NEXT_CAR!

Project Report

Practical Application Assignment 11.1

What drives the price of a Car?

For

Partial Requirements

Of

UC Berkeley Certificate in AI/ML Program

BH-PCMLAI-M01

AUGUST 22, 2022

SHAJI RAVINDRANATHAN
Section B

1 INTRODUCTION

New-vehicle affordability[1] in the USA has steadily worsened over the years. Latest data from Moody's analytics show that vehicle affordability worsened with increases in interest rates and vehicle prices outpacing income growth.

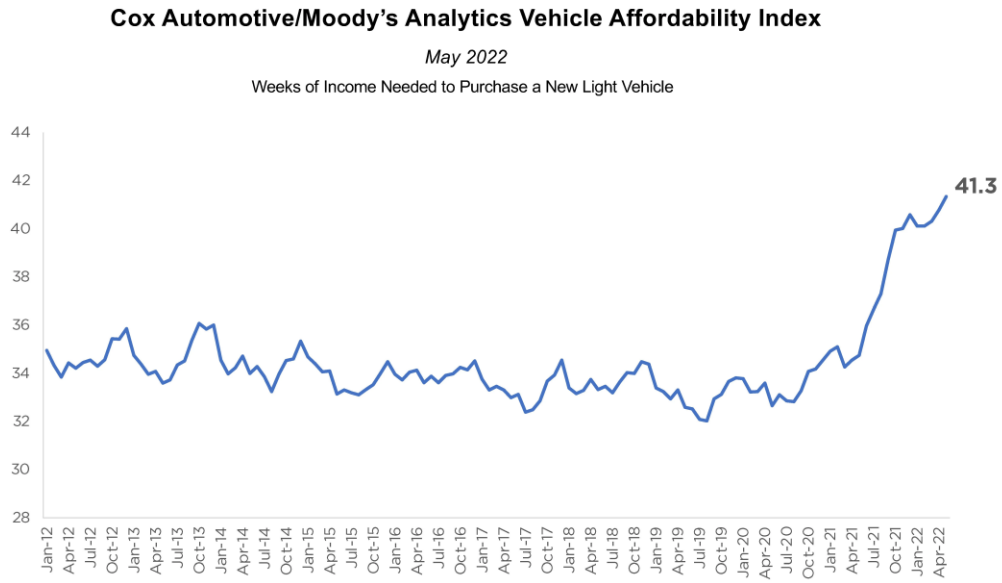


Figure 1 Moody's Analytics Vehicle Affordability Index

Autonation Inc. a leading car dealership reported in their second quarter 2022 operational summary that new vehicle revenue declined 14% compared to the year-ago period, while the used vehicle revenue increased 13% compared to the year-ago period[2].

Nationwide there is an uptick in demand for used cars. Approximately 40 million used vehicles are sold each year in the USA, according to Edmunds.com, an online automotive review site[3].

[1] https://www.coxautoinc.com/market-insights/may-2022-vai/

[2] https://investors.autonation.com/news-and-events/press-releases/press-release-details/2022/AutoNation-Reports-All-Time-Record-Quarterly-EPS/

[3] https://static.ed.edmunds-media.com/unversioned/img/industry-center/insights/2019-used-vehicle-outlook-report-final.pdf

2 BUSINESS UNDERSTANDING

A study from 2021 Cox Automotive Car Buying Journey[4], mentions that shoppers are looking for a fair car price not only for new vehicles, but also for used vehicles and the vehicles they currently own.

There is a reasonable expectation among consumers that as mileage increases, so does wear and tear. A potential purchaser would be less inclined to pay top dollar for a 200,000-mile car verses one with 30,000 miles. Condition is more subjective than mileage—a car dealer selling a reliable, accident-free car with paint scratches and surface rust might describe it as excellent, whereas most buyers might call it good to average—so it might be as important as mileage in assessing value.

A consumer or a used car dealer needs to be able to reasonably estimate the price of a used car based on its features.

Typical business questions a consumer or a car dealer will have is as follows:

What are the various factors involved in the pricing of a used car? Is the pricing dependent on

- Age and Mileage
- Condition of the vehicle
- Geographic Location of the vehicle
- Accident History
- Color of the vehicle
- Make/Options and Add Ons in the Vehicle (Type of Transmission/Fuel etc.)

2.1 Problem Definition

Given the various factors involved in a consumer's buying decision, can the price of a used car be predicted based in its attribute with reasonable accuracy?

2.2 Solution

This report outlines a used car pricing engine **(Next_Car!),** that is designed from the ground up to induce frictionless transaction experience between buyers and sellers. This used car price prediction system to determines the pricing of a used car using a variety of features and attributes. **Next_Car!** is a used car pricing tool to help car buyers and sellers talk about price realistic manner. This application is targeted at used car sellers (individual or used car dealers and individual buyers.

Target Audience: The primary audience for this report is used car dealers, however it can be used by all of the categories of users outlined below.

[4] COX: https://www.coxautoinc.com/wp-content/uploads/2022/01/2021-Car-Buyer-Journey-Study-Overview.pdf

Used Car Dealership: The primary target group for this study. For large segment of used car customers an automobile purchase is a very large purchase that they make once or a few times in their lifetime. A lot of consideration goes into the buying decision. Car pricing is one of the top considerations when they are planning to make a purchase. Consumers compare the prices of used cars before making a purchase. If a dealer understands the attributes that make the used car desirable, they are in a better position than their competition to increase demand for their offerings, via superior service.

Consumers:

Shoppers are looking for a fair car price not only for new vehicles, but also for used vehicles and the vehicles they currently own. Consumers can access this tool via paid or free comparison-shopping portals.

Online price comparison platforms: According to IBISWorld the market size, measured by revenue, of the Used Car Dealers industry is $146.5bn in 2022. Given the vast size of the industry any potential entrepreneur wanting to start an online price comparison service or existing used car dealership needs a good price prediction model. Existing pricing data coupled with Next_Car! Pricing model may help them in providing a better price prediction experience for their customers.

3 DATA UNDERSTANDING

The data used in this project was provided in the application 2 starter package. The original dataset is from Kaggle, that contained information on 3 million used cars. The provided dataset contains information on **4,26,880** cars to ensure speed of processing. It contains most all relevant information a Craigslist used car advertisement, provides on car sales like price, condition, manufacturer etc., for a total of 18 features associated with a used car listing.

