Predictive Analysis for Prices of Used Cars

Group: Ankit Jain, Casey Kan, Dido Chang, Sijia Li, Song Han

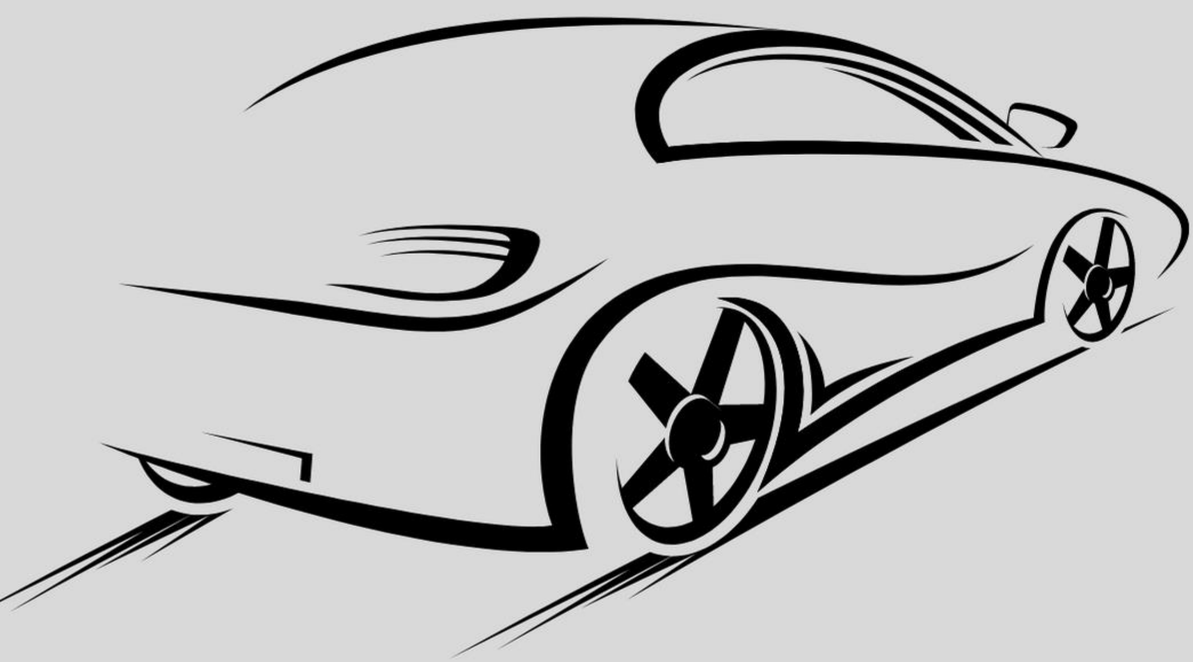


Table of Contents

Executive Summary

Buying used cars is very common in the US. According to Auto Remarketing, there was a 40.4 million sale for used cars in 2019. There are a lot of resources available online when searching for used cars. Yet, determining whether the listed price of a used car is appropriate is a challenging task, due to many factors that may affect a vehicle's price. The main focus of this project is to develop machine learning models that can accurately predict the price of a used car based on its features.

The dataset chosen for this project is from Kaggle highlights all sales of cars made on the popular selling site Craigslist since the year 1900. The dataset consists of over 400,000 rows of data, and has 25 columns representing the different attributes. Yet, we are only going to study data after year 2000 in order to make more updated and current results for our predictive model. Also, we will only pick out the attributes that provide significant insights to our predictive model. Some of the attributes we are planning to use are model, year, manufacturer, odometer, condition, etc. The analysis will be run by association rules, naive bayes and decision trees on Weka and some of the processing is done using Python Programming language on Jupyter Notebooks.

Our findings are intended to provide insight into future used car buyers and sellers to make appropriate predictions when listing and purchasing for used cars.

Business Ideas

Deciding whether a used car is worth the listed prices may be difficult when we simply look at the online listings. There are several factors such as model, year, odometer, manufacturer etc. that can influence the actual worth of the car. Moreover, from the seller perspective, it is also difficult to estimate the appropriate price for a used car. Based on the existing data, we aim to make use of machine learning models to help predict the used car prices.

Thus, by understanding what features of the car can affect the car price, we would like to investigate the following questions,

- What is the degree of influence of different attributes towards the class - price?
- What recommendations can we provide to future buyers when looking for cars that they may possibly buy?
- What recommendations can we provide to future sellers when making appropriate quotes for the cars they sell?

Data Description

The original dataset consists of over 400,000 rows of data, and has 25 columns representing the different attributes. Yet, we have removed all the missing values and trimmed down the number of variables since some of them do not add any importance to our predictive model.

The cleaned final dataset consists of 13 variables and they are shown below with the description of each attribute we used:

1. Price - The price (in USD) listed for the vehicle
2. Year - The year model of the vehicle

3. Manufacturer - The manufacturer of the vehicle
4. Condition - The condition of the vehicle
5. Cylinder - The number of cylinders of the vehicle
6. Fuel - The type of fuel of the vehicle
7. Odometer - The miles traveled by the vehicle
8. Title status - The title status of the vehicle
9. Transmission - The type of transmission of the vehicle
10. Drive - The drive type of the vehicle
11. Size - The size of the vehicle
12. Type - The type of drivetrain of the vehicle
13. Paint color - The color of the vehicle

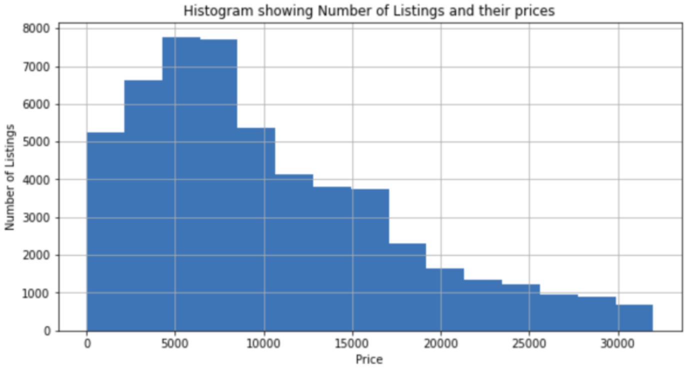
Price and odometer are continuous variables, whereas the rest of the variables are categorical variables.

Preparation and Preprocessing of Data

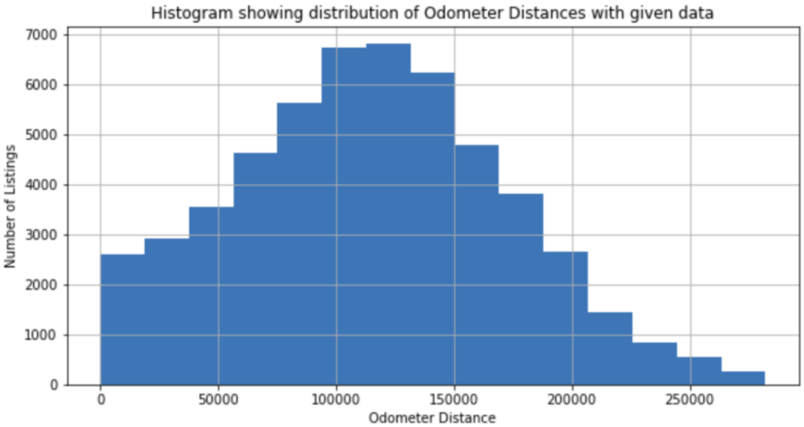
- Our initial raw data consisted of 25 columns and 423857 rows.
- We first removed the columns which were unique to each row such as id, url, region_url, image_url, vin, description, model. After doing an initial and extremely basic correlation analysis, we also noticed that region, county, lat, long were not significant variables and essentially created noise in the data which could contribute to reduction in model accuracies for predicting price, so we removed those columns as well.
- We then removed the rows of data which had missing values in order to get a full and complete dataset and get rid of any potential inconsistencies in the data for analysis.
- After that, we removed all the rows which had outlier values for the odometer and price attributes which were continuous in nature.
- Data dated before the year 2000 seemed to mostly lie in outliers and also were extremely scarce in the dataset, hence we decided to remove those as well and only included rows of data which had the year attribute value of the year 2000 or after. We also changed the numeric attribute of the year to nominal.
- After all of this preprocessing, we ended up with 13 columns and 53390 rows of data.
- We essentially had two versions of our data:
  - Version 1: Price and Odometer Variables are kept continuous
  - Version 2: Price and Odometer Variables are a discretized made into nominal attributes
- For Version 2 of the data, we tried out different numbers of bins for the two continuous variables, with both equal and unequal frequency for the bins, and we arrived at the following:
  - 3 bins were ideal for both attributes to gain a good overview of the data, and adding or reducing the bins made the model either too simple or increased model inaccuracy.
  - For both decision trees and naive bayes, the best accuracies were produced when the predictive model for the price was made when the price attributed was binned into 3 bins of equal size, not necessarily with equal frequency. In the case of the odometer attribute, most value was gained when all the 3 bins had an equal weight, i.e when the bins were of almost equal frequency, not necessarily of the same size.

