Elizabeth Binkina (binkina2)

Project Description:

The primary goal of this project is to identify what features best explain the distribution of used car prices. Each used cars have a unique set of characteristics that influence its final price, such as age, mileage, horsepower, and more. This project aims to uncover patterns and understand how various features affect pricing trends in the used car market.

The first step in this project was to preprocess and clean the data to ensure it could be effectively analyzed for valuable insights. Many of the raw data entries were categorical representations of numerical values, that required conversion. After cleaning the data, I calculated summary statistics and created visualizations to identify key relationships between variables. When selecting clustering algorithms, I considered the data's structure and types to choose the most suitable method. Since the dataset contained both categorical and numeric features, one algorithm I choose was K-medoids, which is well-suited for mixed data types. For regression modeling, I encoded categorical variables into numeric values, allowing these features to be incorporated into regression models.

Using the k-medoids algorithm, I clustered the data into three distinct groups based on horsepower, price, and mileage. Cluster 3 represents luxury/performance, cluster 2 represents economy cars, and cluster 1 represents more affordable cars. Additionally, I applied complete linkage clustering, which segmented the data into five clusters based on mileage, price, and horsepower. Cluster 1 represented newer cars, cluster 2 represented relatively newer cars, cluster 3 represented older cars, cluster 4 represented newer cars, and cluster 5 represented the oldest cars. In my regression analysis, I identified that variables such as color, displacement, and cylinders were not statistically significant. For buyers, older cars with higher mileage but moderate horsepower may represent a good deal. For sellers, maintaining lower mileage and highlighting horsepower will enhance value. For dealers, mileage and horsepower are the attributes that most significantly drive price variance.

I preformed text extraction and parsing, a more granular form of feature engineering and data wrangling. This method was necessary because most of the data was in string format, yet we needed to represent numerical features such as mileage and price. In contrast, the datasets referenced in the literature reviews did not require such techniques, as their variables already came in numerical format.

Most code will not run if there are spelling errors, so I used chat-gpt to proofread my code and identify spelling and syntax mistakes. I also used it to check for grammar in my report. Additionally, I utilized it to recall specific commands for data cleaning, generating ideas for what cluster algorithms could work based on the characteristics of my data.

Literature Review:

1) This study utilizes a random forest model with 500 decision trees, achieving a training accuracy of 95.82% and a testing accuracy of 83.63%. The model predicted car prices based on features most strongly correlated with price. Numerous variables were considered, including paint color, safety features, and fuel economy. The paper outlined the steps for developing a predictive model, beginning with data collection, and identifying important features. The preprocessing step involved removing missing values and discarding irrelevant or unusable features. The cleaned dataset was then used to apply a random forest model. Zhang et al. tested several classification algorithms, such as a decision tree, logistic regression, random forest, and SVM, but found that random forest preforms best for their prediction task. The dataset comprised information for 370,000 used cars from 40 different brands, sold through an online platform. It included 20 features describing car sales. During preprocessing, certain attributed were deemed insignificant for prediction, such as advertisement name and postal code, therefore were removed. They narrowed the dataset by excluding dealership-sold vehicles and retaining only cars sold privately. They also limited the dataset to cars manufactures between 1863 and 2017, added a variable for car age, and filtered for realistic power values, valid registration dates, and available cars with associated prices. Binary variables were converted to numeric values. The most influential features identified were asking price, kilometers driven, manufactures brand, and vehicle type. The dataset revealed an average price of $11,000 and an average mileage of 125,000 km, and half of the cars being 12 years or older. Although linear regression was initially attempted, random forest provided superior performance and mitigated overfitting, and the regression model achieved less than 75 % accuracy on the training data. The grid search algorithm was employed to optimize the number of trees, with the best accuracy achieved using 500 decision trees.

2. This paper aimed to accurately predict used car prices by testing linear, polynomial, support vector, decision tree, and random forest regression models. Model performance was evaluated using R-squared values, with random forest achieving the highest score of 0.90416. The goal of this research was to determine reasonable price for used cars to prevent overpricing and help buyers and sellers save time and effort. The dataset contained 100,000 used cars from the UK, with each row representing a specific vehicle and each column detailing features such as mileage, model year, and selling price. To refine their analysis, the research focused on Mercedes cars, which accounted for over 13,000 rows. The generated a correlation matrix to analyze relationships between variables, with values ranging from -1 to 1, where 1 indicated the strongest positive correlation. Key variables included year, mileage, tax, miles per gallon, and engine size.