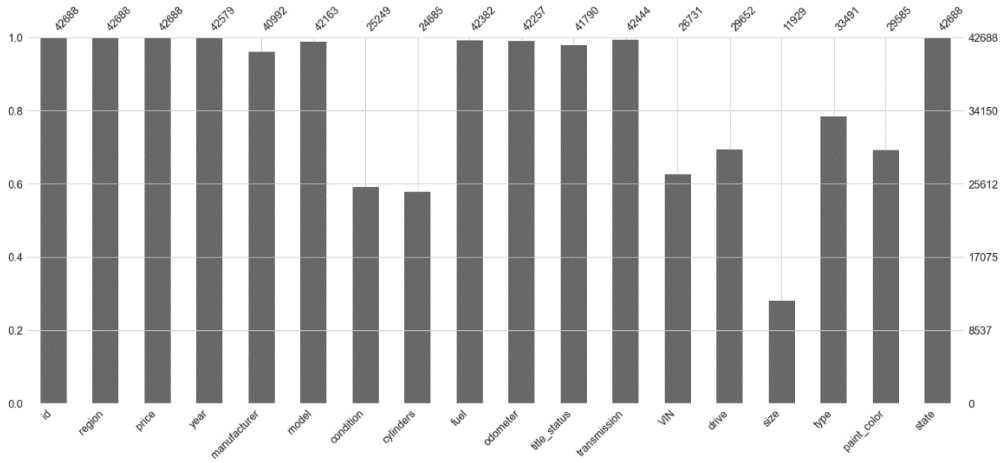


Figure 2 List of features in the vehicle's dataset

We can see from figure 2 which is visual representation of the dataset, the third column features the price. This is the feature we want to predict based on the other 17 features in the dataset. There are number of missing values in many of the feature columns, hence data cleaning is necessary.

	count	unique	top	freq
region	42688	403	columbus	381
manufacturer	40992	41	ford	7220
model	42163	8453	f-150	772
condition	25249	6	good	12114
cylinders	24685	8	6 cylinders	9285
fuel	42382	5	gas	35692
title_status	41790	6	clean	40489
transmission	42444	3	automatic	33772
VIN	26731	20795	1FTER1EH1LLA36301	31
drive	29652	3	4wd	13144
size	11929	4	full-size	6245
type	33491	13	sedan	8647
paint_color	29585	12	white	7955
state	42688	51	ca	5154

Figure 3 Summary Statistics

We can see from the figure 3 that Ford is by far the most popular vehicle listed for sale. 6-cylinder cars and white paint color are most popular and majority of the cars are sedans. California has the largest number of cars listed for sale which is not surprising as the state has the largest number of vehicles registered with the Department of Motor Vehicles.

4 DATA PREPARATION:

4.1 Data Cleaning:

The following steps were undertaken as a part of the data cleaning procedure:

1. Some features like advertisement identification number “id” and “VIN” which is the vehicle identification number has no bearing on the pricing of the vehicles. So those columns were dropped.
2. Next using the ‘missingno’ library and percentage of null data points for a feature were calculated. The rows with missing numbers were dropped.
3. In linear regression, outliers can greatly affect the regression (the slope, r-value, and r-squared). The features were next checked for outlier values that were dropped like so.

```
df2.drop(df2[(df2.price<500 )|( df2.price>28000)].index)["price"].describe()
```

```
count     2626.000000
mean     12448.051028
std       6662.748374
min        500.000000
25%       6995.000000
50%      10995.000000
75%      16995.000000
max      28000.000000
Name: price, dtype: float64
```

```
stats.describe(df2.drop(df2[(df2.price<500 )|( df2.price>28000)].index)["price"])
```

```
DescribeResult(nobs=2626, minmax=(500, 28000), mean=12448.051028179741, variance=44392215.89834752, skewness=0.556970241750928
7, kurtosis=-0.6256984484794295)
```

Figure 4 Removing Outliers

4. Figure 4 shows the cleaned dataset with all the missing data addressed. To speed up the machine learning process we are initially taking a subset of the samples like so:

```
df = originaldf.sample(frac=0.10, random_state = 1)
df.head()
```

Sampling of the dataset was done to work with a more manageable number of records.