  - For association rules, with non-equal frequency bins, one of the values was never predicted with the threshold for Lift set by us, so we decided to do cases where it was once split with equal frequency bins and once without.

- The distribution of the prices of the car seems to be left skewing showing that there aren't a lot of rows present for the higher price ranges, which may cause some inaccuracies in modelling. Due to this uneven distribution, it would make sense to use bins of almost equal frequencies so the bins itself are closer in weight and each bin almost has an equal probability of occurring in the dataset. This could also be possible due to the fact that in general when selling used cars, people don't try to price their cars too high in fear of it not being sold, so that bias is present in pricing.



- In case of the odometer distances, there seems to be lesser skewness compared to the price attribute, so weights of bins of unequal sizes may be similar, but it might be better to use bins of equal frequencies to in turn bring in equalized weight of each odometer range value.



<u>Unsupervised Learning</u>

<u>Association Rules</u>

Before running any supervised learning methods on our data, we wanted to understand the underlying patterns present in the data through unsupervised learning methods, and we decided to go for association rules as it can show how certain attributes may potentially serve as good predictors based on the lift of the rule. Though it may not show causation, the patterns with co-occurrences of the attribute can help us gauge if they're any useful patterns noticed.

We set the minimum metric lift to 1, as a lift greater than one indicates that the confidence of the rule exceeds the benchmark confidence of the attribute being predicted, indicating that the rule is a potential good predictor of the attribute.

Trial 1: Price Discretized to 3 bins of equal size with non-equal frequency

We first performed the Apriori algorithm on the data, where the price is discretized into 3 bins of equal size. We obtained 10754 rules in total, out of which 3786 rules had price in the RHS of the rule [250 for the Mid-Price Range, and 3536 for the Lowest Price Range]. There were no rules which predicted the highest price range, which may be possibly due to the distribution of price skewed towards the left so there are less rows for the higher price range.

Price_Attribute	Condition_Attribute	Lift	Confidence	Number of Matched Rules
Mid Price Range	odometer='(-inf-91000.5]'	1.55	0.43	27
Mid Price Range	drive=4wd	1.27	0.35	27
Mid Price Range	condition=excellent	1.18	0.32	27
Mid Price Range	size=full-size	1.14	0.31	27
Mid Price Range	transmission=automatic	1.01	0.28	81

Price_Attribute	Condition_Attribute	Lift	Confidence	Number of Matched Rules
Lowest Price Range	odometer='(141002.5-inf)'	1.30	0.84	265
Lowest Price Range	drive=fwd	1.21	0.78	633
Lowest Price Range	type=sedan	1.20	0.77	303
Lowest Price Range	condition=good	1.19	0.77	197
Lowest Price Range	size=compact	1.19	0.76	5
Lowest Price Range	paint_color=silver	1.11	0.71	19
Lowest Price Range	cylinders=4_cylinders	1.10	0.71	383
Lowest Price Range	odometer='(91000.5-141002.5]'	1.08	0.69	153
Lowest Price Range	size=mid-size	1.08	0.70	163
Lowest Price Range	cylinders=6_cylinders	1.04	0.67	217
Lowest Price Range	fuel=gas	1.02	0.66	1169
Lowest Price Range	title_status=clean	1.01	0.65	1157

The above tables show the lift and confidence for the rule where the condition attribute is the only attribute in the LHS, and the price attribute is the only attribute in the RHS. It also shows the total number of rules where the condition attribute and price attribute occurred but in combination with other attributes from the dataset.

Observations:

- For both price ranges, odometer variable seems to have the greatest lift for the price ranges, indicating that it is potentially a good predictor for the price ranges. It seems that there is an inverse relation between price and the odometer ranges, as if the odometer distance increases, the price reduces. Logically thinking as well, more the odometer distance, more the car has been used and driven, hence its less new, so it would be sold for much lesser.
- Another rule which had a high lift was the drive attribute, which showed that Four-Wheel Drive Cars (4wd) are more likely to be in the greater price range, and the Front-Wheel Drive (fwd) seem to be in the lower price range.
- A car which is of good condition is more likely to be in a lower price range, but an excellent condition car will be sold for higher.
- In general, if the size of the car increases, it seems to be in the higher price range. For example, the rule for compact size cars and mid-size cars seem to be for the lower price range, whereas the rule for full size cars seem to be for the higher price range.
- An interesting rule which is noticed is that if the paint color of the car is Silver, it is more likely to be in the lower price range. This rule may have biases based on the weather condition of where the car is bought and what condition the cars are in as well, or it could simply be a coincidence, but it was an interesting observation.

Trial 2: Price Discretized to 3 bins of non-equal size with equal frequency

Due to the one-sided distribution of the price attribute in the dataset, we decided to do another trial for the association rules where we discretized the attribute into 3 bins of unequal size, but equal frequency so that each value so produced has an almost equal weight of occurrence in the dataset.

Price_Attribute	Condition_Attribute	Lift	Confidence	Number of Matched Rules
Max Price Range	odometer='(-inf-91000.5]'	1.73	0.58	49
Max Price Range	cylinders=8_cylinders	1.48	0.49	5
Max Price Range	drive=4wd	1.38	0.46	59
Max Price Range	size=full-size	1.22	0.41	75
Max Price Range	condition=excellent	1.18	0.39	69
Max Price Range	type=SUV	1.09	0.36	9
Max Price Range	transmission=automatic	1.01	0.34	135

Price_Attribute	Condition_Attribute	Lift	Confidence	Number of Matched Rules
Mid Price Range	odometer='(91000.5-141002.5]'	1.30	0.43	27
Mid Price Range	cylinders=4_cylinders	1.15	0.38	41
Mid Price Range	drive=fwd	1.12	0.37	43
Mid Price Range	type=sedan	1.11	0.37	19
Mid Price Range	condition=excellent	1.10	0.36	29
Mid Price Range	type=SUV	1.04	0.34	1
Mid Price Range	size=mid-size	1.04	0.35	19
Mid Price Range	transmission=automatic	1.01	0.33	153
Mid Price Range	fuel=gas	1.01	0.34	153

Price_Attribute	Condition_Attribute	Lift	Confidence	Number of Matched Rules
Lowest Price Range	odometer='(141002.5-inf)'	1.63	0.54	27
Lowest Price Range	condition=good	1.38	0.46	27
Lowest Price Range	drive=fwd	1.28	0.43	75
Lowest Price Range	type=sedan	1.27	0.42	57
Lowest Price Range	size=mid-size	1.11	0.37	27
Lowest Price Range	cylinders=6_cylinders	1.11	0.37	27
Lowest Price Range	cylinders=4_cylinders	1.05	0.35	45
Lowest Price Range	fuel=gas	1.03	0.35	203
Lowest Price Range	title_status=clean	1.02	0.34	195

Similar to the previous trial, the above tables show the lift and confidence for the rule where the condition attribute is the only attribute in the LHS, and the price attribute is the only attribute in the RHS. It also shows the total number of rules where the condition attribute and price attribute occurred but in combination with other attributes from the dataset.

Observations:

- There was a total of 7340 rules obtained, out of which 1315 had a price attribute in the RHS.
- There rules obtained for all 3 price range bins, as opposed to just 2 in the previous trial.
- Similar to the previous trial, the maximum lift for all them seems to be for odometer variables, and once again with an inverse relationship that if distance increases, the price reduces.