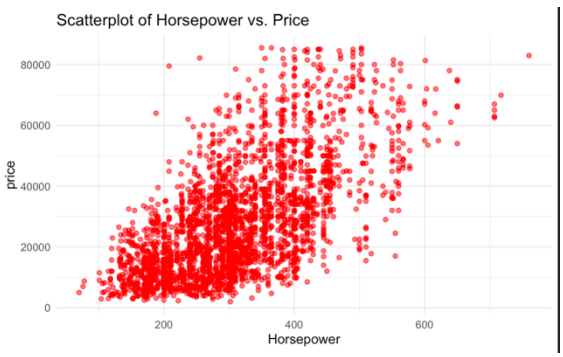
Citations:

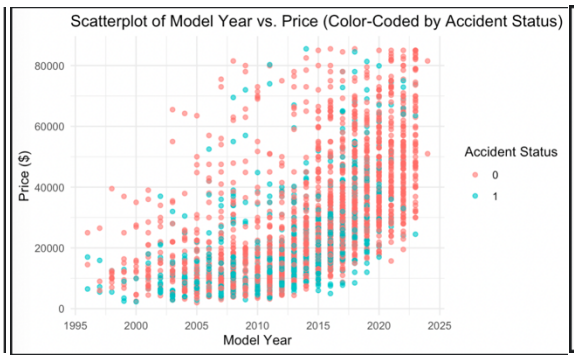
1. Pal, N., Arora, P., Kohli, P., Sundararaman, D., Palakurthy, S.S. (2019). How Much Is My Car Worth? A Methodology for Predicting Used Cars' Prices Using Random Forest. In: Arai, K., Kapoor, S., Bhatia, R. (eds) Advances in Information and Communication Networks. FICC 2018. Advances in Intelligent Systems and Computing, vol 886. Springer, Cham. https://doi.org/10.1007/978-3-030-03402-3_28
2. 2. C. Jin, "Price Prediction of Used Cars Using Machine Learning," 2021 IEEE International Conference on Emergency Science and Information Technology (ICESIT), Chongqing, China, 2021, pp. 223-230, doi: 10.1109/ICESIT53460.2021.9696839. https://ieeexplore.ieee.org/abstract/document/9696839

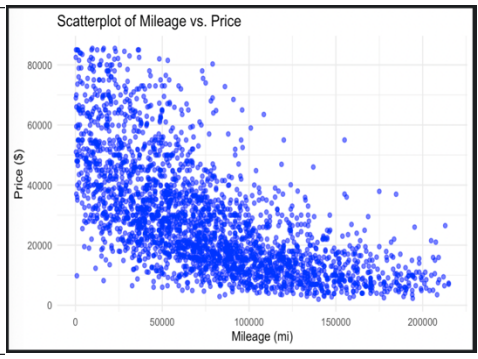
Data Processing and Summary Statistics:

The first step in the data preprocessing phase was creating a new variable "car_age" derived from "model_year" by calculating the absolute difference: 2025 – model_year. To preform text extraction and parsing, I used libraries such as dplyr, stringr, tidyr, and tidyverse, transforming the "engine" variable into multiple features, including horsepower, displacement, engine_type, fuel_type, and cylinders. Next, I removed units of measurement from to convert them into numerical values. For example, I removed the $ symbol from the price variable. For the cylinders feature, I used the str_detect function to extract cylinder counts from the strings, such as converting the string "V6" into numeric value 6. I also encoded variables like "accident", which had yes/no values, into binary values (0 and 1).

Upon plotting these variables, I identified numerous data outliers that heavily skewed the data. To address this, I used the interquartile range (IQR) method to filter out extreme values for price and mileage, focusing on more realistic pricing of used cars. The final preprocessing step involved removing rows with missing values, as most models cannot handle missing data. For instance, some electric cars in the dataset had no horsepower values, and imputing a value such as 0 would distort the results. Scatterplots revealed



