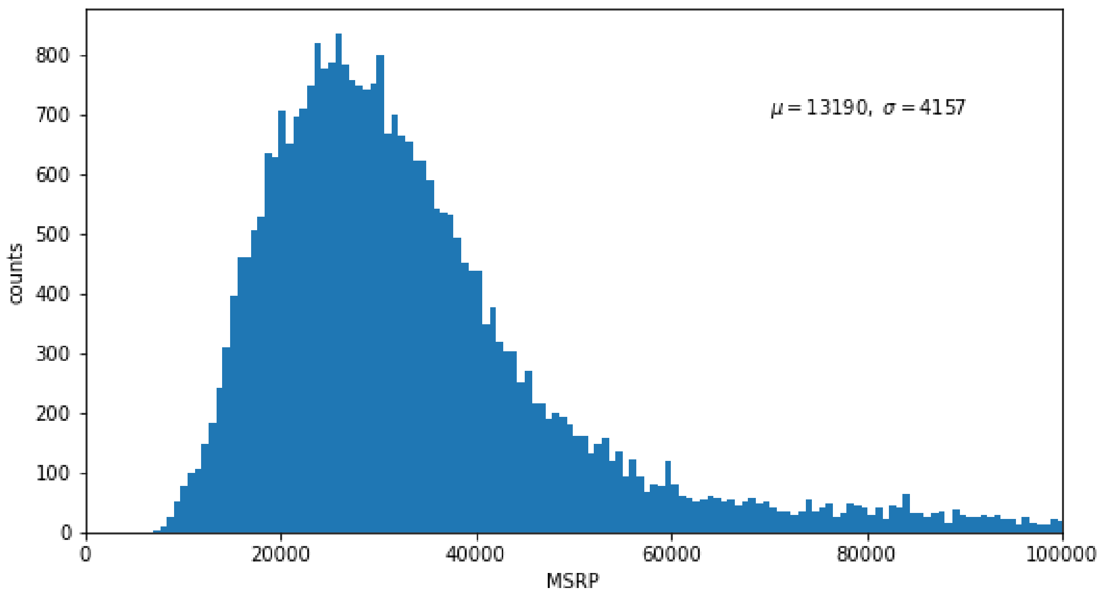
More analysis shows there are 15.7% of total dataset are missing values. The features with the most missing percentage are list below in descending order.

Feature	Fraction of Missing Values
Max_Alt_Watts	0.095107
Maint_Miles	0.083826
Maint_Year	0.083669
Trailer_cpt	0.083346
Volume	0.080817
Susp_Rear2	0.053926
Susp_Front2	0.053641
Battery_Amps	0.051202
MPG_avg	0.046682
Length	0.046416
Road_Assist_Miles	0.041294
Weight	0.034370
Max_Alt_Cp	0.029928
Road_Assist_Years	0.023916
MPG_Hwy	0.016538
MPG_City	0.016525
HP1	0.011472
Drivetrain_Miles	0.009987

Price distribution

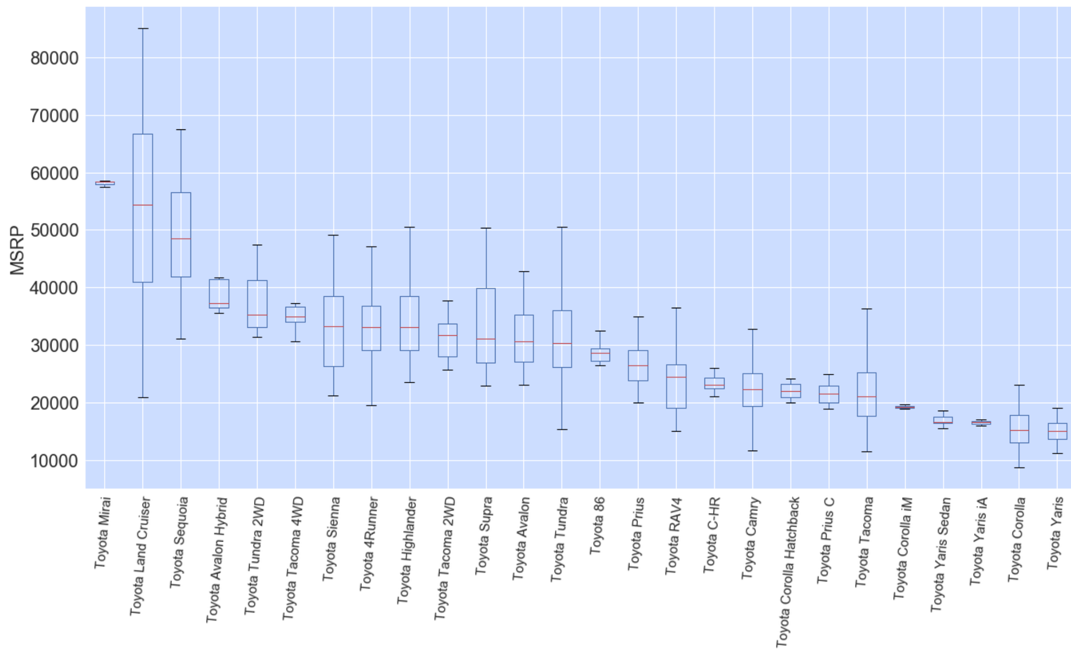
The price distribution of car models is shown below. Prices exceeding $100K for luxury cars are not plotted. The most frequent prices are around $29295. The standard deviation is: 4157.

Price Distribution (less than $100k)



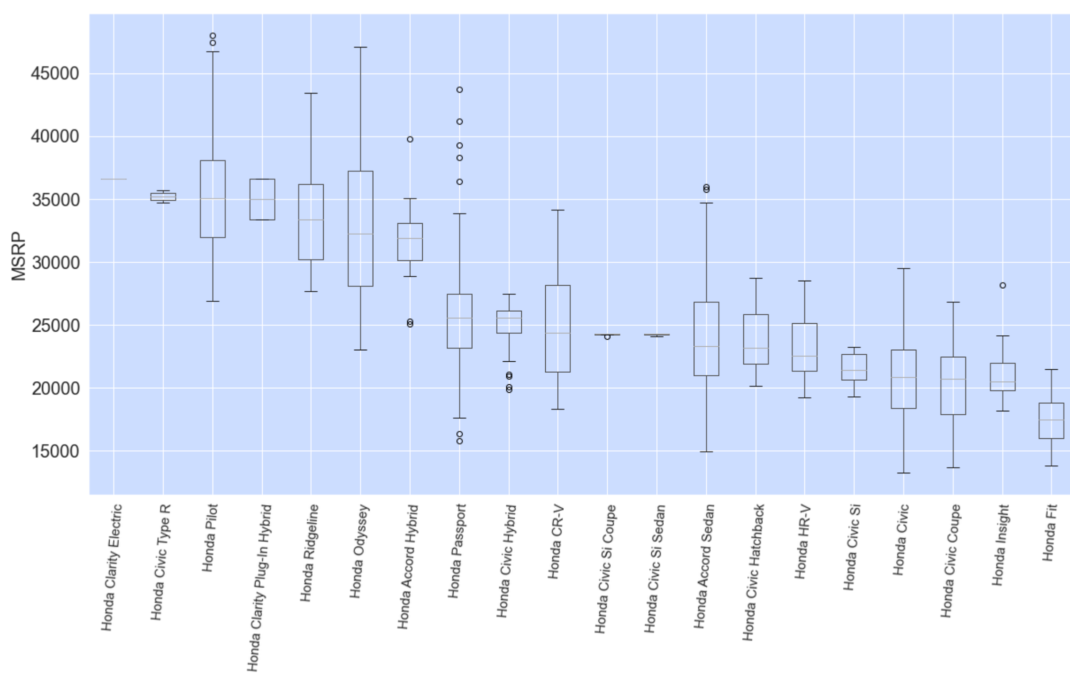
The high-end and low-end car models from Toyota are shown in the figure below with median prices of each model being sorted. Top-end Toyota models include Mirai, Land Cruiser, Sequoia, etc. Low-end models include Yaris, Corolla, etc.