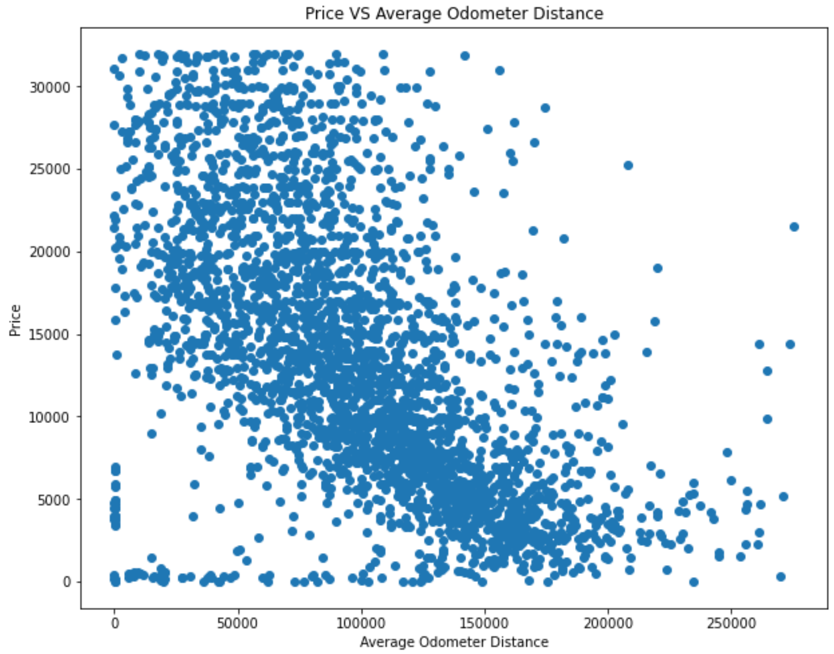
- The fwd seems to be for mid or low-price range, and 4wd is for the maximum price range.
- Something different learned from the previous trial is that the Sedan Type is for the lowest price range, whereas the SUV type is in the mid or higher price range.
- In general, the mid-price range due to it being of almost equal frequency as the other attributes and comparatively smaller bin size, it is a somewhat combination of some of the cars in the highest price range and the lowest price range, because of which there are some intersections.

Conclusions from Association Rules:

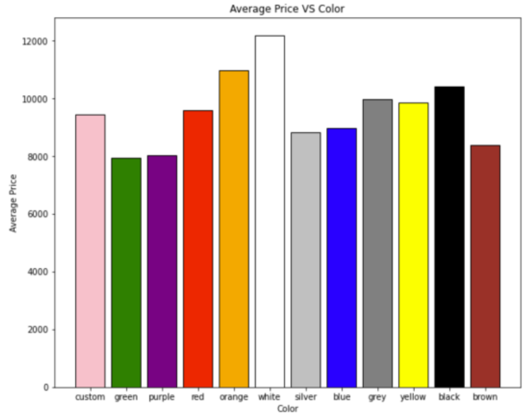
- As odometer distance seems to increase, the price tends to decrease and the lift is always high when the odometer seems to predict the price, indicating it is a strong predictor potentially.
- Four Wheel Drive Cars seem to be sold for much higher price ranges compared to Front Wheel Drive Cars.
- As size increases, price also seems to increase, as well as with condition, as condition improves the price also increases.
- Something common in both trials and all price ranges, is that the title status is often always clean, transmission is automatic and the form of fuel is usually gas. This is mainly due to the other possible values of the three attributes occurring much less frequently compared to these three variables, i.e. most listings are with title status clean, fueled by gas and of automatic transmission, so for further analysis with association rules, it would be useful to get data which has almost equal frequency and weight of the categorical variables as well.

<u>Data Visualization and Analysis</u>

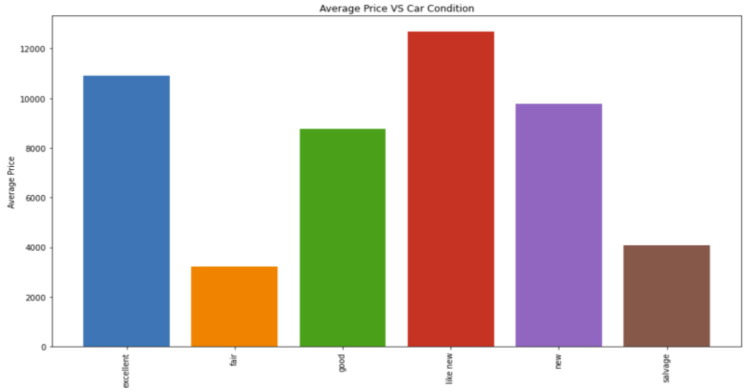
Based on some of the patterns noticed with the unsupervised learning method, we graphed some graphs to learn more about the relationships between price and the other attributes. We graphed several plots, but these were the ones which gave us the most valuable insights as to how what could be used to predict the price of a used car on Craigs List.



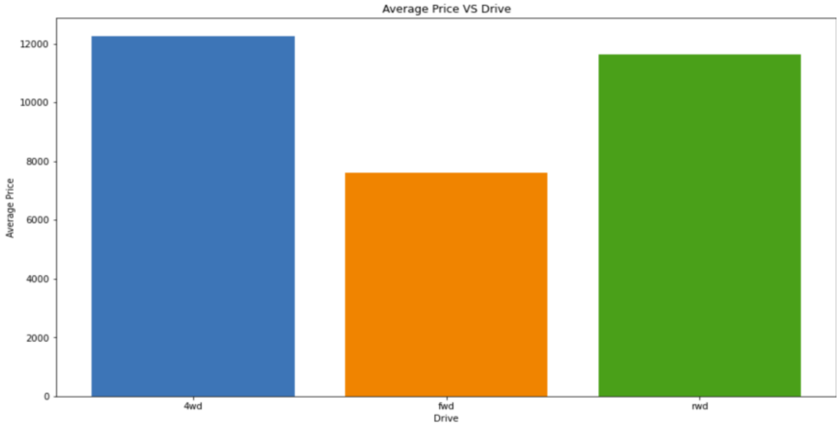
In this first scatterplot for Price VS Average Odometer Distance, we can notice a downward trend, that as the average odometer distance increases, the price of the car reduces as well, a pattern which was noticed in the unsupervised learning model as well.



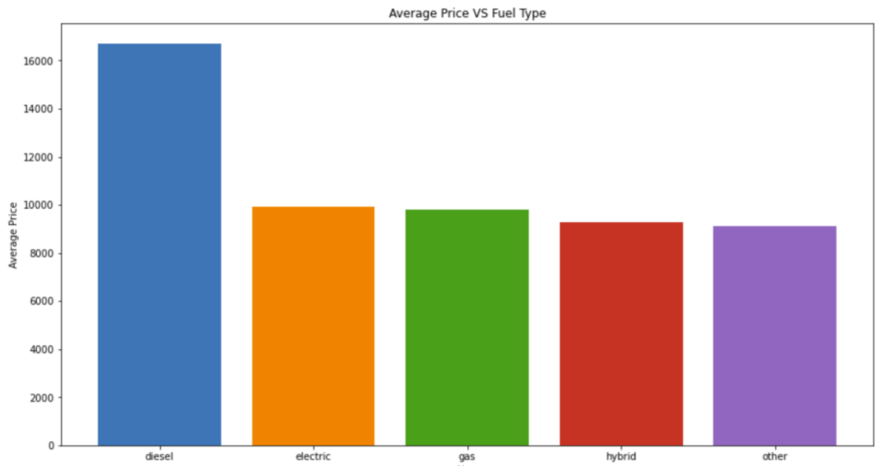
In this graph, we plotted the average price of the cars based on their paint colors. What is interesting that a car which is white seems to be priced higher compared to the rest of the cars, and that purple and green have a lower average price. This could also be due to a bias present in the data, such as some areas where the weather conditions are sunnier and less rain such as California, they're more likely to have a car which is white as it will less likely get dirty. The color silver being in the lower price ranges also is noticed in the association rules method results.