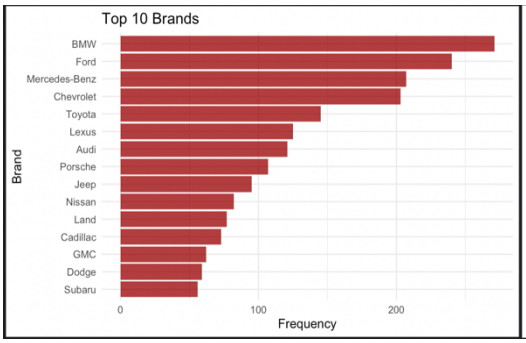


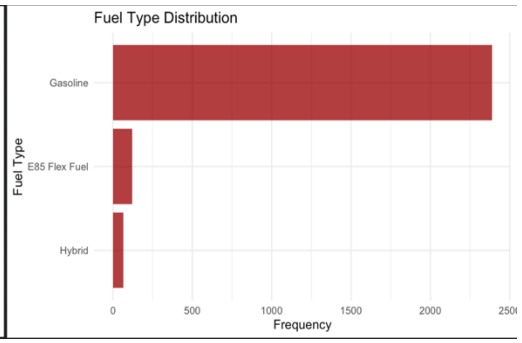


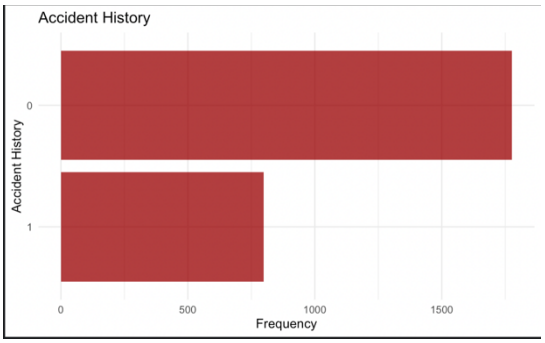
clear trends: cars with higher horsepower, lower mileage, and newer model years tended to have the highest prices.

On average, the dataset's cars had approximately 76,000 miles, an average price of $28,000, an average age of 11 years (2014 make), and an average horsepower of 313.

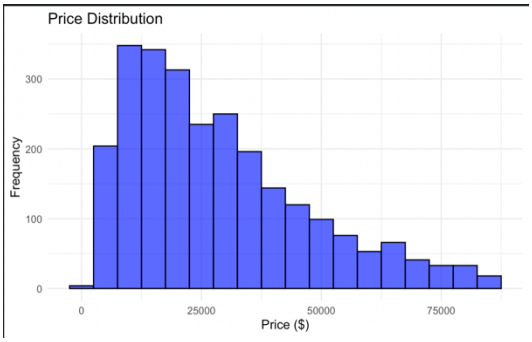
```
   model_year       milage         fuel_type         transmission
Min.   :1996   Min.   :   200   Length:2575        Length:2575
1st Qu.:2010   1st Qu.: 38100   Class :character   Class :character
Median :2015   Median : 69641   Mode  :character   Mode  :character
Mean   :2014   Mean   : 76099
3rd Qu.:2018   3rd Qu.:107000
Max.   :2024   Max.   :215000
     price          car_age        Horsepower     Displacement
Min.   : 2000   Min.   : 1.00   Min.   : 70.0   Min.   :1.000
1st Qu.:14000   1st Qu.: 7.00   1st Qu.:240.0   1st Qu.:2.500
Median :24500   Median :10.00   Median :301.0   Median :3.500
Mean   :28667   Mean   :11.06   Mean   :313.2   Mean   :3.647
3rd Qu.:38600   3rd Qu.:15.00   3rd Qu.:381.0   3rd Qu.:4.600
Max.   :85500   Max.   :29.00   Max.   :760.0   Max.   :8.300
```

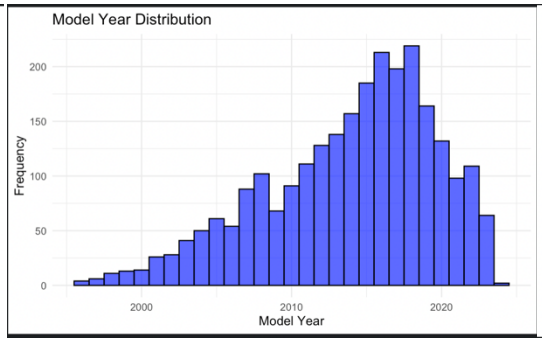


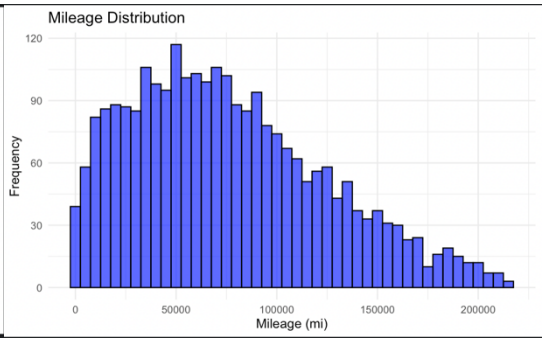




The most popular used cars brands in the dataset are BMW, Ford, Mercedes, Chevrolet, and Toyota, with the vast majority of cars using gasoline as fuel. These cars exhibit a wide range of car prices, although most vehicles are priced below $ 50,000. Since price and mileage show a high correlation (as demonstrated in correlation matrix below), it is not surprising that cars with a broad range of prices also have a wide range of mileage. However, the majority of cars have mileage below 150,000 miles. Additionally, as there are more cars with lower mileage and car age and mileage are also correlated, it follows that most cars in the dataset are relatively newer, typically from the past 10 years. Regarding accident history, approximately one-third of the cars in this