Toyota Models Price Survey



The following is a survey of Honda cars. High-end Honda models include Clarity, Civic R, Pilot, etc. Low-end models include Fit, Insight, Civic Coupe, etc.

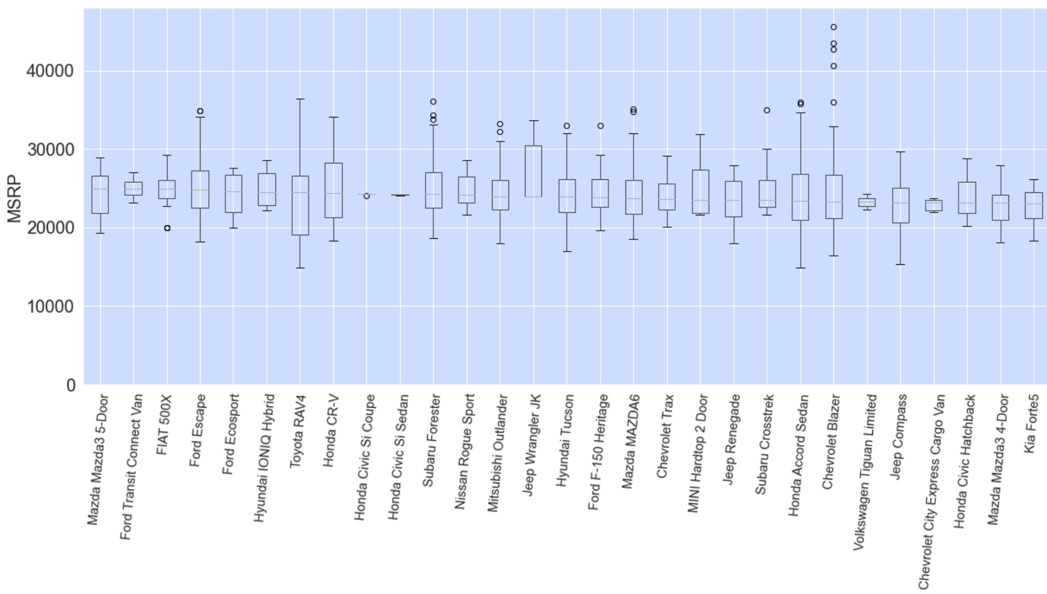
Honda Models Price Survey



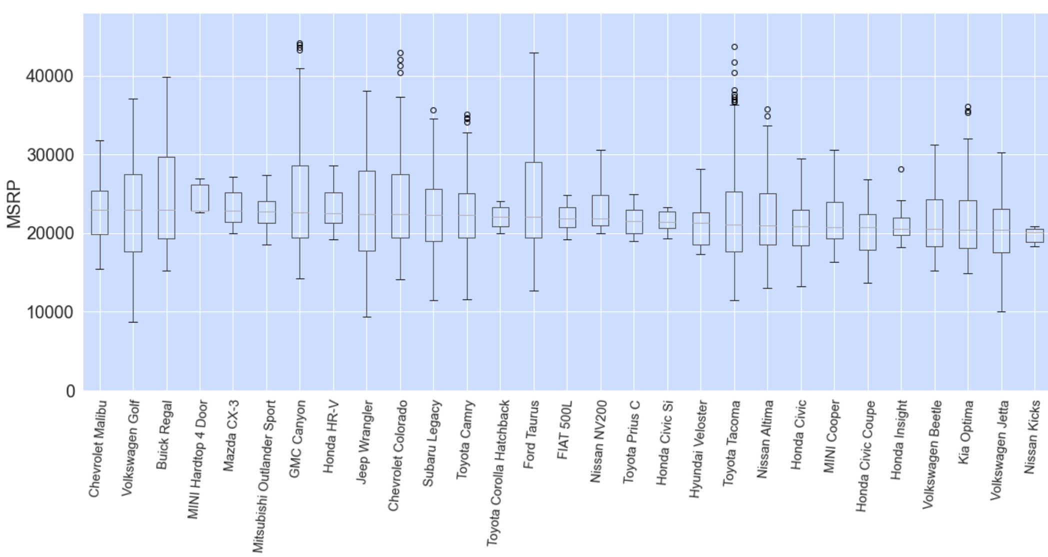
Buying guide for budget-tight customers

For budget-tight customers, the following box plots provide a convenient guidance for frugal customers to choose car models that are below $25,000. There are all together 90 models and split into 3 plots. They are ordered from high to low according median price of each model.