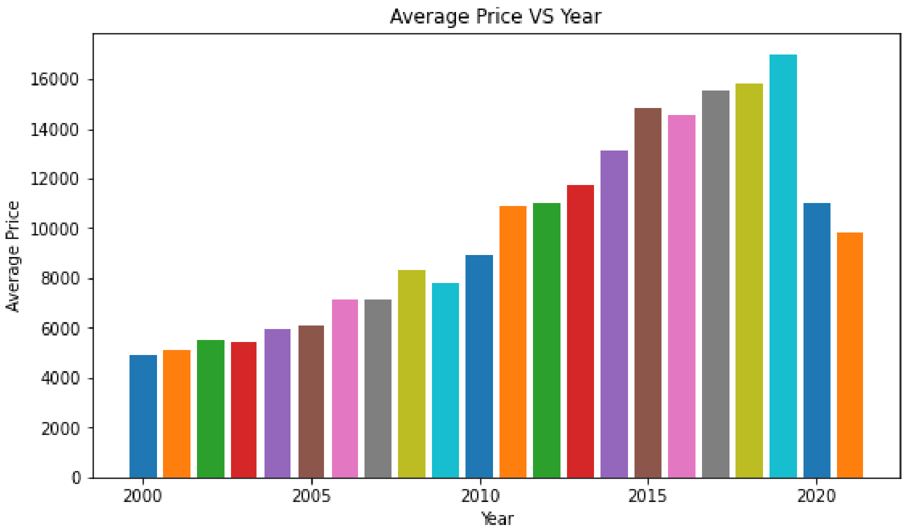
In this graph for the average price of a car based on condition shows that the better the condition of the car, the higher its price will be. What's interesting is that a salvaged car is priced higher than that of a fair condition car. In this case, for future analysis it would help if we had the information of the individuals selling and buying a car of salvage condition and understand the rationale behind it and why the price of those cars might be higher.



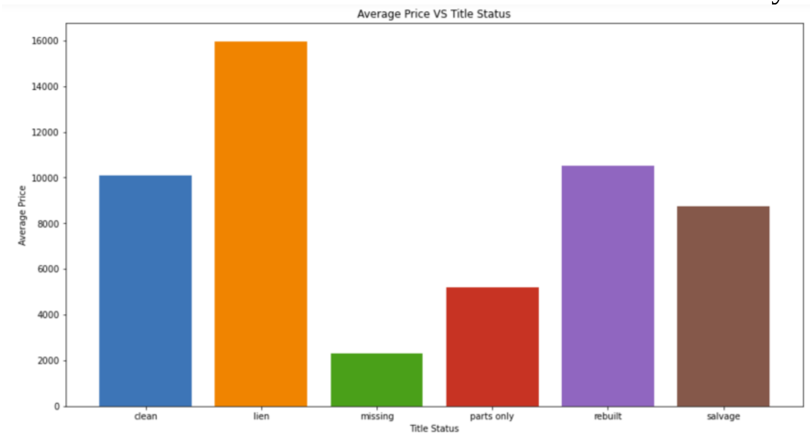
In the graph for the average price of different types of drive method for the car, it seems that a 4wd and Rwd seem to be priced higher than the fwd. We gauged this pattern difference in our association rule analysis, but what is interesting is that in average the rwd car seems to be priced higher than the fwd, giving us some insight into what price ranges it might fall under.



In this graph for the average price vs fuel type, it's interesting to see that diesel based cars seem to have a much higher average price compared to the rest, showing that a diesel based car is more likely to be priced higher.



This scatterplot which shows the average price for a car model of different years shows that older the car is, it is then more likely to be priced lower. What is interesting is that for the recent year models is that they have lower average prices, this may be because the manufacturers themselves are selling the car at the moment and a customer will be more likely to directly buy it from them as opposed to buying it from Craigs List unless it is priced significantly lower than the manufacturer.



This graph between the average price of the car based on title status is interesting because it shows that car is more likely to priced higher if it is of 'lien' status, basically showing that if the seller of the car is selling the car and is basically doing so in order to fulfill a debt, it will be priced higher than normal due to the fact that the seller is trying to fulfill a debt which they owe so they will sell it their car for the maximum amount possible.

Correlation Analysis

```
            Correlation Ranking Filter
Ranked attributes:
 0.2686  10 drive
 0.1827   6 fuel
 0.1729  11 size
 0.1499   4 condition
 0.1439  12 type
 0.1424   5 cylinders
 0.1109   2 year
 0.0783  13 paint_color
 0.0625   3 manufacturer
 0.0545   9 transmission
 0.0196   8 title_status
-0.4335   7 odometer
```

Based on these results obtained from Weka for the correlation coefficients of the attributes with price, it quantifies and to an extent validates our results from the unsupervised learning method and graphical analysis:

- The highest absolute correlation of price is with odometer, and the relationship is inverse and negative in nature, i.e. if odometer distance value increases, price decreases.
- The 2nd attribute which has the highest absolute value of correlation is drive, which was also noticed in the association rules method, that if the car is 4wd or rwd it is more likely to be priced higher, and it is more likely to be priced lower if it is fwd.
- The other attributes in general have a lower correlation coefficient value, and this can be due to the fact that some values are weighted more than the others for attributes such as title_status, manufacturer etc. Due to these uneven distributions of data, and presence of so many nominal attributes, it wouldn't make sense to go for a regression analysis, but rather for a modelling done by Naive Bayes Classification or Decision Trees. Also, the correlation amongst the variables isn't that suitable and high enough to get good results with regression.

Supervised Learning

Naive Bayes Classification

The second method we are going to use is Naive Bayes classifier. The Bayes theorem tells us how to compute the conditional probability - P(A|B) , which is the probability that event A occurs given the fact that event B has occurred. In this case, the Naive Bayes classifier is an effective method to predict the price of used cars based on different characteristics of the car.

We did two pre-processing methods. For our first method, we **kept all duplicate rows** and **discretized price into 5 equal bins**, the results for the Naïve Bayes classifier are the following:

```
Correctly Classified Instances        9549               56.9647 %
Incorrectly Classified Instances      7214               43.0353 %
Kappa statistic                          0.3889
Mean absolute error                      0.2125
Root mean squared error                  0.3278
Relative absolute error                 74.6252 %
Root relative squared error             86.9036 %
Total Number of Instances            16763

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.353    0.108    0.398      0.353    0.374      0.257    0.802     0.387     12276_18413
                 0.597    0.222    0.560      0.597    0.578      0.369    0.766     0.582     6138_12275
                 0.742    0.183    0.719      0.742    0.730      0.556    0.859     0.829     0_6137
                 0.420    0.047    0.301      0.420    0.351      0.319    0.897     0.318     24513_max
                 0.160    0.035    0.278      0.160    0.203      0.161    0.858     0.264     18414_24512
Weighted Avg.    0.570    0.165    0.560      0.570    0.563      0.404    0.821     0.608

=== Confusion Matrix ===

    a    b    c    d    e   <-- classified as
  992 1098  297  264  162 |    a = 12276_18413
  485 3217 1470  118   97 |    b = 6138_12275
  228 1168 4804  145  132 |    c = 0_6137
  228   44   26  327  153 |    d = 24513_max
  560  220   88  231  209 |    e = 18414_24512
```

The correctly classified instances is 9549, and the incorrectly classified instances is 7214, which gives us a <u>56.96% overall accuracy</u>. Compared to the result of association rules, 56.96% accuracy rate is not good enough to predict the price of used cars.

For our second pre-processing method, we **removed duplicate rows** since Naïve Bayes works better with a smaller dataset. We also **discretized price into 3 equal bins.** We chose the 66 percent split test to prevent the overfitting issue.