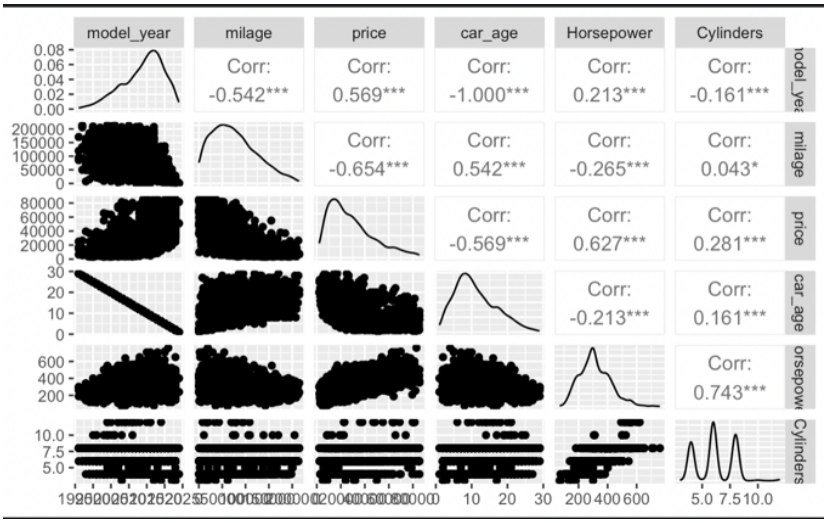






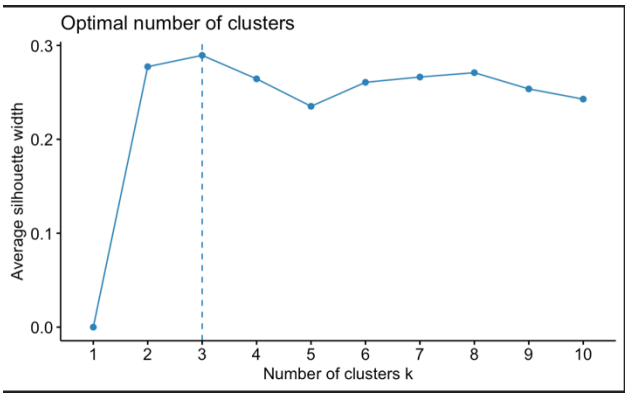
dataset have been involved in an accident.

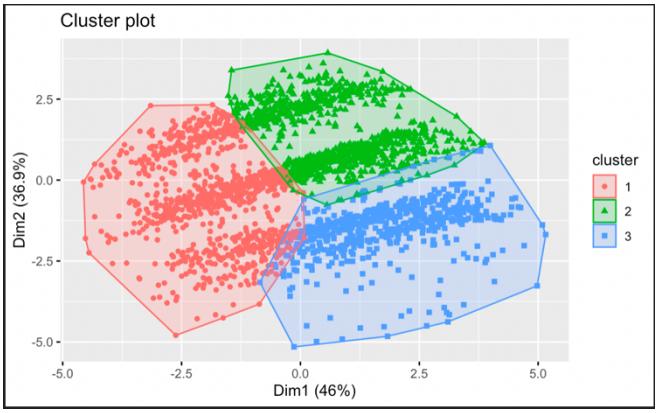
This correlation matrix illustrates the relationships between numerical variables. The highest correlation observed between Cylinders and Horsepower, which is logical as both are engine-related variables. Other notable correlations include mileage vs price, age vs price, mileage vs model year. The higher number of cylinders a car has, the higher the horsepower. Overall, older cars typically tend to accumulate more miles, resulting in lower prices.



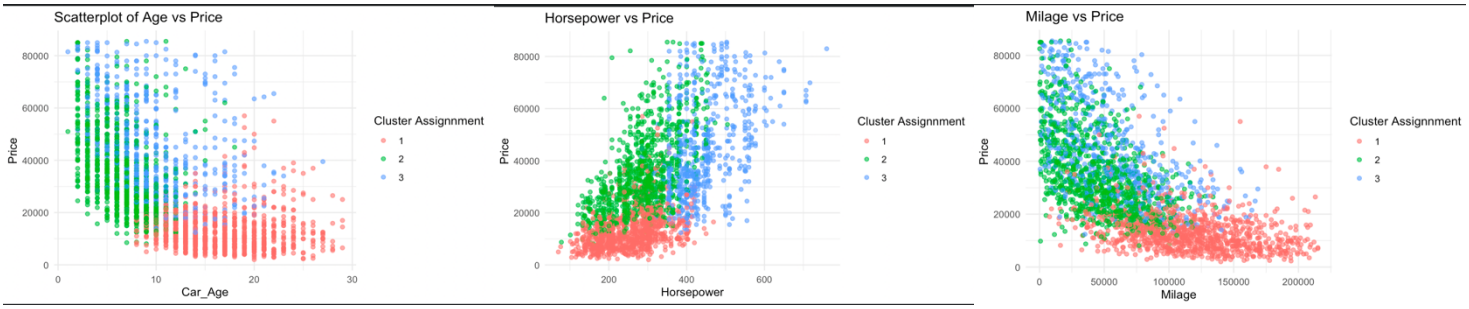
Unsupervised Learning:

Algorithm 1: K-Medoids





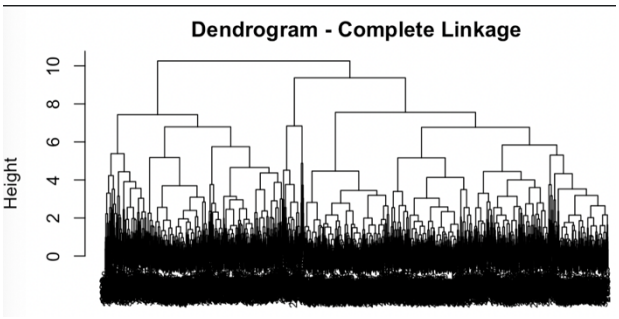
The first step in running a k-medoids algorithm is to find the optimal number of clusters because this algorithm relies on specifying the number of clusters to use. Therefore, I used the packages cluster and factoextra to scale the data and then to create a silhouette plot. The silhouette plot (top left) shows that the optimal number of clusters this algorithm should use is 3. Then, using the same packages, I plotted the data to visualize the clusters (top right). To further visualize the trends of different variables with these assigned clusters, I plotted different variables against price using scatterplots.

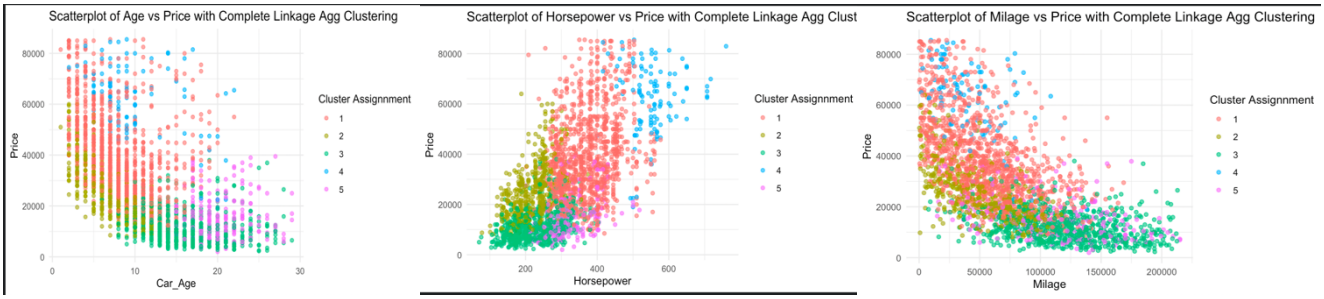


Cluster 3 has the highest horsepower which correspond to higher prices, reflection the premium/luxury nature of these cars. Cluster 2 has mid-level prices and lower horsepower, alighting with mid-range/economy cars. Cluster 1 has the highest mileage and the lowest prices, signaling they are the more affordable cars. Therefore, these clusters are meaningful in terms of their influence on price, the outcome variable. It captures underlying patterns in the data relating to how mileage, horsepower, age can influence of cars in different price ranges. This is useful for supervised learning, because now we know that there are features in this data that can predict car prices very well.

Algorithm 2: Complete Linkage

After applying the complete linkage clustering algorithm to the used cars data, I split the data into 5 clusters based on the dendrogram (right). Complete linkage focuses on the maximum distance between points in clusters that result in more well-separated groups.

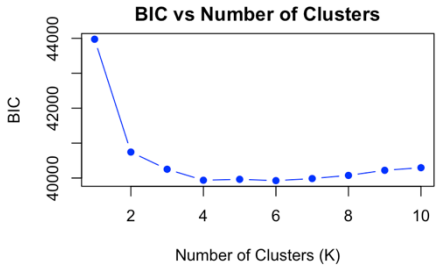


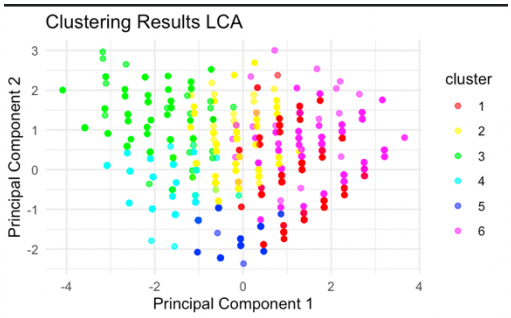


Cluster 1 represents cars that are newer, with relatively higher horsepower, and a variety of different prices. These seem to be the more affordable luxury cars. Cluster 2 represents are the newest cars in the dataset, with relatively lower horsepower, lower mileage, and lower price. These cars seem to be the more mid-range/economy cars. Cluster 3 represents are the older cars, with the lowest horsepower, relatively higher mileage, and the lowest priced cars in the entire dataset. These cars are the most affordable vehicles. Cluster 4 represents have the highest prices and horsepower in the entire dataset, with a wide range of ages. These cars seem to be the most expensive powerful sport and luxury vehicles. Cluster 5 represents the oldest, mid-range horsepower, with a variety of different mileages. The lower prices of these cars remain relatively the same no matter the age, horsepower, and mileage. These seem to be the more reliable and affordable vehicles for buyers.