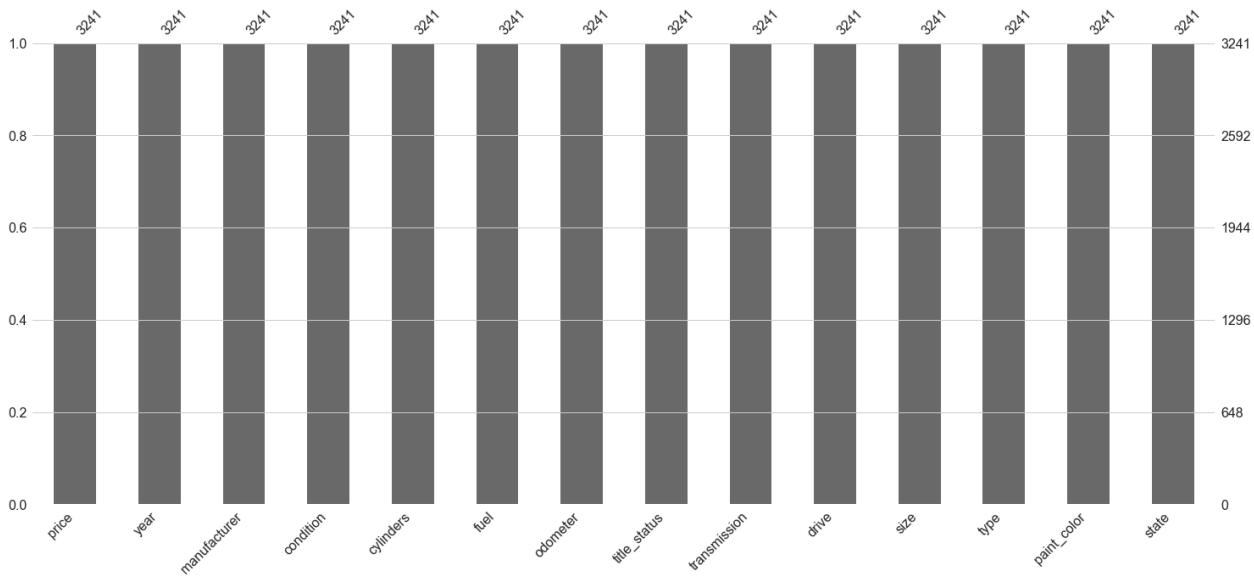


Figure 5 Cleaned Dataset for Machine Learning

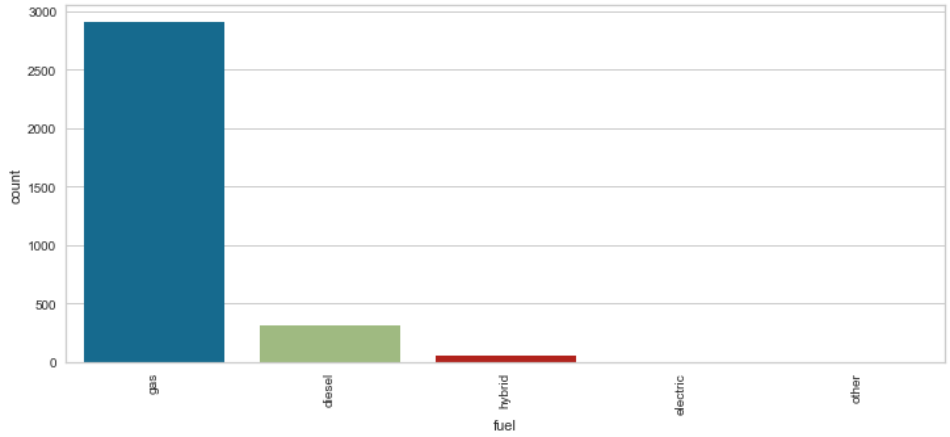
4.2 Exploratory Data Analysis on the clean data

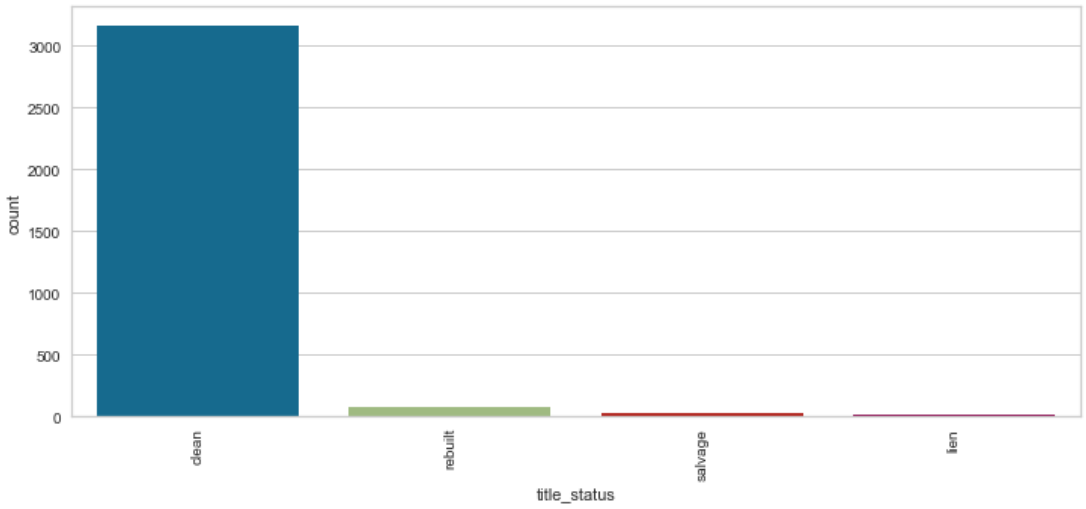
Figures 6 a through g show the basic trends that can be observed in the cleansed dataset.

- Gasoline fueled cars comprise most of the available vehicles for sale.
- Cars with clean titles are mostly likely to be put on sale.

Figure 6 a -g Exploratory Data Analysis

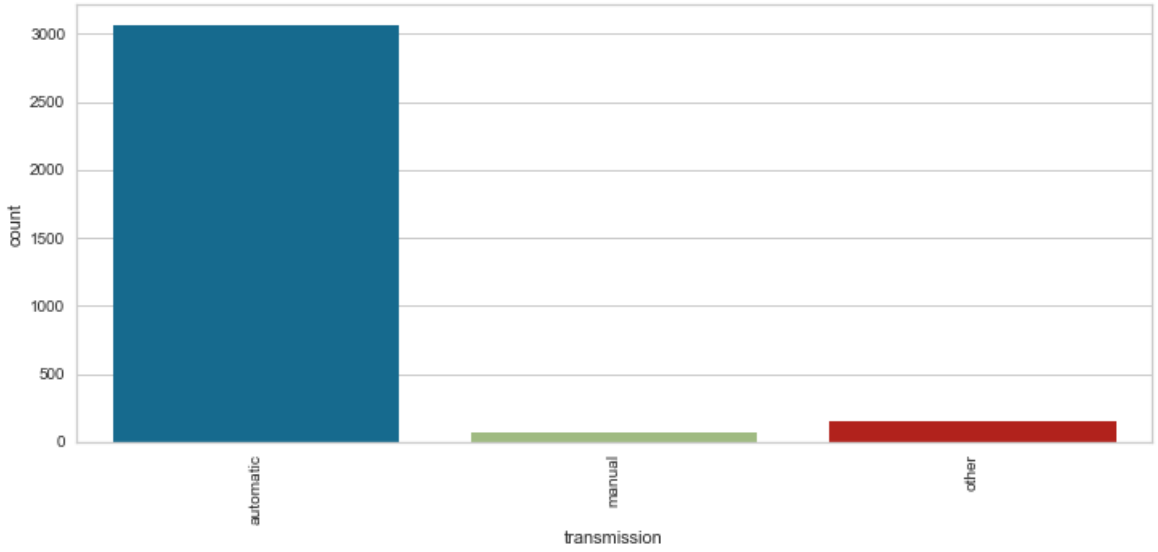
(6a)