Below are the results for the Naïve Bayes classifier,

```
=== Summary ===

Correctly Classified Instances       10873               76.1308 %
Incorrectly Classified Instances      3409               23.8692 %
Kappa statistic                          0.5148
Mean absolute error                      0.2131
Root mean squared error                  0.3273
Relative absolute error                 63.6573 %
Root relative squared error             80.0103 %
Total Number of Instances            14282

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.880    0.267    0.857      0.880    0.868      0.620    0.900     0.946     '(-inf-10666.333333]'
                 0.597    0.155    0.593      0.597    0.595      0.441    0.841     0.594     '(10666.333333-21332.666667]'
                 0.372    0.034    0.488      0.372    0.422      0.383    0.910     0.444     '(21332.666667-inf)'
Weighted Avg.    0.761    0.218    0.755      0.761    0.757      0.552    0.884     0.809

=== Confusion Matrix ===

    a    b    c   <-- classified as
 8109  945  161 |    a = '(-inf-10666.333333]'
 1288 2334  290 |    b = '(10666.333333-21332.666667]'
   66  659  430 |    c = '(21332.666667-inf)'
```

After we implemented these pre-processing techniques, the correctly classified instances is 10873, and the incorrectly classified instances is 3409, which increases the overall accuracy to 76.13%. The stratified accuracies for “a”, “b” and “c” are 88%, 60% and 37%. According to the result of stratified accuracies, Naïve Bayes is good at predicting the lower price and middle price vehicles.

Contingency Tables and Probability (Full tables attached in appendix):

Contingency table and probability give us overall categories attributes and their probability used to predict the price. Below are the observations we have found for each variable from our contingency table.

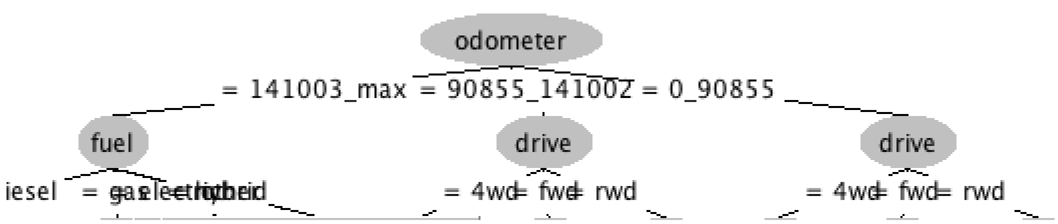
- For odometer, it shows that with a higher mileage, the car is sold with a lower price and vice versa.
- For years, it shows that there is a small trend of the older the car year, the lower the price of the car and vice versa.
- For manufacturers, it shows that most of the cars listed for sale are American and Japanese cars. Yet, most used cars are sold within the lowest price bracket, including European cars. Hence, manufacturer does not seem to be a significant factor to price.
- For condition, 84% of the data falls under the excellent and good condition. Yet, it does not show any trend of the better condition of the car having a higher sold price.
- For cylinders, it shows that the number of cylinders does not really affect the price. Almost for all, except for 12-cylinder cars are sold in the lowest bracket of price. There is no trend shown for the number of cylinders vs. the price of the used car.
- For fuel, it shows that gas, hybrid cars are mostly sold in the lowest bracket of price, while for diesel cars are mostly sold in the middle bracket of the price. It seems like the type of fuel has an influence on the price of the used car.
- For the Title Status, it does not show a specific trend with price, because a clean title is important for every price range.
- For the Transmission, it does not really affect the price. Automatic, as the most common transmission type, it falls in every price range with over 90% probability.

- For the Drive, 4-wheel-drive cars are mostly sold in the middle and highest price range, front-wheel-drive cars are mostly sold in lowest and middle price range, while rear-wheel-drive cars have an equal probability for each price range. In this case, drive type is an important factor to predict price.
- For the Size, it has a clear trend that the bigger the car, the higher probability to be sold in the middle and highest price range.
- For the Type, it shows that most sedan types are sold at lowest and middle range, while truck types are more likely to be sold at middle and higher range. While the SUV type is more likely to be sold in every price range with around 30% probability. So, the type of cars basically follows the same trend with size, the bigger the car, the higher the price.
- For the Paint Color, it shows an interesting trend, white cars are more likely to be sold in the middle and highest price range, while other paint colors mostly fall into the lowest and middle price range.

From our results, we can see that there are a few variables which carry more weight due to greater frequency in the dataset that may contribute to the price of the used cars. Due to the biases in the data, it is hard to make any conclusion on which variables have the most influence on price and there is a possible 25% inaccuracy rate which needs to be recovered in order to get better results. Therefore, we will run the decision tree as our next model to give us a more informed result.

<u>Decision Trees</u>

The third method we used is the Decision Tree. Building a decision tree will help us identify the feature importance from the most informative to the least by looking at the splitting nodes according to certain cutoff values in the feature. Since this whole dataset contains more than 10 attributes, visualizing the entire tree was a bit difficult, but below we show a partial decision tree generated by Weka.



From above, we can see that the root node of the tree is the “odometer”, which indicates the “odometer”, as an attribute, best splits the data. It gives the largest information gain compared to other attributes, which means “odometer” is the most informative attribute to predict the price of used cars. Besides, the “condition” and “paint color” are two least informative attributes shown in the bottom nodes of the decision tree, which give the lowest information gains. Therefore, we can conclude that when making predictions on the price of used cars, in all of these nodes all the other features of the data, “odometer” gave us the best results, while the “condition” and “paint color” are not as important as the other features like fuel, drive and year.

In addition, we got the summary & confusion matrix as followed:

```
=== Summary ===

Correctly Classified Instances       14702               80.9894 %
Incorrectly Classified Instances      3451               19.0106 %
Kappa statistic                          0.641
Mean absolute error                      0.1711
Root mean squared error                  0.314
Relative absolute error                 48.2972 %
Root relative squared error             74.4099 %
Total Number of Instances            18153

=== Confusion Matrix ===

    a    b    c   <-- classified as
 9796 1069  177 |    a = '(-inf-10666.333333]'
 1234 3816  283 |    b = '(10666.333333-21332.666667]'
  132  556 1090 |    c = '(21332.666667-inf)'
```

From the above summary, we get overall accuracy **80.99%,** with 14,702 out of 18,153 instances correctly classified. Next, we take the pre-processing method of **removing all duplicate rows** and **discretized price into 3 unequal bins**. Below is the new summary & confusion matrix:

```
=== Summary ===

Correctly Classified Instances       11438               80.0868 %
Incorrectly Classified Instances      2844               19.9132 %
Kappa statistic                          0.6023
Mean absolute error                      0.1833
Root mean squared error                  0.3161
Relative absolute error                 54.7404 %
Root relative squared error             77.2562 %
Total Number of Instances            14282

=== Confusion Matrix ===

    a    b    c   <-- classified as
 8132  950  133 |    a = '(-inf-10666.333333]'
  946 2737  229 |    b = '(10666.333333-21332.666667]'
   77  509  569 |    c = '(21332.666667-inf)'
```

We got the accuracy here is **80.09%,** with 11,438 out of 14,282 instances correctly classified.
With this preprocessing method, we got a bit lower accuracy than directly using the original dataset, but still a good number.

Next, we try another pre-processing method of **removing duplicates** but **with equal bins on price**.
Below is the new summary & confusion matrix:

```
=== Summary ===

Correctly Classified Instances       10247               71.7477 %
Incorrectly Classified Instances      4035               28.2523 %
Kappa statistic                          0.5762
Mean absolute error                      0.2583
Root mean squared error                  0.3755
Relative absolute error                 58.1248 %
Root relative squared error             79.6518 %
Total Number of Instances            14282

            === Confusion Matrix ===

               a    b    c   <-- classified as
             3294  967  483 |    a = '(-inf-5502.5]'
              982 2964  834 |    b = '(5502.5-10999.5]'
              117  652 3989 |    c = '(10999.5-inf)'
```

Here we got the overall accuracy of **71.75%** with 10,247 out of 14,282 instances correctly classified. The accuracy dropped a lot compared to what we got from the previous two analyses. When looking at the stratified accuracies of the classifier, we got:

'a'= 69.44%
'b'=62.01%
'c'=83.84%

With comparing all three decision trees we got from above, it seems keeping duplicates and discretizing the price variable into bins of equal size, but non equal frequency makes the model more accurate than directly building trees from the unprocessed data.