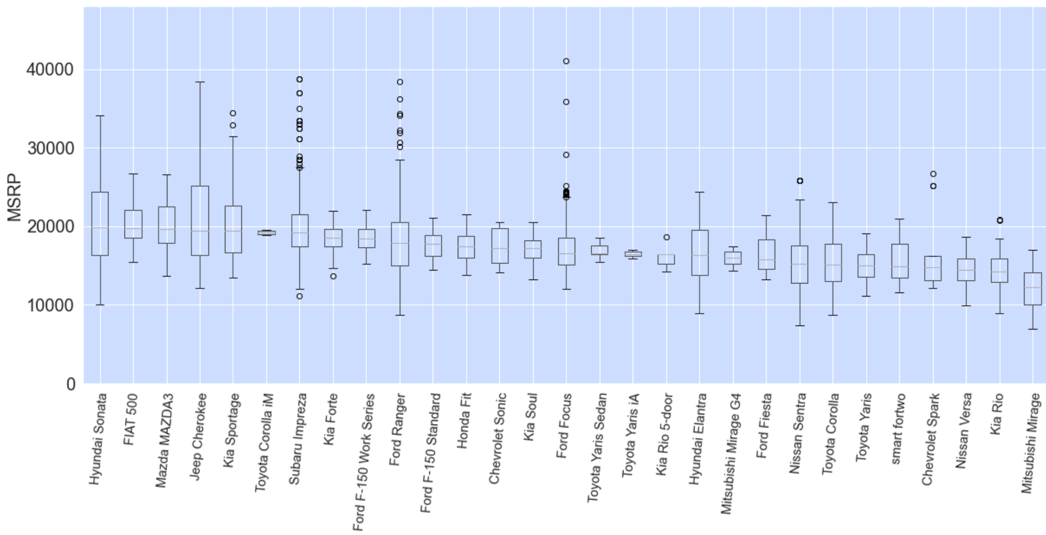
Car Models Less than 25K -1/3



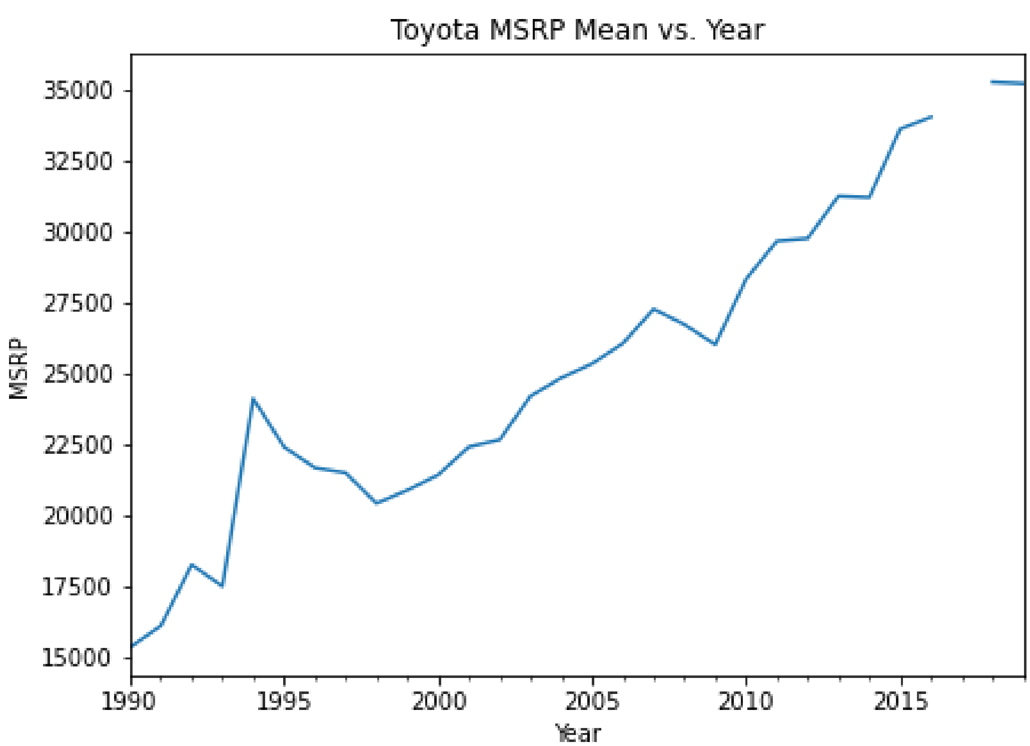
Car Models Less than 25K -2/3



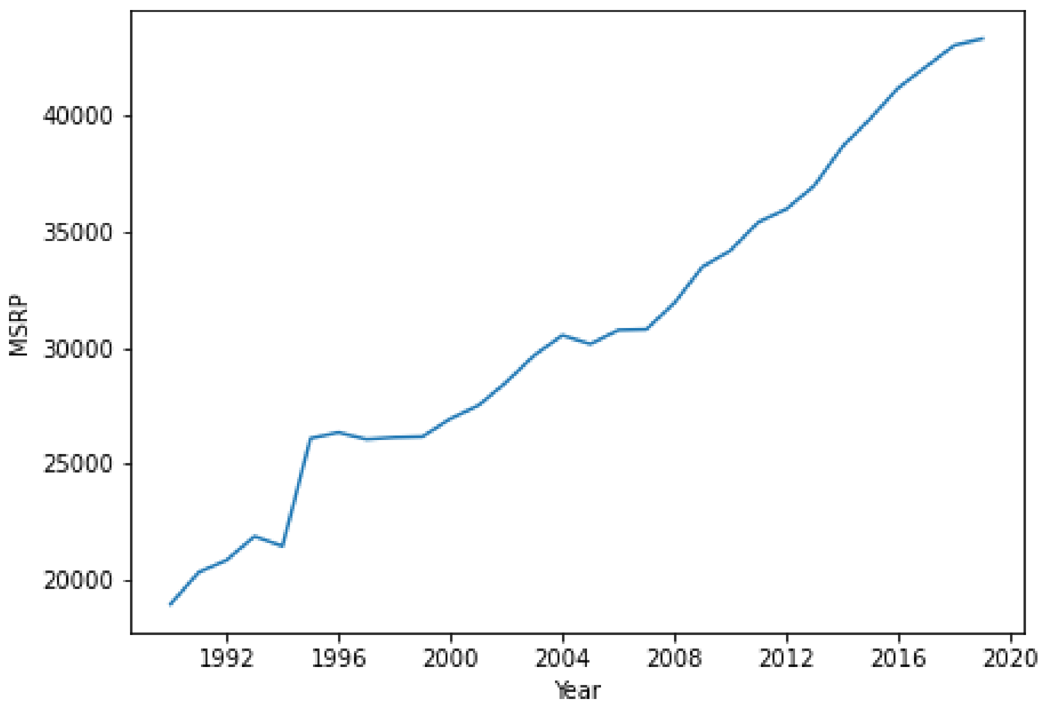
Car Models Less than 25K -3/3



Evolution of car prices over the years



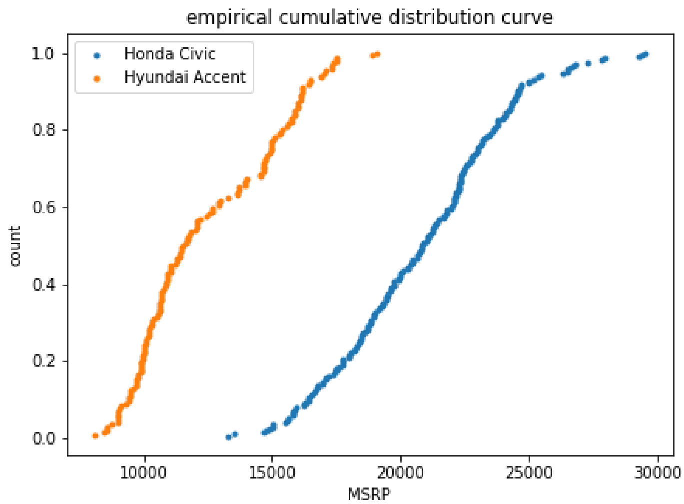
Evolution of Average Price Over All Models

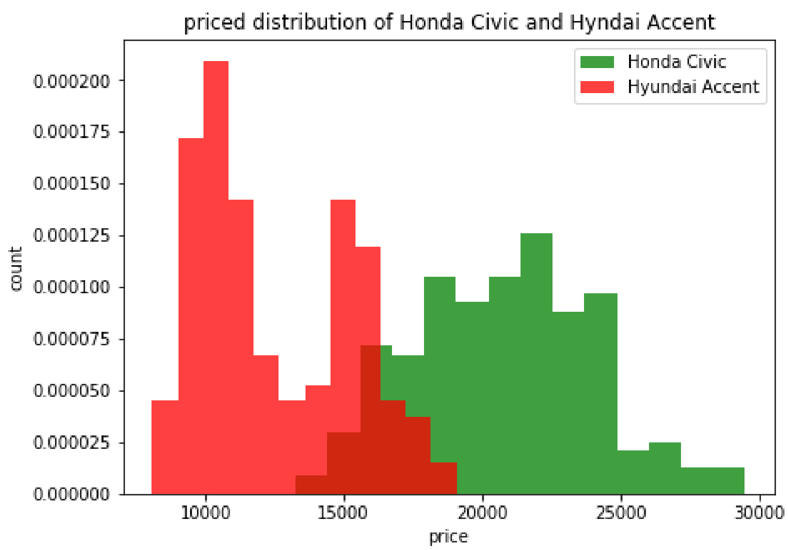


Car prices have been constantly increasing generally linearly over the last two decades, as shown above for Toyota and all models combined. This indicates year is an important feature to predict prices.