As a result, these clusters provide a very meaningful insight as to how each feature behaves and how combinations of them are what control and predict the price. This analysis could be useful for supervised techniques and models because it could provide a segmentation of the used car market. For example, cluster 2 is the more mid-range/economy cars which will appeal to a different demographic than cluster 4 which contain more luxury/sports cars. Predictive regression models can incorporate these clusters to recommend optimal pricing strategies tailers to different car markets. Since the algorithm separated these clusters based on economy, luxury, sport, etc., these results could be used in supervised learning by separating these clusters and running different regressions on each car category. It will predict prices more accurately for that price range. Another way is to reduce dimensionally and run a supervised model with the cluster assignment instead of the individual features to improve accuracy and reduce overfitting.

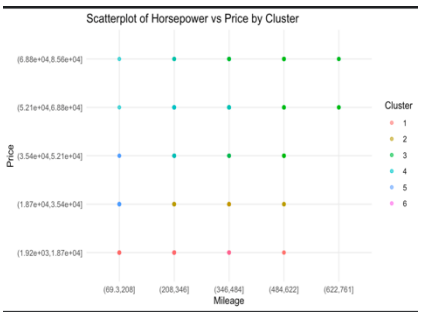
Algorithm 3: LCA Clustering

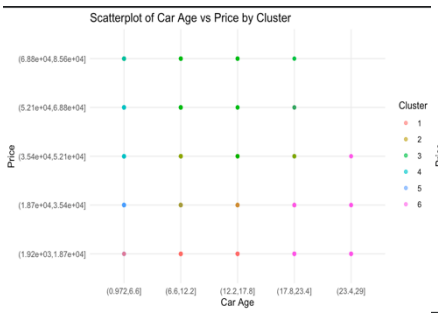


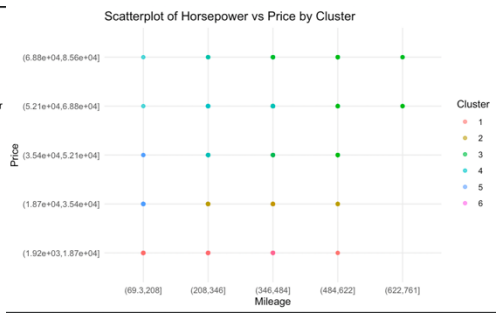


According to the figure (top left), the BIC score is the lowest when there are 6 clusters. Cluster 1 represents cars with lower mileage, younger age, moderate horsepower, and low to moderate price range. It likely consists of newer, less-used economy cars. Cluster 2 represents cars with moderate milage, moderate car age, moderate horsepower, and middle range pricing. It likely represents mid-range vehicles with a relatively moderate feature. Cluster 3 represents cars with higher mileage, older age, lower horsepower, and a lower price range. It likely represented older, more used, and budget friendly vehicles. Cluster 4 represents cars with very high mileage, older age, lower horsepower, and has the lowest price range. It likely represents older, more used, and budget-oriented vehicles. It represents cars that are the cheapest but are nearing the end of their lifespan and are less desirable. Cluster 5 are cars with low mileage, newer age, high horsepower, and the highest price range. These cars are likely performance or luxury vehicles that are appealing for the high-end buyers. Cluster 6 combines moderate mileage, age, and horsepower. It has a moderate to high price range. This cluster likely represents sports cars or high-performance vehicles are not that as brand new as cluster 5.

Therefore, a combination of mileage, car age, and horsepower seem to have a significant effect on used car prices. This algorithm captures latent structures in the data very efficiently, which can be valuable in supervised learning because it can be used to target marketing and pricing strategies for different demographics of buyers.

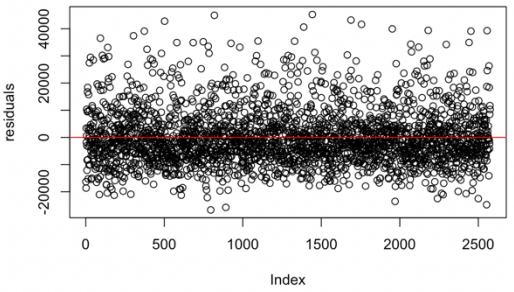






Prediction Models:

Prediction Model 1: OLS Regression



After encoding the categorical variables to make them numeric, I ran the OLS regression model on all features in the data. Using stepwise selection with direction = "both", I selected relevant predictors based on AIC. Next, I trained the linear model using 10-fold cross-validation using the caret package to estimate the model's performance using only statistically significant predictors. Next, I calculated the RMSE and R-squared on the training data to access model fit and accuracy. The RMSE is 100123.88, R-squared value is 0.71, and MAE of 7586.59. For a dataset that is large and complex, 71 % is acceptable. Finally, I plot the residual plot which shows the residuals are scattered around zero, which indicates no obvious patterns or trends. The relevant predictors are brand, fuel type, transmission, interior color, age of car, mileage, horsepower, and whether a car has been in an accident.