<u>Random Forests</u>

The last method we used is random forests. As we learned in lecture, in order to account for overfitting in decision trees, we decided to run a random forests classification on Weka. In addition, random forests consist of multiple single trees each based on a random sample of the training data. After checking a variation of preprocessing steps and discretization of the price and odometer attributes, the best results were obtained with the following steps:

- Price discretized into 3 bins of equal size
- Odometer discretized into 3 bins of equal frequency
- Duplicates Rows NOT removed

To get the appropriate number of attributes for random forest we got the following results:
Number of Attributes = m = 13
Option 1: m^(½) = 3.6
Option 2: log2(m) = 3.7

Based on these calculations, we checked the random forests for: 2 features, 3 features and 4 features and these were the results we obtained:

4 Features:

```
=== Summary ===

Correctly Classified Instances       15400               84.8345 %
Incorrectly Classified Instances      2753               15.1655 %
Kappa statistic                          0.7132
Mean absolute error                      0.1489
Root mean squared error                  0.269
Relative absolute error                 42.0231 %
Root relative squared error             63.7586 %
Total Number of Instances            18153

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.914    0.161    0.898      0.914    0.906      0.756    0.952     0.968     '(-inf-10666.333333]'
                 0.763    0.098    0.764      0.763    0.764      0.665    0.926     0.839     '(10666.333333-21332.666667]'
                 0.698    0.021    0.781      0.698    0.737      0.711    0.968     0.828     '(21332.666667-inf)'
Weighted Avg.    0.848    0.129    0.847      0.848    0.847      0.725    0.946     0.916
```

With 4 features chosen, we got an overall accuracy of 84.83% with 15,400 out of 18,153 instances correctly classified.

3 Features:

```
=== Summary ===

Correctly Classified Instances       15421               84.9501 %
Incorrectly Classified Instances      2732               15.0499 %
Kappa statistic                          0.7146
Mean absolute error                      0.1525
Root mean squared error                  0.2687
Relative absolute error                 43.0464 %
Root relative squared error             63.6709 %
Total Number of Instances            18153

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.918    0.165    0.896      0.918    0.907      0.758    0.952     0.968     '(-inf-10666.333333]'
                 0.759    0.096    0.767      0.759    0.763      0.666    0.927     0.840     '(10666.333333-21332.666667]'
                 0.696    0.020    0.788      0.696    0.739      0.715    0.969     0.830     '(21332.666667-inf)'
Weighted Avg.    0.850    0.130    0.848      0.850    0.848      0.727    0.946     0.917
```

With 3 features chosen, we got an overall accuracy of 84.95% with 15,421 out of 18,153 instances correctly classified.

2 Features:

```
=== Summary ===

Correctly Classified Instances       15406               84.8675 %
Incorrectly Classified Instances      2747               15.1325 %
Kappa statistic                          0.7113
Mean absolute error                      0.1578
Root mean squared error                  0.2703
Relative absolute error                 44.5428 %
Root relative squared error             64.0535 %
Total Number of Instances            18153

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.923    0.177    0.890      0.923    0.906      0.755    0.951     0.966     '(-inf-10666.333333]'
                 0.749    0.092    0.771      0.749    0.760      0.662    0.925     0.840     '(10666.333333-21332.666667]'
                 0.684    0.019    0.799      0.684    0.737      0.714    0.969     0.833     '(21332.666667-inf)'
Weighted Avg.    0.849    0.137    0.846      0.849    0.847      0.724    0.945     0.916
```

With 2 features chosen, we got an overall accuracy of 84.87% with 15,406 out of 18,153 instances correctly classified.

In general, the accuracies of the model were more than that of Decision trees, with 3 attributes being considered being the most accurate, so we explored that further. This was our observations:

```
=== Confusion Matrix ===

     a     b     c   <-- classified as
 10134   804   104 |     a = '(-inf-10666.333333]'
  1055  4049   229 |     b = '(10666.333333-21332.666667]'
   116   424  1238 |     c = '(21332.666667-inf)'
```

When looking at the stratified accuracies of the classifier, we got:
‘a’ = 91.78%
‘b’ =75.92%
‘c’ =69.63%

```
Attribute importance based on average impurity decrease (and number of nodes using that attribute)

      0.76 ( 46042)  year
      0.63 ( 65740)  manufacturer
      0.58 ( 90404)  paint_color
      0.52 ( 46887)  type
      0.5  ( 52738)  odometer
      0.48 ( 76137)  condition
      0.47 ( 51734)  cylinders
      0.45 ( 62963)  size
      0.43 ( 48174)  drive
      0.42 ( 23730)  fuel
      0.37 ( 32408)  title_status
      0.36 ( 23289)  transmission
```

These accuracies were much better compared to what we got in decision trees. This random forest model gives a really high accuracy on correctly classified price falling in the minimum range under $10,666 as it prevents the issue of overfitting and also considers more attributes. In general, random forests enforce diversity because of which our highly correlated attributes: odometer, drive and fuels weren’t necessarily considered but rather the year, manufacturer and paint color attributes were used to give us the results as seen in the attribute importance ranking above. Random forests try to improve on bagging by “de-correlating” the trees and deals with the overfitting which might have occurred with decision trees, because of which we are getting higher accuracy in this model compared to the rest.

<u>Conclusions and Takeaways</u>

The main goal of our project is to use machine learning methods to predict what are the major attributes that may influence the price of a used car and help users understand how better to price their cars or expect from their purchase of their car on Craigslist. From our test results, we found out that odometer is the strongest attribute to provide prediction to the price of used cars.

From association rules:

- Odometer and Drive attributes are potentially good predictors for price as they have high lift for their rules
- There seems to be a potential inverse relationship between price and the odometer distance.
- There is a potential positive relationship between price and size of car and price and condition

- The best predictor for price range based on lift seems to be the odometer variable, and the better trial is when the price is discretized into bins of equal frequency

From graphical and statistical analysis,

- We quantified and visualized the patterns with noticed with association rules, and gained some more insight about some other factors which might affect the price of a listing

For naive bayes:

- After taking pre-processing steps, the accuracy has improved from 56.96% to 76.13%.
- According to the stratified accuracies, our model seems to be doing a good job on predicting cars that are sold in the lower price bracket and middle price bracket.
- From the contingency tables and probabilities, we can see that there are a few attributes such as odometer, drive, fuel, paint color etc that are correlated with the price of the used cars. Yet, just by looking at the tables, we cannot really tell the level of information gain of the attributes with price . Therefore, we may need the help from decision trees to make a more concise conclusion.

For the decision tree：

- Gives the accuracy of 80.99%.
- Odometer, as a feature, gives the highest information gain, and best splits the tree among all the other attributes, which matches what we've found from association rule.
- May have issue of overfitting

From random forests,

- Gives the highest accuracy of 84.9501%
- Overcomes issue of overfitting
- Considers attributes which don't necessarily have high correlation and seems to be the best model for us

At the moment, since our random forest model seems to produce the highest accuracy, as well as give us a strong overview of the full dataset, that is the model we propose to use for business-based purposes. Our analysis has given a better overview as to how the price of a used car is influenced by attributes such as odometer distance, type of drive, transmission. With our model, a potential buyer will be able to have a clear idea of how a car would be based on the price range they have in their budget, i.e. that if a buyer has a budget set for prices between 1 to 10000, our model will show them that the car is more likely to have a high odometer distance, it is more likely to be a sedan type, front-wheel drive and of good to fair condition. If that doesn't meet their needs, then they would have to change their budget accordingly. Buyers could also use our model to decide the budget for their purchase based on the preferences they have, as our model would give them a price range for what type of used car they hope to purchase.

In case of sellers, our model will give them a better idea of how to price their cars appropriately when selling on Craigs List based on the attributes their car possesses. For example, a car which is driven only 1000 miles, four-wheel drive, SUV Type is more likely to be in the highest price range of 21330 or above.