Hypothesis testing on two low-end car models

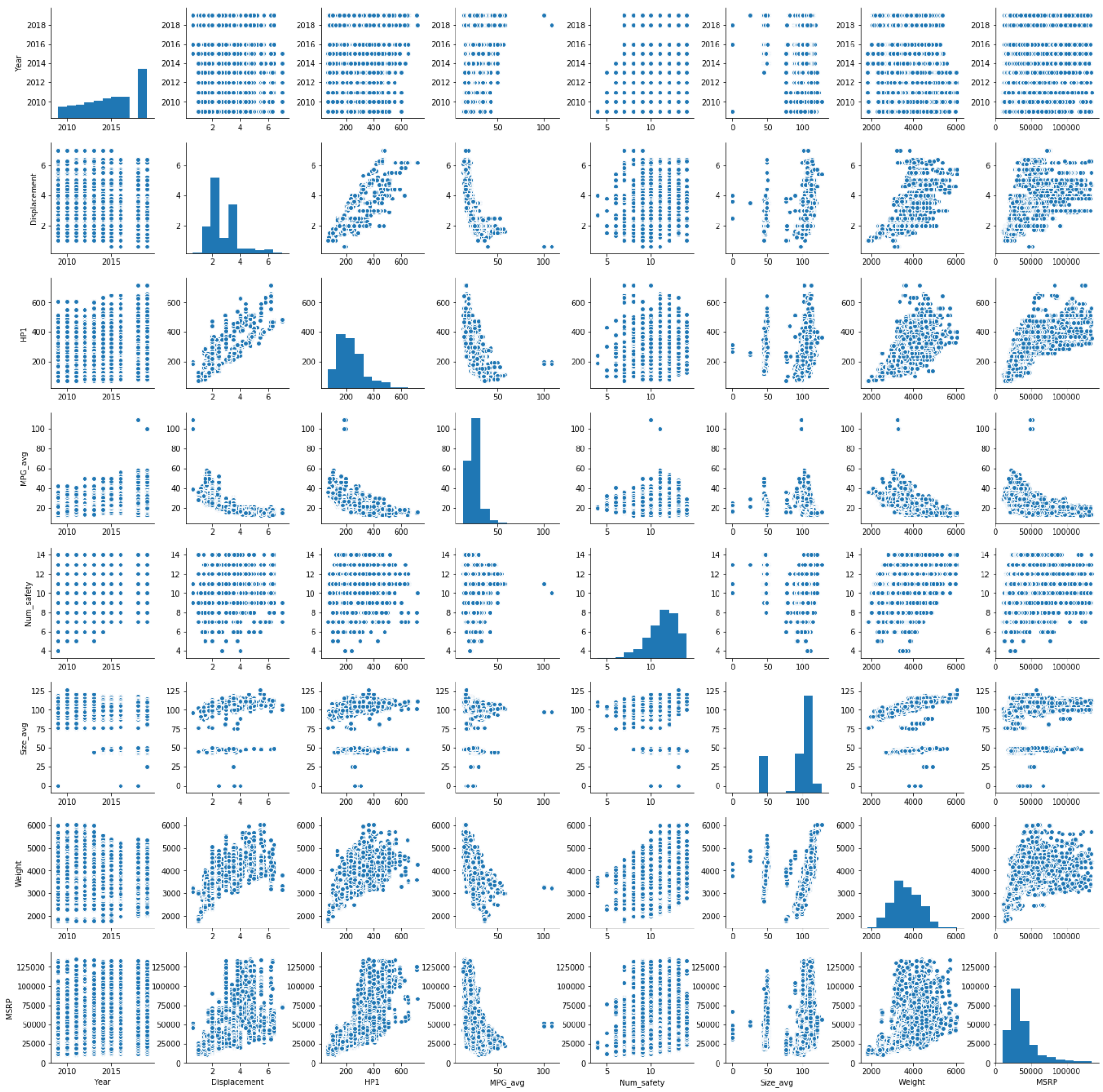
It was told that both Honda Civic and Hyundai Accent are good quality cars with good awesome deals. The box plot above shows the two models have large overlap in their price range. Is the difference in their mean price significant or negligible? Both empirical cumulative distribution carves and the statistic price distribution figure below clearly indicate a significant difference. A small fraction of price overlap exists.

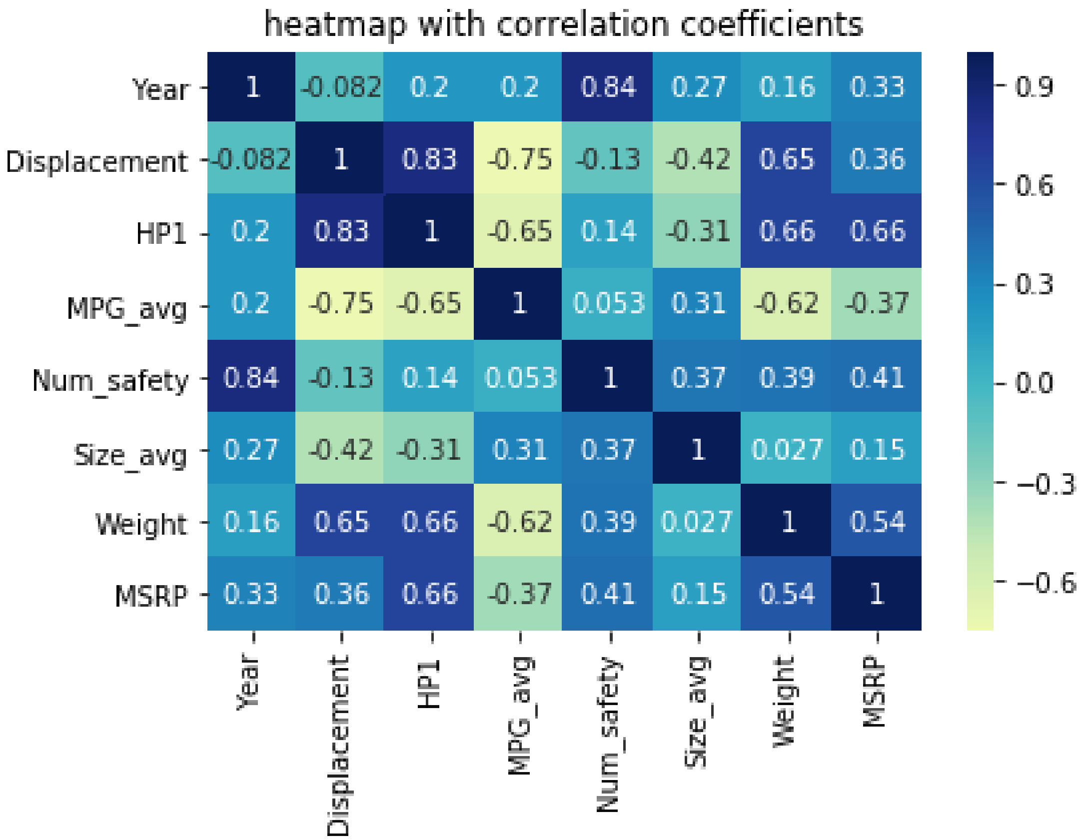




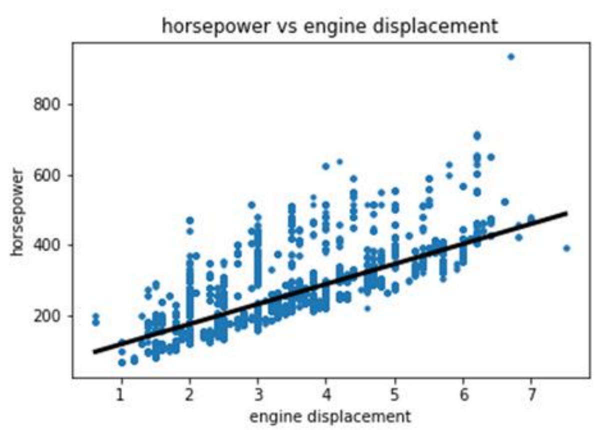
Correlation investigations

Possible correlations between features can be investigated by using the pair plots function coupled with heatmap function from seaborn, as shown below.

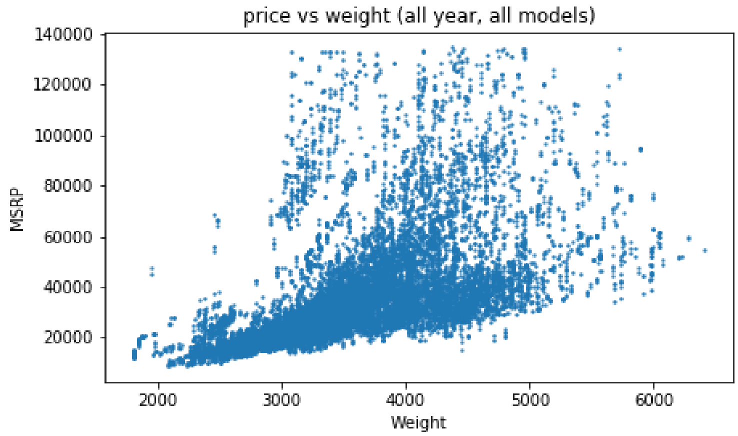

Year
Displacement
HP1
MPG_avg
Num_safety
Size_avg
Weight
MSRP



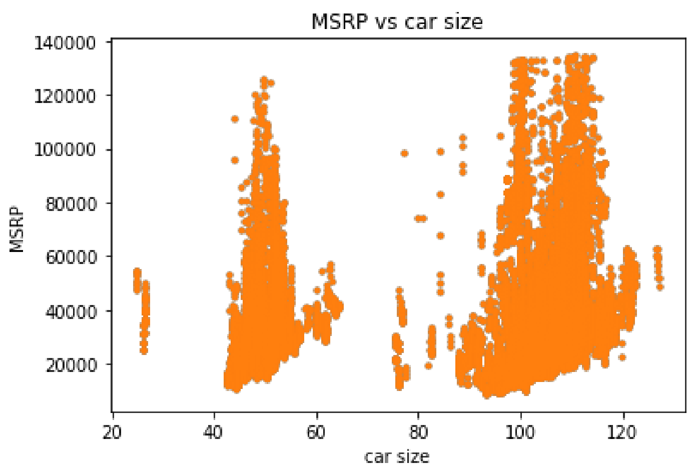
Both the pair plot and heatmap indicate a positive correlation between horsepower and engine displacement, which agrees well with physics and engineering principles. A linear fitting line was is shown in the figure below. Some correlation also exists between size and displacement, horsepower and MPG, price and horsepower, price and weight.