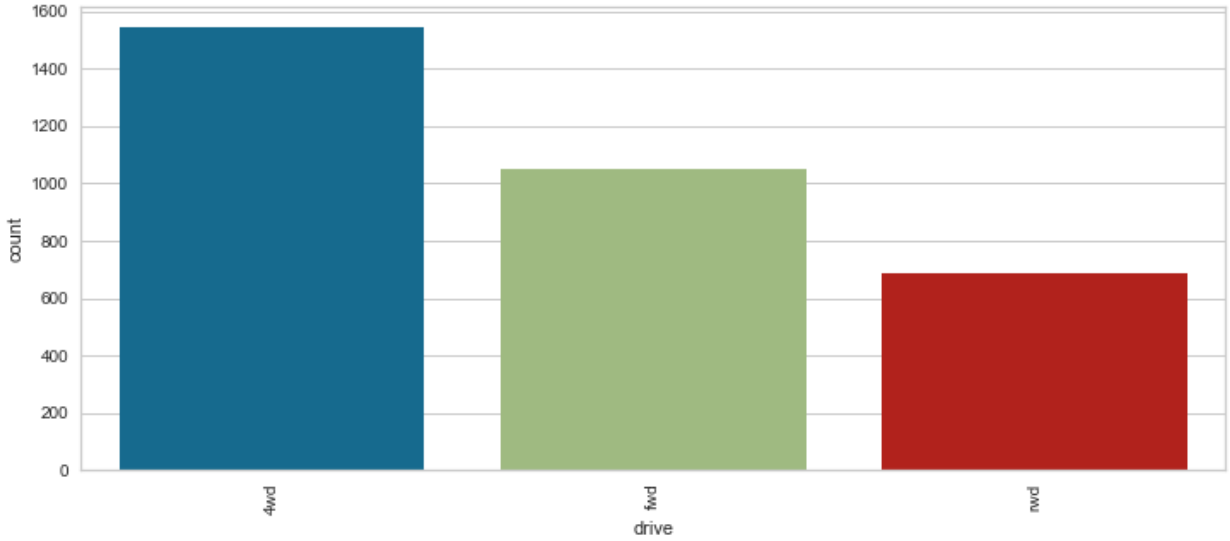




(6b)

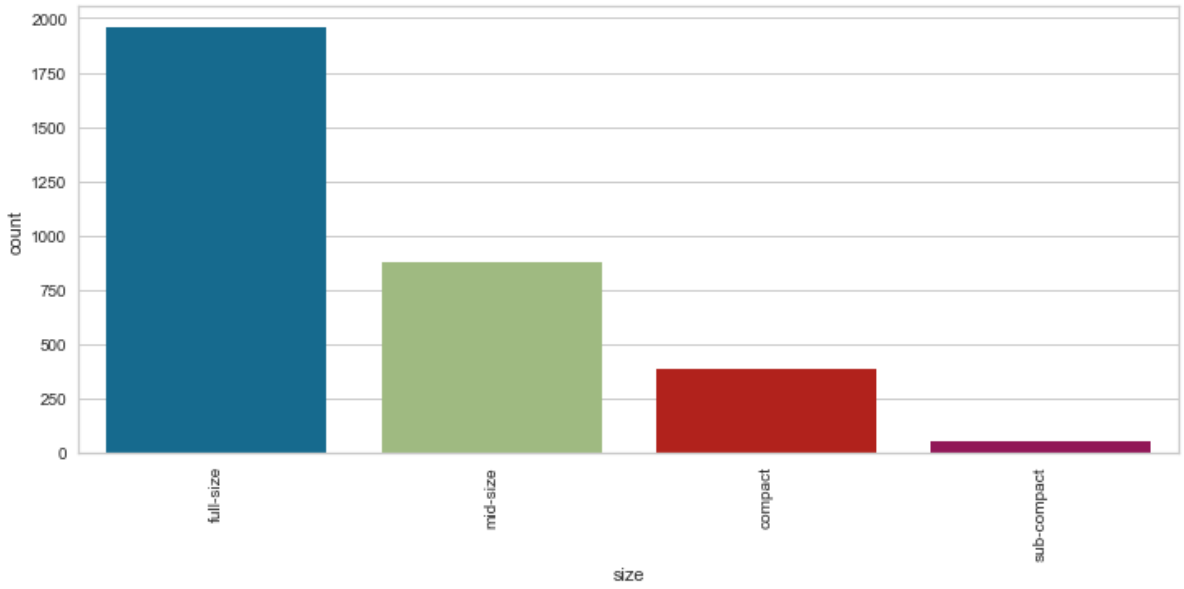
- Automatic Transmission with four-wheel drive constitute majority of the listings **(6c &6d)**

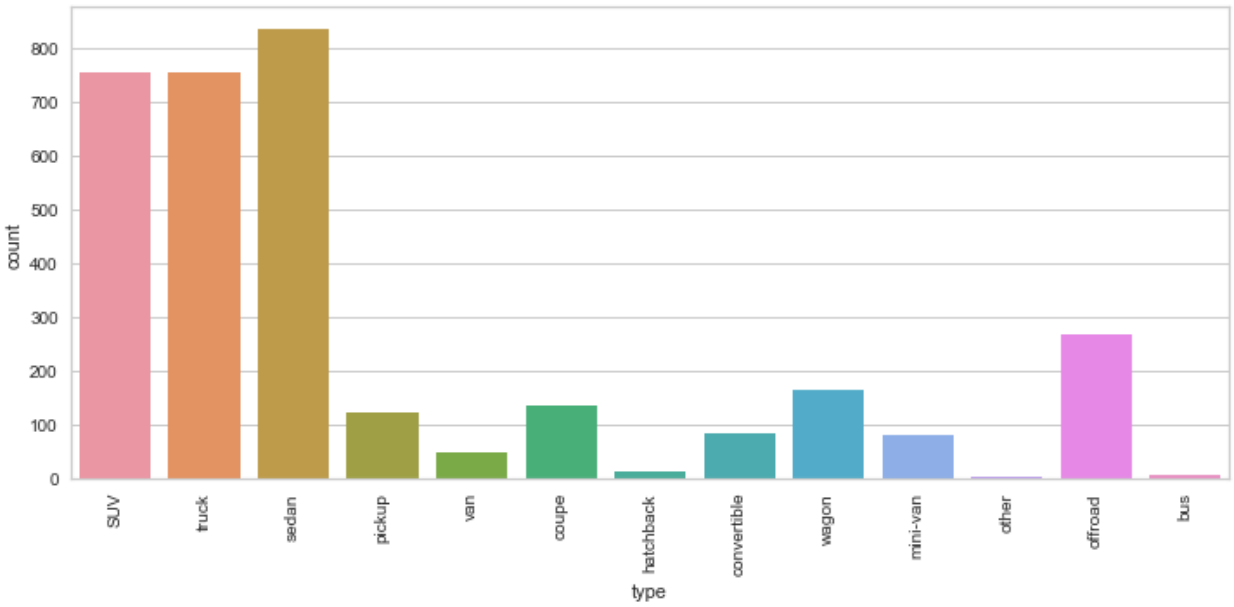




(6e & 6f)