Limitations and Improvements

Even though our test results have provided useful information, there are few limitations for us to improve upon our model with:

- Lack of continuous variables and use of mainly nominal variables: we should include more continuous variables such as the original price of the car, the average price of a similar model, so that we can do a regression analysis (logistic, polynomial or linear) in the future, to better quantify the relationship between the attributes present and give a direct number for the price. At the moment, our model is using nominal attributes and resulting in a price range, but with regression we would be able to give exact numerical values which would be more useful for users.
- Improve Accuracy: our accuracy at the moment is a maximum of 80%, with more attributes and with more quantified data, we hope to increase the accuracy for both the decision tree and naive bayes model.
- Lack of details on the condition of the car: it would be better if we could include more specific elements of the condition of the vehicle, such as the battery life, tire condition and interior quality. We could then include more information about how these condition-based attributes contribute to the price as opposed to simply just having it stated as ‘good’, ‘excellent’, ‘fair’. These attribute values are rather ambiguous and do not necessarily represent the exact condition of the car.
- The data only consists of information about the US market - we also plan to look into used cars data from other countries to make comparisons, as Craigslist is present all around the world and not just in the US. In general, the regional data we did have did not give us much insight into how the data is influenced by the region.
- Our data in general seems to have some categorical attribute values which weigh significantly higher compared to the rest of the attributes. This can cause some biases in the data towards certain attributes, for example, almost all cars had the title status as clean, and there were only few listings with a different value, due to which getting an understanding as to how those values could influence the price. We would like to do a future analysis with data which has attribute values well distributed for the categorical variables as well.
- Existence of biases in the data, there are some cars which are essentially sold for free even though they may be brand new, and we don’t have the user information to properly understand as to properly account for those situations. In the future, using user information we will be able to better grasp the user psyche and understand what hand they might have in the price of a used car. User information will tell us about why the car is being sold by the seller or why the user is buying a certain car, and we would be able to provide more informed suggestions and listing recommendations to them on the platform.

Appendix

Contingency and Probability Tables

Title_Status

Count of price	Column Labels				
Row Labels	'\'(-inf-10666.333333]\"	'\'(10666.333333-21332.666667]\"	'\'(21332.666667-inf)\"	(blank)	Grand Total
clean	25477	10594	3123		39194
lien	170	302	155		627
missing	23				23
'parts only'	11	2			13
rebuilt	954	470	91		1515
salvage	460	151	23		634
(blank)					
Grand Total	**27095**	**11519**	**3392**		**42006**

Probability

Title_Status/Price	'\'(-inf-10666.333333]\"	'\'(10666.333333-21332.666667]\"	'\'(21332.666667-inf)\"
clean	0.94	0.92	0.92
lien	0.01	0.03	0.05
missing	0.00	0.00	0.00
'parts only'	0.00	0.00	0.00
rebuilt	0.04	0.04	0.03
salvage	0.02	0.01	0.01

Transmission

Count of transn	Column Labels			
Row Labels	'\'(-inf-10666.333333]\"	'\'(10666.333333-21332.666667]\"	'\'(21332.666667-inf)\"	Grand Total
automatic	25062	10794	3201	39057
manual	1899	656	168	2723
other	134	69	23	226
Grand Total	**27095**	**11519**	**3392**	**42006**

Probability

Transmission/Price	'\'(-inf-10666.333333]\"	'\'(10666.333333-21332.666667]\"	'\'(21332.666667-inf)\"
automatic	0.92	0.94	0.94
manual	0.07	0.06	0.05
other	0.00	0.01	0.01

Drive

Count of drive	Column Labels			
Row Labels	'\'(-inf-10666.333333]\"	'\'(10666.333333-21332.666667]\"	'\'(21332.666667-inf)\"	Grand Total
4wd	8223	5553	2179	15955
fwd	14869	3789	431	19089
rwd	4003	2177	782	6962

Probability

Drive/Price	'\'(-inf-10666.333333]\"	'\'(10666.333333-21332.666667]\"	'\'(21332.666667-inf)\"
4wd	0.30	0.48	0.64
fwd	0.55	0.33	0.13
rwd	0.15	0.19	0.23

rwd: less common, less owners

Size

Count of size	Column Labels			
Row Labels	'\'(-inf-10666.333333]\"	'\'(10666.333333-21332.666667]\"	'\'(21332.666667-inf)\"	Grand Total
compact	4782	1302	168	6252
full-size	11924	6512	2424	20860
mid-size	9876	3548	779	14203
sub-compact	513	157	21	691
Grand Total	**27095**	**11519**	**3392**	**42006**

Probability

Size/Price	'\'(-inf-10666.333333]\"	'\'(10666.333333-21332.666667]\"	'\'(21332.666667-inf)\"
compact	0.18	0.11	0.05
full-size	0.44	0.57	0.71
mid-size	0.36	0.31	0.23
sub-compact	0.02	0.01	0.01

Type

Count of type	Column Labels				
Row Labels	'\'(-inf-10666.333333]\"	'\'(10666.333333-21332.666667]\"	'\'(21332.666667-inf)\"	(blank)	Grand Total
bus	26	13	9		48
convertible	509	258	87		854
coupe	1300	453	171		1924
hatchback	1670	394	43		2107
mini-van	1060	270	67		1397
offroad	41	37	27		105
other	121	54	20		195
pickup	1114	1007	464		2585
sedan	10531	2768	303		13602
SUV	7530	3824	962		12316
truck	1740	1734	983		4457
van	725	468	208		1401
wagon	728	239	48		1015
(blank)					
Grand Total	**27095**	**11519**	**3392**		**42006**

Probability

Type/Price	'\'(-inf-10666.333333]\"	'\'(10666.333333-21332.666667]\"	'\'(21332.666667-inf)\"
bus	0.00	0.00	0.00
convertible	0.02	0.02	0.03
coupe	0.05	0.04	0.05
hatchback	0.06	0.03	0.01
mini-van	0.00	0.02	0.02
offroad	0.00	0.00	0.01
other	0.00	0.00	0.01
pickup	0.04	0.09	0.14
sedan	0.39	0.24	0.09
SUV	0.28	0.33	0.28
truck	0.06	0.15	0.29
van	0.03	0.04	0.06
wagon	0.03	0.02	0.01

Paint_Color

Count of paint_	Column Labels				
Row Labels	'\'(-inf-10666.333333]\"	'\'(10666.333333-21332.666667]\"	'\'(21332.666667-inf)\"	(blank)	Grand Total
black	4673	2262	673		7608
blue	3284	1054	258		4596
brown	977	248	64		1289
custom	685	245	75		1005
green	929	211	57		1197
grey	3451	1545	410		5406
orange	128	52	30		210
purple	107	27	7		141
red	2820	1094	271		4185
silver	5004	1634	367		7005
white	4887	3074	1160		9121
yellow	150	73	20		243
(blank)					
Grand Total	**27095**	**11519**	**3392**		**42006**

Probability

Paint_Color/Price	'\'(-inf-10666.333333]\"	'\'(10666.333333-21332.666667]\"	'\'(21332.666667-inf)\"
black	0.17	0.20	0.20
blue	0.12	0.09	0.08
brown	0.04	0.02	0.02
custom	0.03	0.02	0.02
green	0.03	0.02	0.02
grey	0.13	0.13	0.12
orange	0.00	0.00	0.01
purple	0.00	0.00	0.00
red	0.10	0.09	0.08
silver	0.18	0.14	0.11
white	0.18	0.27	0.34
yellow	0.01	0.01	0.01

Count of manufacturer		Column Labels			
Row Labels	'\'(-inf-10666.333333]'	'\'(10666.333333-21332.66666	'\'(21332.666667-in	(blank)	Grand Total
acura	313	120	22		455
alfa-romeo	2		3		5
audi	267	148	68		483
bmw	640	384	94		1118
buick	452	169	20		641
cadillac	375	249	54		678
chevrolet	3502	1590	597		5689
chrysler	822	126	29		977
dodge	1155	348	110		1613
fiat	73	16	8		97
ford	3942	2204	820		6966
gmc	710	522	214		1446
harley-davidsor	2	4			6
honda	2310	687	106		3103
hyundai	966	328	21		1315
infiniti	204	99	28		331
jaguar	50	36	4		90
jeep	967	568	200		1735
kia	734	244	17		995
'land rover'	4	1			5
lexus	368	195	70		633
lincoln	259	92	34		385
mazda	535	130	12		677
mercedes-benz	416	265	89		770
mercury	244				244
mini	200	82	4		286
mitsubishi	213	44	9		266
nissan	1855	686	105		2646
pontiac	369	23	3		395
ram	376	428	273		1077
rover	74	47	15		136
saturn	246	7			253
subaru	804	357	85		1246
tesla	1		1		2
toyota	2534	986	235		3755
volkswagen	835	286	34		1155
volvo	276	48	8		332
(blank)					
Grand Total	27095	11519	3392		42006