It is interesting to notice that the bottom one in the price range with the same weight is almost linearly proportional to price, as shown in the figure below. ¶



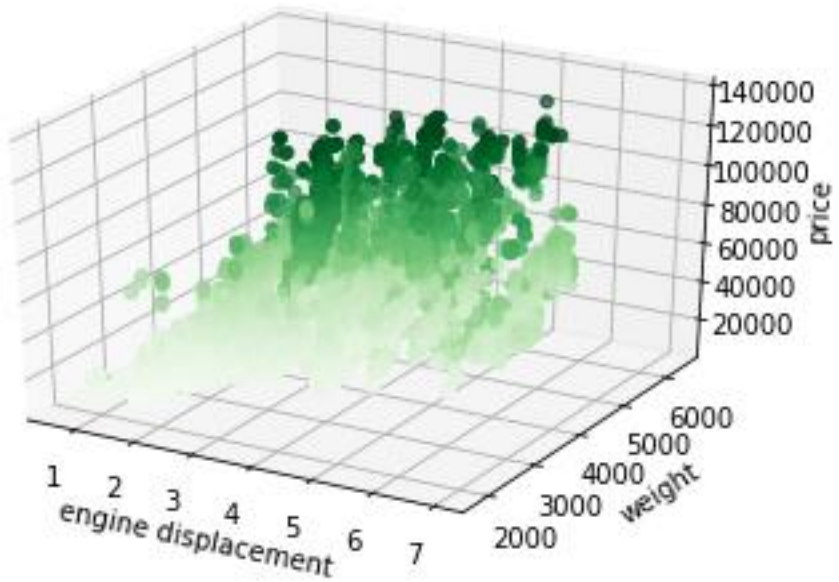
The influence of car sizes on the price are aggregated or clustered. There is no obvious trend for the price related to size.



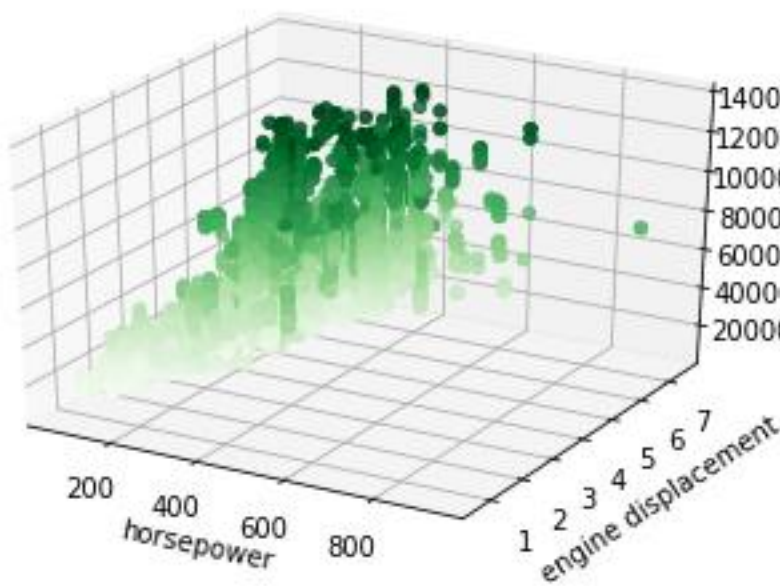
Explore correlations using 3-D Plots

By exploring combination of any two different features to see their effect on MSRP in the 3D plots, it looks like the following features are well correlated to the car price,

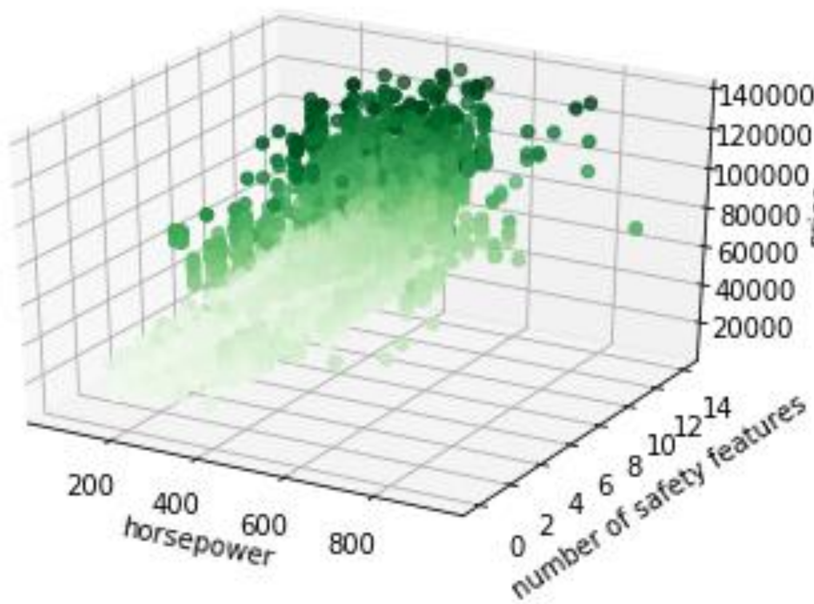
- horsepower
- weight
- engine displacement
- number of safety features



140000
120000
100000
80000
60000
40000
20000
price
1
2
3
4
5
6
7
engine displacement
2000
3000
4000
5000
6000
weight



140000
120000
100000
80000
60000
40000
20000
price
200
400
600
800
horsepower
1
2
3
4
5
6
7
engine displacement



Machine Learning

Imputation

An algorithm was developed to fill some the missing weight values in the dataset. The entry in 'Specs' is most reliable to infer the car price because it has brand, model and year information that determine the price in a predominant way. Therefore, by comparing the distance of strings between the missing and neighbors, we can make a reasonable guess of weight data. Levenshtein distance was used to calculate the distance. The rest of missing values was imputed by using MissForest and SimpleImputer.

One-hot encoding

All the categorical features were transformed to one-hot numeric arrays using get_dummies from sklearn.

Metrics

The coefficient of determination, denoted R2 or r2, will be used to evaluate the each model.

$$R^2(y,\hat{y}) = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \overline{y})^2}$$

It represents the proportion of variance (of y) that has been explained by the independent variables in the model. It provides an indication of goodness of fit and therefore a measure of how well unseen samples are likely to be predicted by the model.

Data shuffling

Since the data was aggregated by the car model and year in the dataset, it is necessary to shuffle the data. Otherwise, the train set and test might be inclined to certain car models. The drawback is not seeing the variety during the learning and prediction score is biased. Therefore, train_test_split is used here for the single purpose of shuffling the dataset. That is why the test_size is set very small, it is not really going to used for evaluation. The evaluation is done inside cross_val_score function using 20% of X_train and y_train, i.e., cv=5.

Random Forest has a built-in mechanism that is similar to cross-validation. 30% of dataset defined by test_size parameter in train_set_split() was used as held-out set for evaluation purpose.

Models

Linear Regression

Linear regression is a linear approach to modeling the relationship between a dependent variable and one or more independent variables. Linear regression is perhaps one of the most well-known and well understood algorithms in statistics and machine learning.

Ridge Regression

Ridge regression is useful to mitigate the problem of multicollinearity in linear regression, which commonly occurs in models with large numbers of parameters. In general, the method provides improved efficiency in parameter estimation problems in exchange for a tolerable amount of bias.

Lasso Regression

Lasso regression utilizes a penalty equal to the absolute value of the magnitude of coefficients. This type of regularization can result in sparse models with few coefficients, which is the ideal for producing simpler models. In contrast, Ridge regression doesn’t result in elimination of coefficients or sparse models.