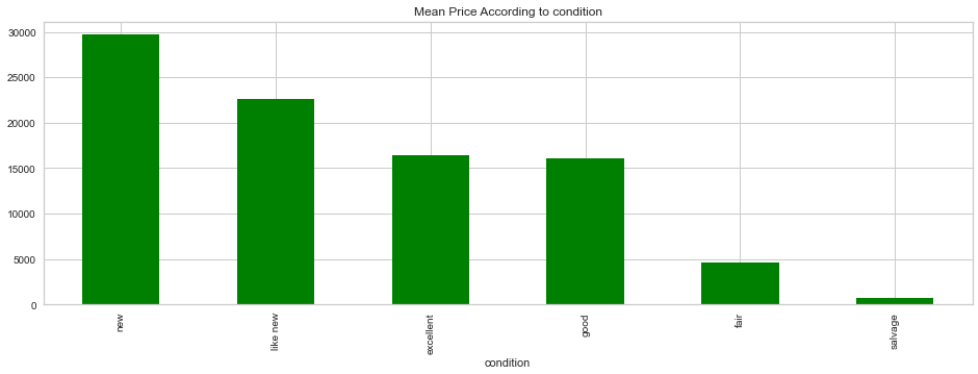
- Full size sedans seem to be popular vehicle choices for most customers



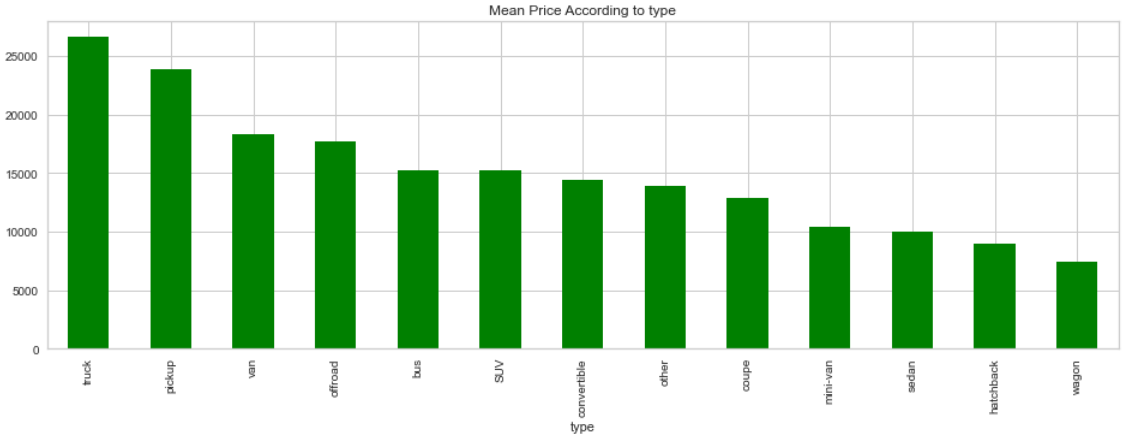


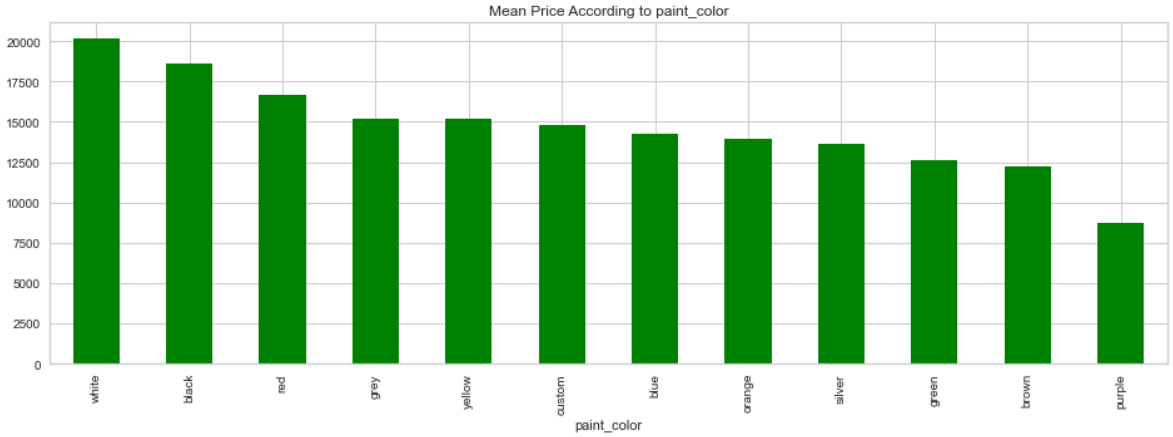
- **Pricing Trends:** White color seems to fetch the highest price, new condition cars with automatic transmission. Likewise diesel powered vehicles had the highest pricing among the cars available for sale.

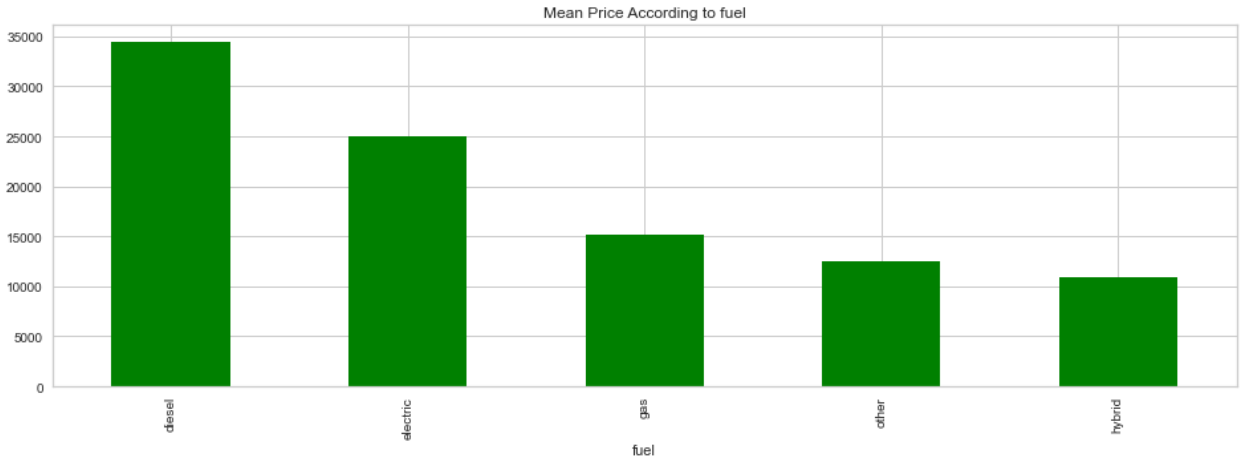
(6g,h,i)



(6j,k,l)







- **Geographic trends:** California had the largest % of care for sale and Ford was the brand that was on sale the most. Most of the used cars were listed for approximately less than $20,000.