Count of manufacturer		Column Labels	
Row Labels	'\'(-inf-10666.333333]\''	'\'(10666.333333-21332.666667]\''	'\'(21332.666667-inf)\''
acura	0.011551947	0.010417571	0.006485849
alfa-romeo	7.38E-05	0	0.000884434
audi	0.009854217	0.012848338	0.02004717
bmw	0.023620594	0.033336227	0.027712264
buick	0.016682045	0.014671412	0.005896226
cadillac	0.013840192	0.02161646	0.015919811
chevrolet	0.129248939	0.138032815	0.176002358
chrysler	0.030337701	0.01093845	0.008549528
dodge	0.042627791	0.030210956	0.032429245
fiat	0.002694224	0.001389009	0.002358491
ford	0.145488097	0.191336053	0.241745283
gmc	0.026204097	0.045316434	0.063089623
harley-davidson	7.38E-05	0.000347252	0
honda	0.085255582	0.059640594	0.03125
hyundai	0.035652334	0.028474694	0.006191038
infiniti	0.007529064	0.008594496	0.008254717
jaguar	0.001845359	0.003125271	0.001179245
jeep	0.035689242	0.049309836	0.058962264
kia	0.027089869	0.021182394	0.005011792
'land rover'	0.000147629	8.68E-05	0
lexus	0.013581842	0.016928553	0.020636792
lincoln	0.009558959	0.007986804	0.010023585
mazda	0.01974534	0.011285702	0.003537736
mercedes-benz	0.015353386	0.023005469	0.026238208
mercury	0.009005352	0	0
mini	0.007381436	0.007118673	0.001179245
mitsubishi	0.007861229	0.003819776	0.002653302
nissan	0.068462816	0.059553781	0.030955189
pontiac	0.013618749	0.001996701	0.000884434
ram	0.013877099	0.037156003	0.080483491
rover	0.002731131	0.004080215	0.00442217
saturn	0.009079166	0.000607692	0
subaru	0.029673371	0.030992274	0.025058962
tesla	3.69E-05	0	0.000294811
toyota	0.09352279	0.085597708	0.06928066
volkswagen	0.030817494	0.024828544	0.010023585
volvo	0.010186381	0.004167028	0.002358491

Count of condition		Column Labels			
Row Labels	'\'(-inf-10666.333333]'	'\'(10666.333333-21332.66666	'\'(21332.666667-in	(blank)	Grand Total
excellent	12208	6810	1956		20974
fair	1541	32			1573
good	10909	2648	622		14179
'like new'	2233	1983	774		4990
new	95	35	38		168
salvage	109	11	2		122
(blank)					
Grand Total	27095	11519	3392		42006

Count of condition		Column Labels	
Row Labels	'\'(-inf-10666.333333]\''	'\'(10666.333333-21332.666667]\''	'\'(21332.666667-inf)\
excellent	0.450562834	0.591197153	0.576650
fair	0.056873962	0.002778019	
good	0.40262041	0.229881066	0.183372
'like new'	0.082413729	0.17215036	0.228183
new	0.003506182	0.003038458	0.01120
salvage	0.004022882	0.000954944	0.000589

Count of cylinders		Column Labels			
Row Labels	'\'(-inf-10666.333333]'	'\'(10666.333333-21332.66666	'\'(21332.666667-in	(blank)	Grand Total
'10 cylinders'	106	68	20		194
'12 cylinders'	3	8	2		13
'3 cylinders'	61	22	5		88
'4 cylinders'	11911	4347	581		16839
'5 cylinders'	397	47	9		453
'6 cylinders'	10126	3687	1333		15146
'8 cylinders'	4491	3340	1442		9273
(blank)					
Grand Total	27095	11519	3392		42006

Count of cylinders		Column Labels	
Row Labels	'\'(-inf-10666.333333]\''	'\'(10666.333333-21332.666667]\''	'\'(21332.666667-inf)\
'10 cylinders'	0.003912161	0.00590329	0.005896
'12 cylinders'	0.000110722	0.000694505	0.000589
'3 cylinders'	0.002251338	0.001909888	0.001474
'4 cylinders'	0.439601402	0.377376508	0.171285
'5 cylinders'	0.01465215	0.004080215	0.002653
'6 cylinders'	0.373722089	0.320079868	0.392983
'8 cylinders'	0.165750138	0.289955725	0.425117

Count of fuel		Column Labels			
Row Labels	'\'(-inf-10666.333333]'	'\'(10666.333333-21332.66666	'\'(21332.666667-in	(blank)	Grand Total
diesel	450	746	410		1606
electric	9	4	1		14
gas	26122	10559	2956		39637
hybrid	477	189	22		688
other	37	21	3		61
(blank)					
Grand Total	27095	11519	3392		42006

Count of fuel		Column Labels	
Row Labels	'\'(-inf-10666.333333]\''	'\'(10666.333333-21332.666667]\''	'\'(21332.666667-inf)\
diesel	0.01660823	0.064762566	0.120872
electric	0.000332165	0.000347252	0.000294
gas	0.964089315	0.916659432	0.871462
hybrid	0.017604724	0.016407674	0.006485
other	0.001365566	0.001823075	0.000884

Count of odometer		Column Labels			
Row Labels	'\'(-inf-10666.333333]'	'\'(10666.333333-21332.66666	'\'(21332.666667-in	(blank)	Grand Total
'\'(141002.5-inf)'	11758	1932	289		13979
'\'(91000.5-14100	9718	3620	676		14014
'\'(-inf-91000.5]\	5619	5967	2427		14013
(blank)					
Grand Total	27095	11519	3392		42006

Count of odometer		Column Labels	
Row Labels	'\'(-inf-10666.333333]\''	'\'(10666.333333-21332.666667]\''	'\'(21332.666667-inf)\''
'\'(141002.5-inf)\''	0.433954604	0.167722893	0.085200472
'\'(91000.5-141002.5]\''	0.35866396	0.314263391	0.199292453
'\'(-inf-91000.5]\''	0.207381436	0.518013716	0.715507075

Count of price		Column Labels			
Row Labels	'\'(-inf-10666.333333]'	'\'(10666.333333-21332.66666	'\'(21332.666667-in	(blank)	Grand Total
2000	707	68	6		781
2001	820	79	13		912
2002	1109	107	13		1229
2003	1398	130	23		1551
2004	1724	177	30		1931
2005	1896	210	23		2129
2006	2148	316	48		2512
2007	2395	378	48		2821
2008	2491	489	80		3060
2009	1760	323	43		2126
2010	1927	566	63		2556
2011	1888	996	153		3037
2012	1865	978	186		3029
2013	1556	1266	235		3057
2014	1120	1247	355		2722
2015	774	1219	462		2455
2016	557	1112	413		2082
2017	432	1040	500		1972
2018	195	498	315		1008
2019	221	262	315		798
2020	110	58	67		235
2021	2		1		3
(blank)					
Grand Total	27095	11519	3392		42006

Count of price		Column Labels	
Row Labels	'\'(-inf-10666.333333]\''	'\'(10666.333333-21332.666667]\''	'\'(21332.666667-inf)\''
2000	0.026093375	0.00590329	0.001768868
2001	0.030263886	0.006858234	0.003832547
2002	0.040930061	0.009289001	0.003832547
2003	0.051596235	0.011285702	0.00678066
2004	0.063627976	0.015365917	0.00884434
2005	0.06997601	0.018230749	0.00678066
2006	0.079276619	0.027432937	0.014150943
2007	0.088392692	0.032815349	0.014150943
2008	0.091935782	0.042451602	0.023584906
2009	0.064956634	0.028040629	0.012676887
2010	0.071120133	0.04913621	0.018573113
2011	0.069680753	0.086465839	0.045106132
2012	0.068831888	0.084903203	0.054834906
2013	0.05742757	0.109905374	0.06928066
2014	0.04133604	0.108255925	0.104658019
2015	0.028566156	0.105825158	0.13620283
2016	0.020557298	0.096536158	0.121757075
2017	0.015943901	0.090285615	0.14740566
2018	0.0071969	0.04323292	0.092865566
2019	0.008156486	0.02274503	0.092865566
2020	0.00405979	0.005035159	0.019752358
2021	7.38E-05		0.000294811