Decision Tree

Decision Trees predicts the value of a target variable by making and learning simple decision rules inferred from the data features. Tree models where the target variable can take a discrete set of values are called classification trees; in these tree structures, leaves represent class labels and branches represent conjunctions of features that lead to those class labels. Decision trees where the target variable can take continuous values are called regression trees. Decision trees are among the most popular machine learning algorithms given their intelligibility and simplicity.

Random Forest

Random forest is a supervised learning algorithm. The "forest" it builds, is an ensemble of decision trees, usually trained with the “bagging” method. The general idea of the bagging method is that a combination of learning models increases the overall result. Random forest builds multiple decision trees and merges them together to get a more accurate and stable prediction.

Results of Models

Scoring

Machine learning scores for each model are detailed below. It takes much longer time for Lasso Regression and Random Forest to finish the task.

```
Linear Regression Cross-Validation Scores:
0.9627886187122039
0.9691926386058723
0.9657677293636685
0.9745384929385765
0.9674591456627235
Average Score: 0.9679493250566089
Wall time: 16.8 s

Ridge Regression Cross-Validation Scores:
0.9616672391622018
0.9637299689901893
0.9629544548681291
0.9670442472466491
0.9654943849039259
Average Score: 0.964178059034219
Wall time: 5.95 s

Lasso Regression Cross-Validation Scores:
0.96169873941943
0.9675678834459969
0.9651152364645428
0.9713790553792578
0.9664975109884204
Average Score on 5-Folds: 0.9664516851395296
```

```
Wall time: 2min 13s

Decision Tree Cross-Validation Scores:
0.9137332526752456
0.9138718489890844
0.9062166810688369
0.9124803029278153
0.9206118581501173
Average Score: 0.91338278876222

Random Forest Score:
Test score: 0.9863621135441437
Wall time: 2min 18s
```

The average r2 score for each model is list in descending order in the table below.

Models	r2_score	negative mean absolute error
Random Forest	0.986	-1443
Linear regression	0.968	-1002483
Lasso Regression	0.967	-3262
Ridge Regression	0.964	-3525
Decision Tree	0.908	-5470

Most of the models used give excellent goodness of fit that their r2 scores are mostly above 0.96. This indicate these models are of good choice.

Random Forest gives the best performance with r2 being very close to 0.99. Decision Tree has the poorest performance with a r2_score of 0.908.

The reason that Random Forest is about 0.02 higher in r2_score than linear approach is because there are a lot of categorical features in the dataset that are more meaningful in decision making instead of numeric significance.

The negative mean absolute error from sklearn.metrics was also used to examine model performance. The scores are listed in the table above. It can be seen Decision Trees still tops the perforce. Scoring for Linear Regression is, however, exceedingly low. This tell us selection of metrics has a great influence on ranking the performance of algorithms.

Feature Importance

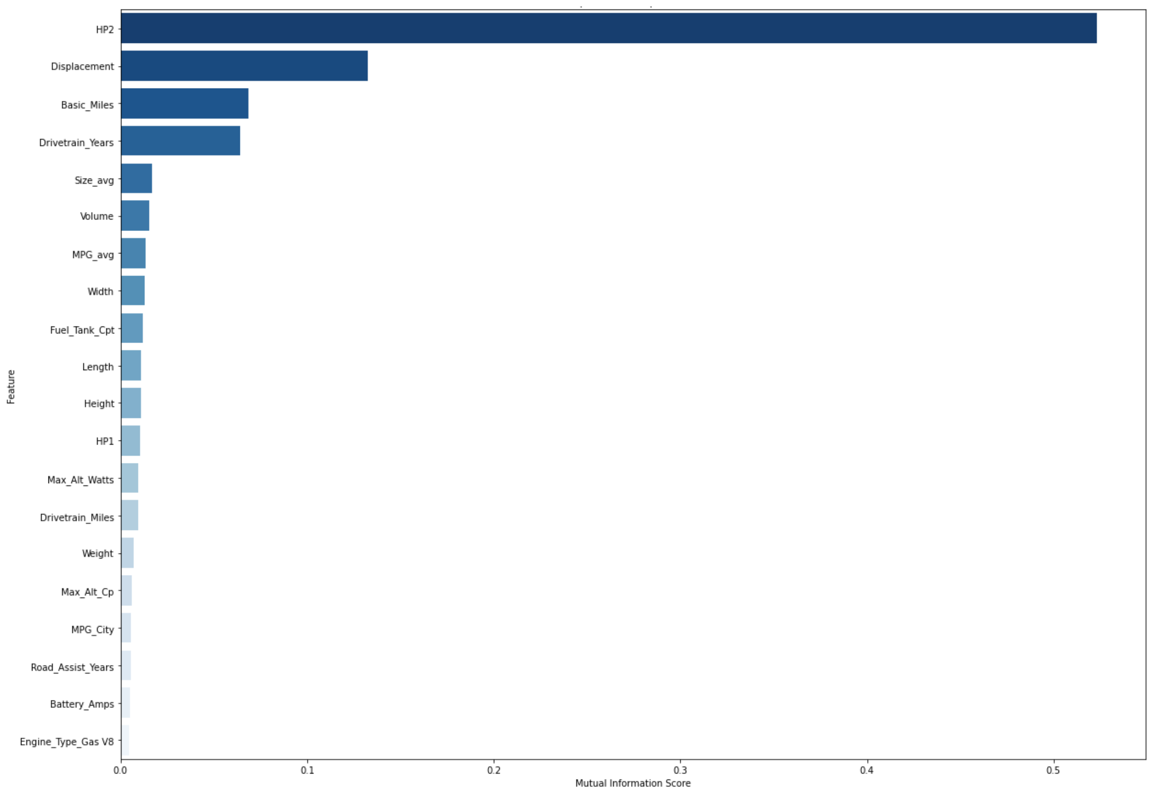
The feature importance results are shown in the table below. The bar graph shows top 20 features.

Rank	Feature	Importance
0	HP2	0.523067
1	Displacement	0.132579
2	Basic_Miles	0.068280

3	Drivetrain_Years	0.064147
4	Size_avg	0.016712
5	Volume	0.015565
6	MPG_avg	0.013185
7	Width	0.013044
8	Fuel_Tank_Cpt	0.011811
9	Length	0.011152
10	Height	0.011051
11	HP1	0.010560
12	Max_Alt_Watts	0.009645
13	Drivetrain_Miles	0.009586
14	Weight	0.007124
15	Max_Alt_Cp	0.005830
16	MPG_City	0.005474
17	Road_Assist_Years	0.005334
18	Battery_Amps	0.004968
19	Engine_Type_Gas V8	0.004681
20	Trans_Speed	0.003093
21	Road_Assist_Miles	0.002801
22	Model_Lamborghini Aventador	0.002461
23	Trailer_cpt	0.002054
24	Parking_Aid_No	0.001821
25	Num_safety	0.001802
26	Parking_Aid_Yes	0.001720
27	Psg_Cp	0.001657
28	Maint_Miles	0.001538
29	Maint_Year	0.001379
30	Fog_Lamps_Yes	0.001372

31	Backup_Camera_Yes	0.001357
32	Fog_Lamps_No	0.001270
33	Psg_Doors	0.001248
34	Corrosion_Years	0.001235
35	Model_Maserati Quattroporte	0.001120
36	Model_Porsche 911	0.001021
37	Backup_Camera_No	0.001017
38	MPG_Hwy	0.000987
39	Engine_Type_Twin Turbo Premium Unleaded V-12	0.000974
40	Model_Bentley Continental GT	0.000779
41	Basic_Year	0.000681
42	Night_Vision_Yes	0.000676
43	Front_Wheel_Mat_Steel	0.000661
44	Rear_Wheel_Mat_Steel	0.000642
45	Trac_Control_No	0.000642
46	Engine_Type_Premium Unleaded V-12	0.000615
47	Engine_Type_Gas V12	0.000596
48	Model_Porsche Panamera	0.000574
49	Susp_Front2_Double Wishbone	0.000553