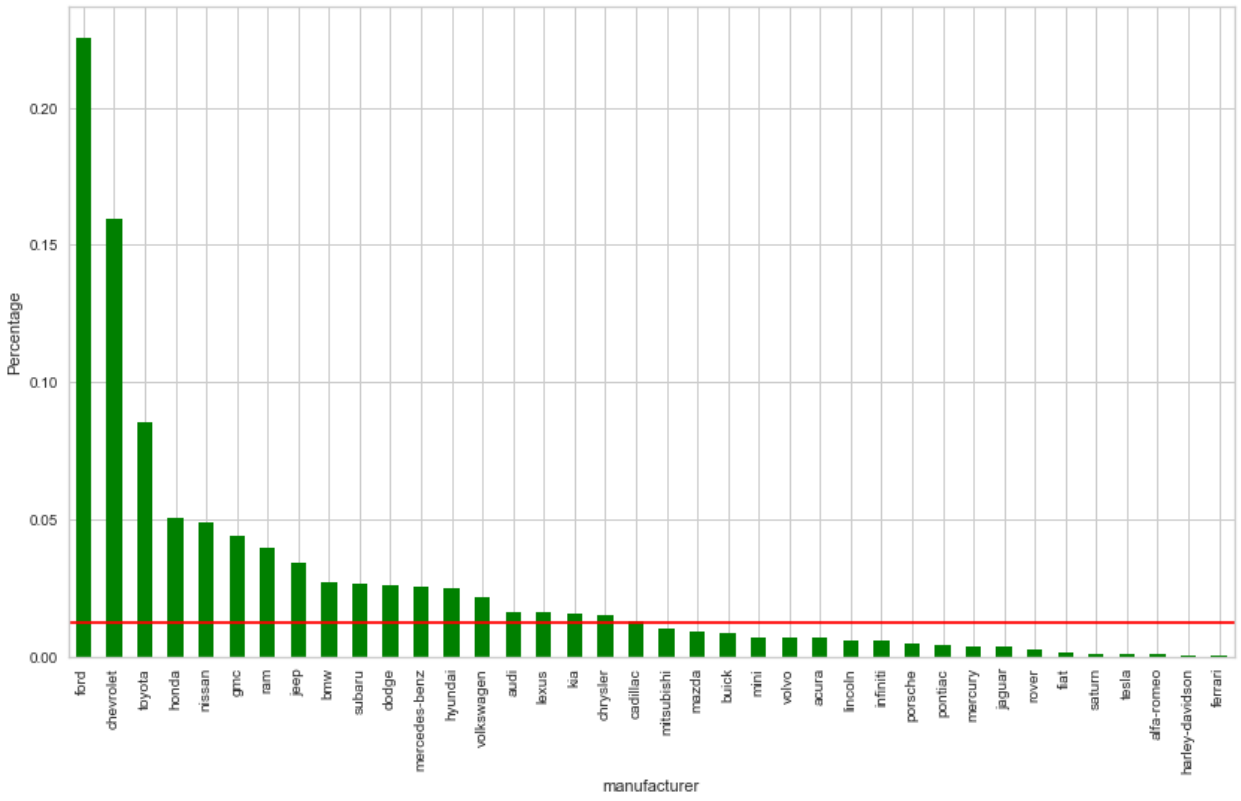
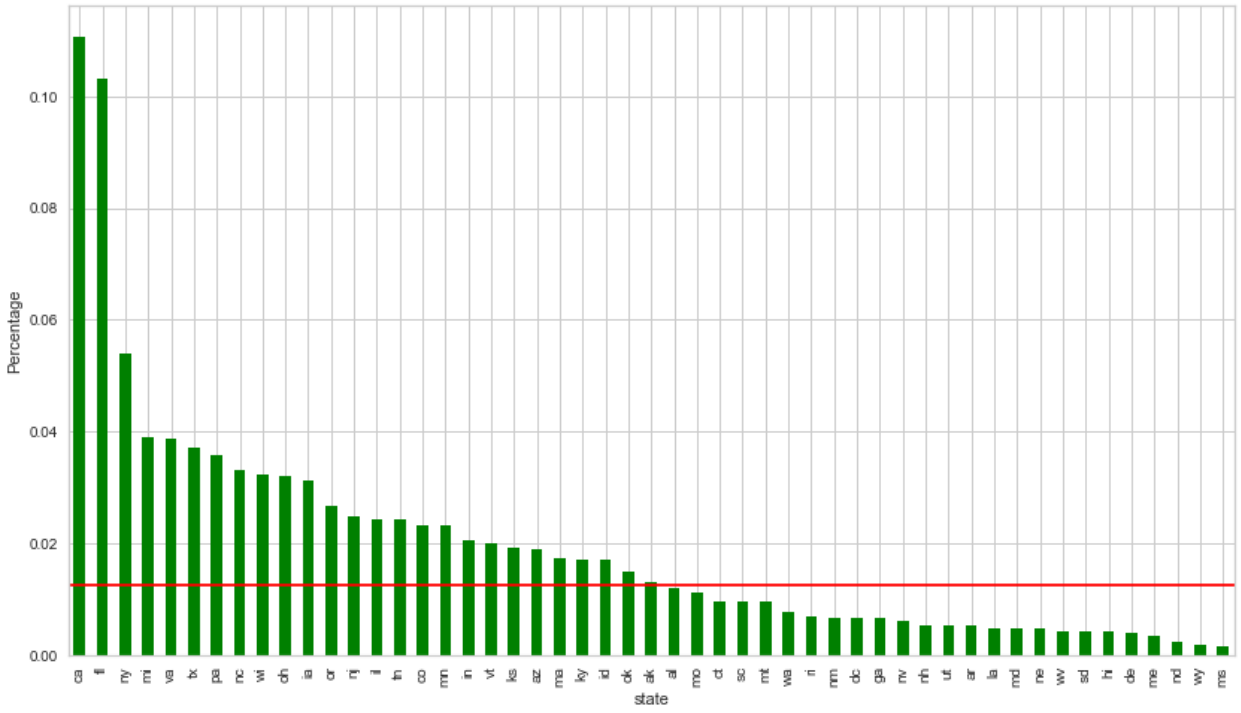