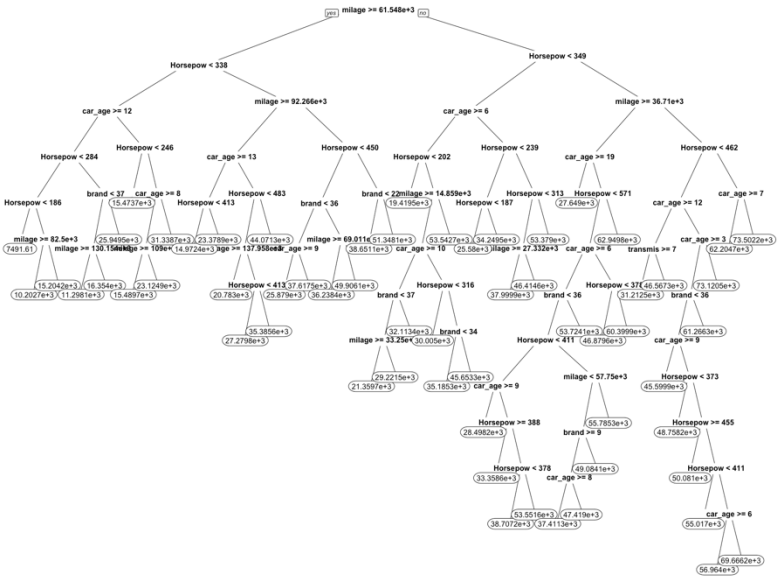
```
Call:
glm(formula = price ~ brand + fuel_type + transmission + ext_col +
    int_col + car_age + milage + Horsepower + Displacement +
    Cylinders + Accident_Label, data = df_cars_numeric)

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)     1.664e+04  2.055e+03   8.095 8.73e-16 ***
brand           1.150e+02  1.555e+01   7.396 1.89e-13 ***
fuel_type       2.082e+03  7.732e+02   2.693 0.007127 **
transmission   -2.058e+02  5.360e+01  -3.840 0.000126 ***
ext_col         2.803e+01  4.300e+01   0.652 0.514435
int_col         1.853e+02  7.982e+01   2.322 0.020304 *
car_age        -8.496e+02  4.544e+01 -18.699  < 2e-16 ***
milage         -1.425e-01  5.264e-03 -27.072  < 2e-16 ***
Horsepower      9.113e+01  3.468e+00  26.280  < 2e-16 ***
Displacement    9.788e+01  3.651e+02   0.268 0.788666
Cylinders      -2.745e+02  3.508e+02  -0.782 0.434048
Accident_Label1 -1.838e+03  4.467e+02  -4.115 4.00e-05 ***
```

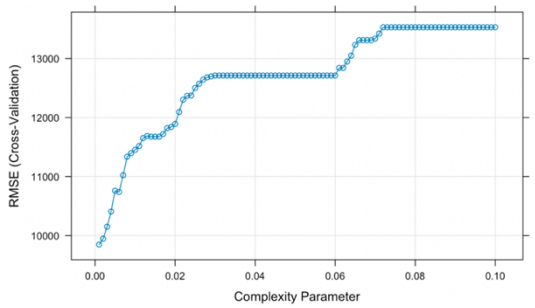
```
Start:  AIC=54811.43
price ~ brand + fuel_type + transmission + int_col + car_age +
    milage + Horsepower + Accident_Label

                Df   Deviance   AIC
<none>              2.6261e+11 54811
- int_col        1 2.6317e+11 54815
- fuel_type      1 2.6344e+11 54818
- transmission   1 2.6412e+11 54824
- Accident_Label 1 2.6435e+11 54826
- brand          1 2.6828e+11 54864
- car_age        1 3.0582e+11 55202
- milage         1 3.4047e+11 55478
- Horsepower     1 4.5073e+11 56200
```

```
Linear Regression

2575 samples
   8 predictor

No pre-processing
Resampling: Cross-Validated (10 fold)
Summary of sample sizes: 2317, 2318, 2316, 2318, 2318, 2318, ...
Resampling results:

  RMSE      Rsquared   MAE
  10123.88  0.7084121  7586.589

Tuning parameter 'intercept' was held constant at a value of TRUE
```

Prediction Model 2: Regression Tree

First, I plotted a regression tree using ISLR package using a training percentage of 75 and the variables that are statistically significant. Next, I tunned the model using cross-validation to asses model performance and avoid overfitting. I then used the cp (complexity parameter) to find out the optimal size of the tree. I used 10-fold cross validation to test different cp values and selected the lowest cross-validation error. The best cp turned out to be 0.001 with RMSE value 9292.523. Then, I refitted the model using the optimal cp value. The R-squared value is 0.748.