Top 20 Features



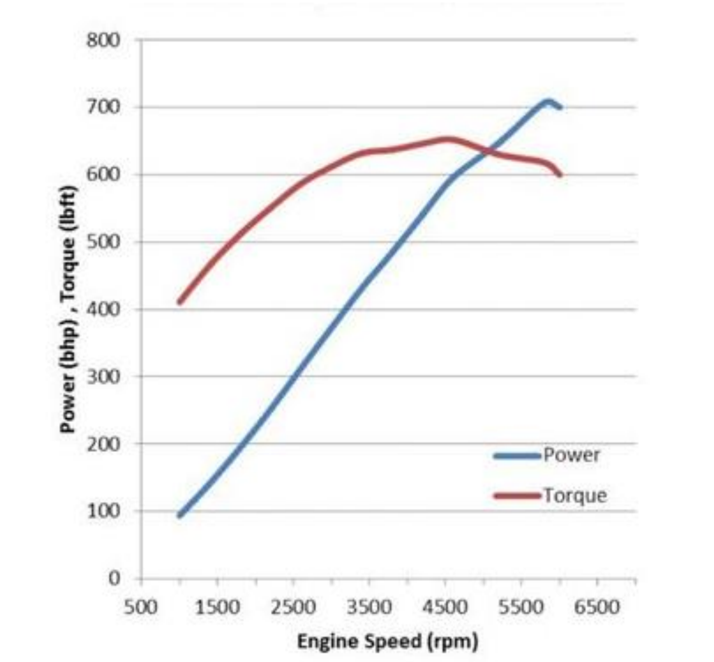
The HP2 (torque spec) has the highest importance, which is at least 4 times higher than the rest. The 2nd highest is Displacement, which is at least 2 times higher than the rest. These top 2 features are all specs related to car engines. The engine is the heart of a car that is the most dominating part for the major performance of car for example the car's lifetime, speed, driving smoothness, horsepower, fuel efficiency, etc. So, their influence on car price should be greater than other features.

Basic_Miles and Driverain_Years are in the warranty category and are surprisingly in the top ranking. The domain knowledge is required to explain it.

Size_avg, Volume, Width, Length, and Height are specs for car's dimensions. Their feature importance is somewhat close to each other and ranges from 0.011 to 0.016. This again proves the Decision Tree is an intelligent ensemble learning method.

HP1 is the horse power. Its feature importance is 0.01 that is 52 times lower than that of torque spec (HP2). Careful examination of two series of the data reveal HP1/HP2 ratio has a wide range from 0.5 to 10.8, in stead of being a constant. This is probably because the car manufacturers are not adopting the same engine speed to obtain horsepower and torque specs. The following figure helps to understand that the horsepower follows a nearly linear relationship with engine speed, however, the torque doesn't.

Horsepower and Torque Varied with Engine Speed



https://www.caranddriver.com/news/a15347872/horsepower-vs-torque-whats-the-difference/

Weight ranks 14th. Miles per gallon MPG_City ranks 16th. It is common knowledge that a heavier car consumes proportionally higher amount of gasoline per mile. This model precisely reveals the close relationship between the two specs.

Some car models like Model_Lamborghini Aventador, Model_Maserati Quattroporte, and Model_Bentley Continental GT are in top 50 important features and are more important than technical specs. For example, the feature importance of Model_Lamborghini Aventador is greater than that of number of passenger doors. This is probably because these are luxury cars that have a lot of weight on price tag.

Multicollinearity

Although correlation heatmap help find multicollinearity, it is limited to bivariate relationship between the independent variables. VIF, variance_inflation_factor, instead show the correlation of a variable with all other variables. High multicollinearity is indicated by VIF exceeding 5 or 10.

Package statsmodels.stats was used to calculate vif. The results are shown below.

VIF calculated with all variable

	variables	VIF
0	Volume	5
1	Corrosion_Years	13
2	Corrosion_Miles	13
3	Num_safety	18
4	Psg_Cp	20
5	Trans_Speed	22
6	Psg_Doors	23
7	Size_avg	23
8	Trailer_cpt	24
9	Battery_Amps	25
10	Max_Alt_Cp	46
11	Drivetrain_Years	53
12	Max_Alt_Watts	54
13	HP1	55
14	Road_Assist_Miles	58
15	Drivetrain_Miles	60
16	Displacement	66
17	HP2	76
18	Fuel_Tank_Cpt	86
19	Basic_Miles	87
20	Road_Assist_Years	101
21	Basic_Year	179
22	Maint_Year	208
23	Maint_Miles	209
24	Weight	264
25	Height	264
26	Length	356
27	MPG_Hwy	386
28	MPG_City	453
29	MPG_avg	847
30	Width	860

VIF calculated after removing 2 variables

	variables	VIF
0	Volume	4
1	Corrosion_Miles	12
2	Corrosion_Years	12
3	Num_safety	18
4	Psg_Cp	20
5	Trailer_cpt	21
6	Trans_Speed	22
7	Psg_Doors	22
8	Size_avg	23
9	Battery_Amps	24
10	Max_Alt_Cp	45
11	Drivetrain_Years	52
12	Max_Alt_Watts	53
13	HP1	54
14	Road_Assist_Miles	57
15	Drivetrain_Miles	60
16	Displacement	64
17	HP2	74
18	Fuel_Tank_Cpt	85
19	Basic_Miles	86
20	Road_Assist_Years	101
21	MPG_City	140
22	Height	170
23	Basic_Year	173
24	Maint_Year	201
25	Maint_Miles	207
26	MPG_Hwy	226
27	Weight	259
28	Length	274

It appears that VIF is all very high in overall, considering 10 as threshold. MPG_avg, Width are at least about two times higher than all the other variables. VIF results from removing the two variables improve the correlations of the rest variables.

The effect of the two variables with highest VIF on the models' performance are evaluated.

Scoring with all variables

Models	r2_score	negative mean absolute error

Random Forest	0.986	-1443
Linear regression	0.968	-1002483
Lasso Regression	0.967	-3262
Ridge Regression	0.964	-3525
Decision Tree	0.908	-5470

Scoring after removing two variables ('MPG_avg', 'Width')

Models	r2_score	negative mean absolute error
Random Forest	0.985	-1529
Linear regression	0.964	-3131
Lasso Regression	0.963	-3311
Ridge Regression	0.960	-3566
Decision Tree	0.913	-5618

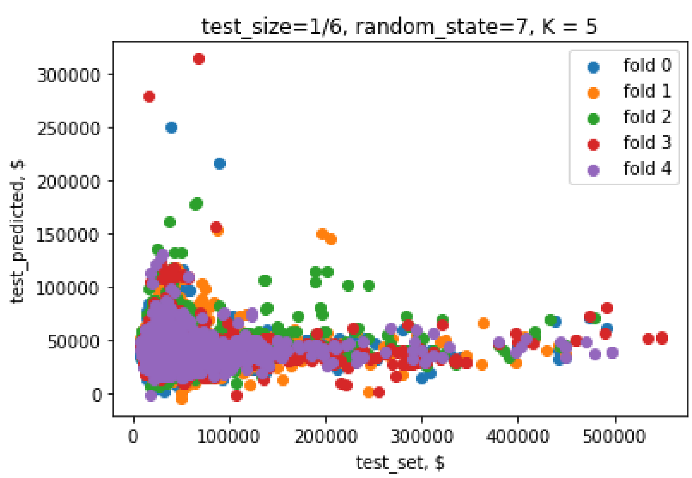
The models' performance actually slightly weakens after the two variables. It proves that multicollinearity may not necessarily have negative effect on model's predictability.

The high correlation between hp and displacement found in the EDA earlier doesn't suggest a bad apple neither.