Percentage
0.20
0.15
0.10
0.05
0.00
ford
chevrolet
toyota
honda
nissan
gmc
ram
jeep
bmw
subaru
dodge
mercedes-benz
hyundai
volkswagen
audi
lexus
kia
chrysler
cadillac
mitsubishi
mazda
buick
mini
volvo
acura
lincoln
infiniti
porsche
pontiac
mercury
jaguar
rover
fiat
saturn
tesla
alfa-romeo
harley-davidson
ferrari
manufacturer

Figure (6, l,m)


Percentage
0.10
0.08
0.06
0.04
0.02
0.00
ca
fl
ny
mi
va
tx
pa
nc
wi
oh
ia
or
nj
il
tn
co
mn
in
vt
ks
az
ma
ky
id
ok
ak
al
mo
ct
sc
mt
wa
ri
nm
dc
ga
nv
nh
ut
ar
la
md
ne
wv
sd
hi
de
me
nd
wy
ms
state

- Most popular used cars had odometer around 100k-150k range and there are many cars on the market in this range.
- Inferential statistical methods were not much helpful for analyzing this data as the dataset had only 3 numerical features among the 18 features. Measures like Pearson's correlation were not very helpful. The advertisement id feature was discarded as it does not have any bearing on the pricing.

	id	price	year	odometer
count	4.268800e+04	4.268800e+04	42579.000000	4.225700e+04
mean	7.311444e+09	1.503687e+05	2011.249419	9.980050e+04
std	4.507152e+06	1.697124e+07	9.387330	2.515770e+05
min	7.213844e+09	0.000000e+00	1901.000000	0.000000e+00
25%	7.308030e+09	5.888000e+03	2008.000000	3.761800e+04
50%	7.312538e+09	1.370000e+04	2014.000000	8.600000e+04
75%	7.315236e+09	2.599500e+04	2017.000000	1.339080e+05
max	7.317094e+09	3.009549e+09	2022.000000	1.000000e+07

- So various regression machine learning models were applied to this cleaned data and model performance was compared using the mean absolute error (MAE), mean squared error (MSE) and root mean squared error (RMSE).

5 DATA MODELING

5.1 Methodology used in solving the problem

5.1.1 **Training the data**. In this process, 20% of the data was split for the test data and 80% of the data was taken as train data.

5.1.2 **Scaling the Data**. The data is not normally distributed. To avoid machine learning model disregarding coefficients of features with low values the minmax scaler was applied.

5.1.3 Categorical Data Pre-processing the data

In the dataset, there are 18 predictors, only 3 of them are numerical variables while rest of them are categorical. In order to apply machine learning models, we need the features to be represented in a numerical format. Therefore, all non-numeric features were transformed into numerical form. **Feature-engine** a Python library with multiple transformers was used to engineer and select features to use in machine learning models. Feature-engine preserves Scikit-learn functionality with methods fit() and transform() to learn parameters from and then transform the data.

Feature-engine includes transformers for:

- Missing data imputation
- Categorical encoding
- Outlier capping or removal
- Variable transformation
- Variable creation
- Variable selection
- Datetime features
- Time series
- PreprocessingAn example of the transformation for manufacturer, state and title_status is shown in figure

```
In [87]: rare_encoder = rare_label.RareLabelEncoder(
             # minimum threshold value to be considered as a non-rare

             tol=0.0125,

             #minimum category count to regroup rare categories

             n_categories=10,

             # features to re-group

             variables=["manufacturer","state","title_status"]
         )

In [88]: rare_encoder.fit(X_train)

Out[88]: RareLabelEncoder
         RareLabelEncoder(tol=0.0125,
                          variables=['manufacturer', 'state', 'title_status'])
```

Figure 7 Label Encoding for Categorial Features

5.2 Models used to solve the problem

OLS, Ridge, Lasso and ElasticNet regression models were built using the transformed and scaled data set and the results were analyzed like so:

```
models=[]
models.append(LinearRegression())
models.append(Ridge(alpha=0.1, random_state=0))
models.append(Lasso(alpha = 0.1))
models.append(ElasticNet(alpha=0.1, l1_ratio=0.7))

r2_values_test = []
r2_values_train=[]
rmse_values_test=[]
mse_values_test=[]


for model in models:
    model_=model.fit(X_train_encoded,y_train)
    y_pred=model_.predict(X_test_encoded)

    r2_train=model_.score(X_train_encoded,y_train)
    r2_values_train.append(r2_train)

    r2 = model_.score(X_test_encoded,y_test)
    r2_values_test.append(r2)


    rmse_test=np.sqrt(mean_squared_error(y_test,y_pred))
    rmse_values_test.append(rmse_test)

    mse_test=mean_squared_error(y_test,y_pred)
    mse_values_test.append(mse_test)

result=pd.DataFrame(list(zip(r2_values_test,r2_values_train)),columns=["r2_score_test","r2_score_train"])
result["rmse_test"] =rmse_values_test
result["mse_test"]=mse_values_test
#result["model"]=["Linear","Ridge","Lasso","RandomForest","XGBoost","LGBM"]

result["model"]=["Linear","Ridge","Lasso","Elastic" ]

result=pd.DataFrame(result)
result.set_index('model')
```

5.3 Model Analysis

During the raw run of the model building exercise no parameters were tuned to get a rough idea on the best suited model for this problem domain. OLS, Ridge and Lasso showed very little difference in the performance.

:

model	r2_score_test	r2_score_train	rmse_test	mse_test
Linear	-1.550315	0.642083	19663.179610	3.866406e+08
Ridge	-1.357396	0.642076	18904.843405	3.573931e+08
Lasso	-1.514408	0.642082	19524.263486	3.811969e+08
Elastic	0.330066	0.612717	10077.961180	1.015653e+08

As next step the OLS regression was chosen and parameter selection was done using GridSearch.

In addition to manually writing method for feature importance, I used a package called 'Yellowbrick' to visually plot feature importance. This can be viewed in the notebook output.