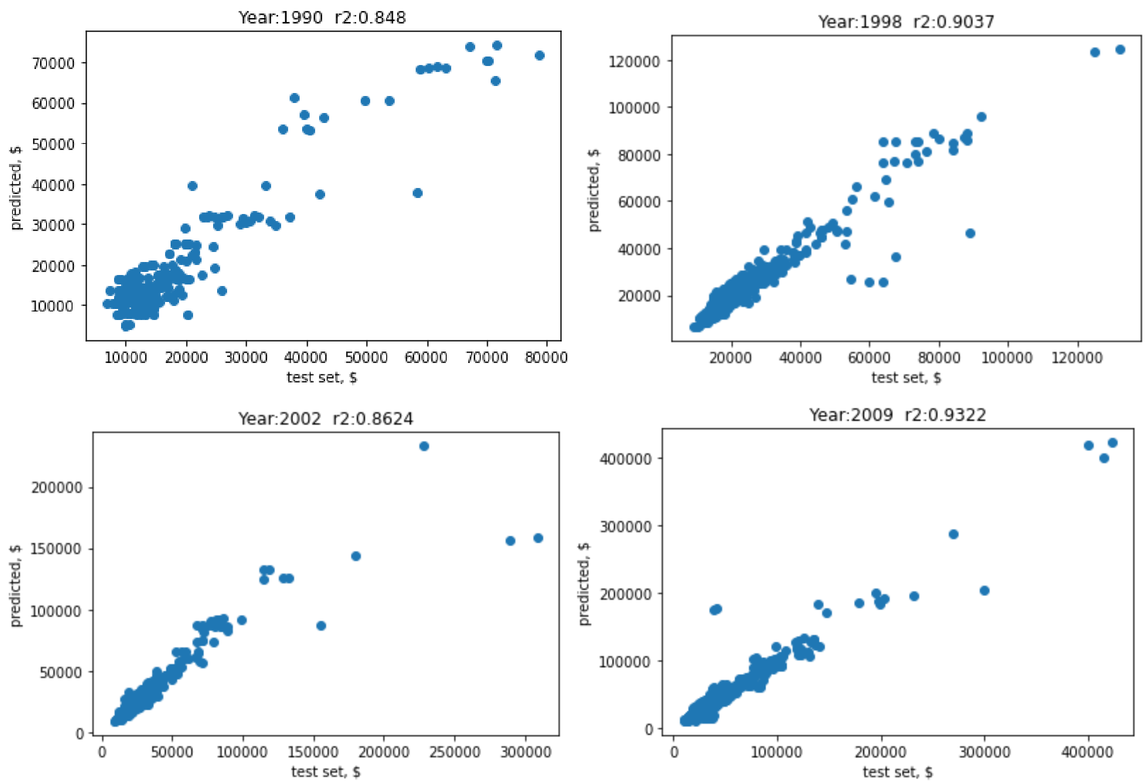
This seems to indicate that correlations may not be an effective tool for feature engineering.

Problems with Linear Regression

It was found that linear regression sometimes gave vary large negative scores. Using cross validation method, below is an example that shows the predication is very far from the test set.



This problem is most likely caused by the random split that certain cars that are in the test set but not in the train set. In order to solve this problem, Year was used as an integer. Set aside all the cars of one single year as test set, all the rest years' cars are used an train set. Shown below are the results from picking 5 years. R2 scores are also recorded. Therefore, by using choosing test set based on year, the problem with imbalanced dataset can be solved.



Year as Integer: Its Influence on Model Performance

Since the car prices usually rise each year, it is reasonable to use this variable as integer.

Models	r2_score	negative mean absolute error
Random Forest	0.985	-1691
Lasso Regression	0.962	-3398
Linear regression	0.961	-3200
Ridge Regression	0.958	-3671
Decision Tree	0.905	-5820

However, models performance remains somewhat similar to using year as category. Random Forest ranks 1st position that scores much higher than other models. Random Forest actually performs better when using year as categorical data type.

Conclusions

Tremendous efforts were made on data wrangling/cleaning. A small program was coded to successfully impute missing entries in the weight data. This imputation strategy relies on insightful domain knowledge.

Using hypothesis testing, it was found there is significant difference in the mean price between two low-end popular car models: Hyundai Accent and Honda Civic.

Exploratory data analysis was conducted to visualize missing values over all dataset, provide buying guide for low-income customers by extracting all lowly-priced car models and sorting in order. The pair plot and heatmap indicate a positive correlation between horsepower and engine displacement, which agrees well with physics and engineering principles. Linear line fit well the correlations between size and displacement, horsepower and MPG, price and horsepower, price and weight.

Imputation and one-hot encoding were done prior to apply various machine learning algorithms. We take advantage of machine learning algorithm called Missforest to automatically fill a large amount of missing values, which is one of the reasons we end up with very high predicting accuracy.

Five models were experimented including linear regression, ridge regression, lasso regression, decision trees and random forest. Except lower performance of decision trees, all the other models deliver very good r2 scores higher than 96%. The best model is random forest that scored close to 99%.

Feature importance analysis revealed the torque spec (HP1) and displacement are the most important factor determining the car prices. This result indicates the power of random forest because these two features are related to the heart of car: engine.

Influence of correlations by calculating the VIF indicates it may not be an effective tool for feature engineering.

Linear regression sometimes results in abnormally high r2 score. It was found the random split is the culprit because some of the car models may be in the test set but not in the train set. Modifying year to be numerical and setting aside one year's data as test set successfully solved the problem.

Future Work

- Acquire more domain knowledge for the purpose of feature engineering
    - remove unnecessary features

    - create new features
- Tune models hyperparameters to marginally improve performance
- Obtain data for newer car models to test the model
- Search for even better algorithms

Appendix 1

Features Index

Features	Full Name
Volume	Cargo Volume (ft^3)
Width	Width Max w/o mirrors (in)
Height	Height Overall (in)
Length	Length Overall (in)
Weight	Base Curb Weight (lbs)
Trailer_cpt	Maximum Trailering Capacity (lbs)
Fuel_Tank_Cpt	Fuel Tank Capacity Approx (gal)
MPG_avg	Fuel Economy Est-Combined (MPG)
MPG_City	EPA Fuel Economy Est - City (MPG)
MPG_Hwy	EPA Fuel Economy Est - Hwy (MPG)
Trans_Speed	Trans Type
Airbag_Front_Driver	Air Bag-Frontal-Driver
Airbag_Front_Psg	Air Bag-Frontal-Passenger
Airbag_Psg_Switch	Air Bag-Passenger Switch (On/Off)
Airbag_SBF	Air Bag-Side Body-Front
Airbag_SBR	Air Bag-Side Body-Rear
Airbag_SHF	Air Bag-Side Head-Front
Airbag_SHR	Air Bag-Side Head-Rear
Child_Door_Locks	Child Safety Rear Door Locks
Day_Lights	Daytime Running Lights
Trac_Control	Traction Control
Night_Vision	Night Vision
Rollover_Prt	Rollover Protection Bars
Fog_LampsParkingAid	Fog Lamps
Tire_P_Monitor	Tire Pressure Monitor
Backup_Camera	Back-Up Camera
Stability_Ctl	Stability Control
Susp_Front	Suspension Type - Front
Susp_Front2	Suspension Type - Front (Cont.)
Susp_Rear	Suspension Type - Rear
Susp_Rear2	Suspension Type - Rear (Cont.)
Cold Cranking Amps @ 0° F (Primary)	Battery_Amps
SAE Net Torque @ RPM	HP1
SAE Net Horsepower @ RPM	HP2
Engine Type	Engine_Type
Corrosion Years	Corrosion_Years
Corrosion Miles/km	Corrosion_Miles
EPA_Class	EPA Class
Body_Stype	Body Style
Front_Wheel_Mat	Front Wheel Material
Rear_Wheel_Mat	Rear Wheel Material
Psg_Cp	Passenger Capacity
Psg_Doors	Passenger Doors
Basic_Miles	Basic Miles/km
Basic_Year	Basic Years
Drivetrain_Miles	Drivetrain Miles/km
Drivetrain_Years	Drivetrain Years
Road_Assist_Miles	Roadside Assistance Miles/km
Road_Assist_Years	Roadside Assistance Years
Max_Alt_Cp	Maximum Alternator Capacity (amps)
Maint_Miles	Maintenance Miles/km
Maint_Year	Maintenance Years
Max_Alt_Watts	Maximum Alternator Watts
Other_Features	Other Features