An elaborate pipeline was created for developing the model parameter selection as shown in the figure below:

```
Fitting 5 folds for each of 20 candidates, totalling 100 fits
```

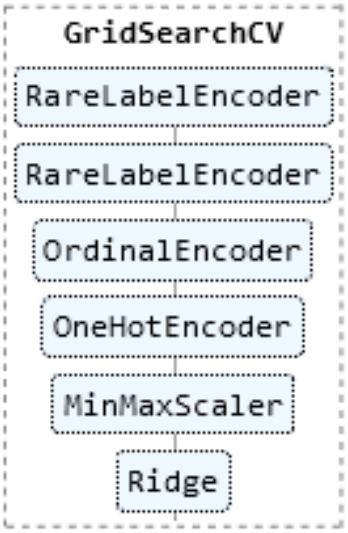


Figure 8 Model Building Pipeline with GridSearch for automatic best parameter selection

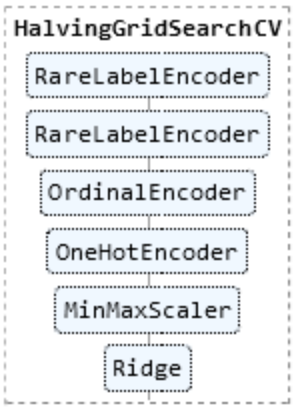
Since the number of features are very high and the parameter selection takes a lot of time even with all the cores allocated to GridSearch. After consulting the Scikit learn documentation I tried to improve the search performance by using an experimental feature called Searching for optimal parameters with successive halving[5].

The Gridsearch saw a marked improvement in performance. Conventional Gridsearch took more than an hour, whereas the HalvingSearch crunched through under 5 minutes. Figure 9 shows the modified datapipeline.

[5] Successive Halving GridSearch speed up: Reference: https://scikit-learn.org/stable/modules/grid_search.html

```
min_resources_: 32
max_resources_: 2592
aggressive_elimination: False
factor: 3
----------
iter: 0
n_candidates: 240
n_resources: 32
Fitting 5 folds for each of 240 candidates, totalling 1200 fits
----------
iter: 1
n_candidates: 80
n_resources: 96
Fitting 5 folds for each of 80 candidates, totalling 400 fits
----------
iter: 2
n_candidates: 27
n_resources: 288
Fitting 5 folds for each of 27 candidates, totalling 135 fits
----------
iter: 3
n_candidates: 9
n_resources: 864
Fitting 5 folds for each of 9 candidates, totalling 45 fits
----------
iter: 4
n_candidates: 3
n_resources: 2592
Fitting 5 folds for each of 3 candidates, totalling 15 fits
```

[187]:



[188]:
```
ridge_halving_grid_search.best_params_
```

[188]:
```
{'ridgemodel__alpha': 0.1,
 'ridgemodel__fit_intercept': True,
 'ridgemodel__solver': 'sag'}
```

Figure 9 Halving Grid Search for boosting model building performance

The model performance was boosted from 64% to 68%. The best score for Ridge regression from halving grid search was: 0.682. This is likely to improve with additional tweaks in the pipeline by adding a Polynomial stage of degree 2.

6 VALIDATION

```
print(("Best score for Ridge regression from halving grid search: %.3f" % ridge_halving_best_model.score(X_train, y_train))
```

```
Best score for Ridge regression from halving grid search: 0.671
```

```
n [188]: ridge_halving_grid_search.best_params_

ut[188]: {'ridgemodel__alpha': 0.1,
          'ridgemodel__fit_intercept': True,
          'ridgemodel__solver': 'sag'}
```

6.1 Predictions

A back end function was developed to predict the results using the ridge regression model. This backend function called by a web app to retrieve and present the price to the end user.

```
def predicted_price(year,manufacturer,condition,cylinders,fuel,odometer,title_status,transmission,drive,size,type,paint_color,sta

    vehicle={'price':[int(float(0.00))],'year':[(2022-int(float(year)))],'manufacturer':[manufacturer],'condition':[condition],'c

    vehicle=pd.DataFrame(vehicle)

    X=vehicle.drop(columns=["price"])


    used_car_price=ridge_halving_best_model.predict(X)

    used_car_price=used_car_price[0].round(2)

    print("The Estimated price for this used car is: $", used_car_price)
    return
```

The following is the predictions made by method when invoked using the jupyter notebook.

```
8]: predicted_price(2018,'chevrolet','like new','8 cylinders','gas','29051','clean','manual','rwd ','full-size','coupe','red','nc')
    The Estimated price for this used car is: $ 19518.22

9]: predicted_price(2014,'toyota','like new','4 cylinders','gas','25000','clean','automatic','rwd ','full-size','coupe','red','ca')
    The Estimated price for this used car is: $ 16573.2

0]: predicted_price(2006,'toyota','good','4 cylinders','gas','50000','clean','automatic','rwd ','full-size','coupe','red','pa')
    The Estimated price for this used car is: $ 12978.86
```

Ideally this would be the model part of a model-view-controller webapp. The car pricing estimates using different criteria show that the model is working as per the current model performance.

7 CONCLUSION:

Ridge Regression gave the best performance of all the regression models tried. year, odometer, make, drive, fuel, manufacturer, cylinders are the features that have the most significance in deciding the price of a used car. Location and paint color are other factors that come into play in deciding the price of a used car.

8 LIMITATIONS OF THE CURRENT PROJECT AND SCOPE FOR FUTURE IMPROVEMENT:

1. This project considered standard variations of regression models. Newer and faster techniques like Extreme Gradient Boosting (XGBoost) provides an efficient and effective implementation of the gradient boosting algorithm that can be used for regression predictive modeling. Other regressors we should consider is the LightGBM Classifier and Regressor as it has been known to do well with datasets with large amount of categorical variables.
2. Hyperparameter tuning coupled with the use of XGboost regressor should improve the current performance of this model.
3. Due to lack of time a suitable webapp frontend was not developed to make this user friendly and deployable
4. The total number of observations used was only 42688. This is a relatively small dataset for making a strong inference because I have used only a sampling of the dataset available to build a model with reasonable accuracy. More data should definitely provide robust and improved predictions.
5. Use of additional features could potentially improve model performance. Polynomial processing in the pipeline stage was not tried.
6. The data transformation stage of the pipeline could involve more transformers and imputers. Some of the numerical features could be converted to ordinal form. Different scaling transformer can be also be tried to boost the prediction power of models

9 REFERENCES:

1. Edmunds used vehicle outlook, Retreived from: https://static.ed.edmunds-media.com/unversioned/img/industry-center/insights/2019-used-vehicle-outlook-report-final.pdf
2. Cox Car Buyer study, Retrieved from: https://www.coxautoinc.com/wp-content/uploads/2022/01/2021-Car-Buyer-Journey-Study-Overview.pdf
3. Missingno library, Retreived from: https://github.com/ResidentMario/missingno
4. Feature Importance, Retreived from: https://www.scikit-yb.org/en/latest/api/model_selection/importances.html
5. Successive Halving GridSearch, Retrieved from: https://scikit-learn.org/stable/modules/grid_search.html