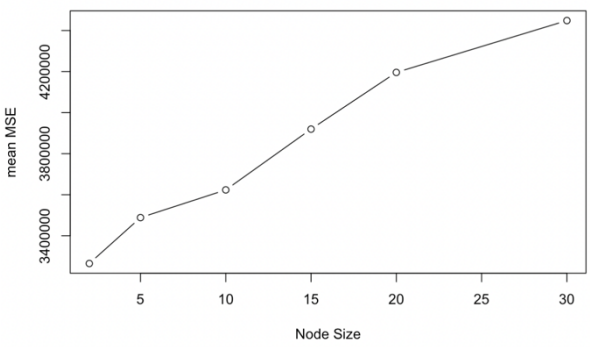


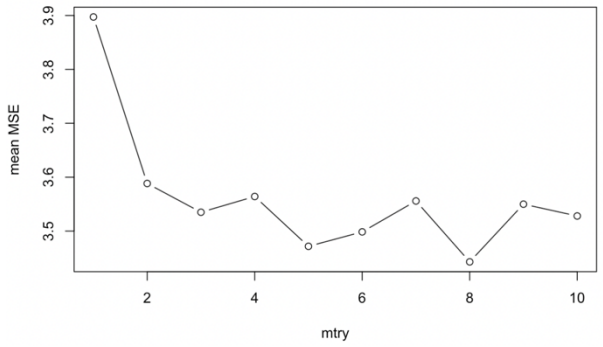


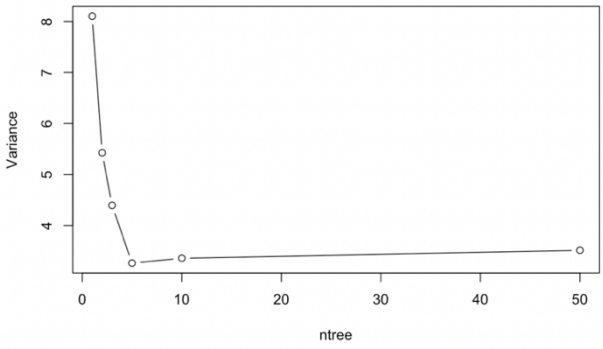
Prediction Model 3: Random Forest

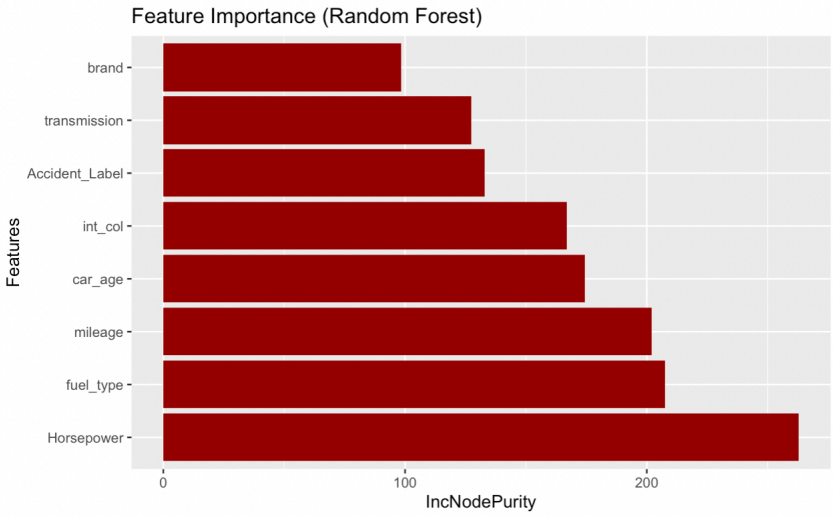
When fitting this random forest model, I used the regression model with all statistically significant variables including brand, fuel type, transmission, interior color, age of car, mileage, horsepower, and whether a car has been in an accident. I fitted the model with test and train data and got an MSE of 1652272.1.

To tune this model, I found the optimal nodesize to control the tree depth by adjusting the minimum number of observations required in terminal nodes. The optimal node size is 1 (top right) Next, I tuned mtry to control for the number of predictors considered for each spilt. The optimal mtry value is 8 (middle right). Then, I tuned ntree to control the number of trees in the forest. The optimal value for ntree is 5 (bottom right).



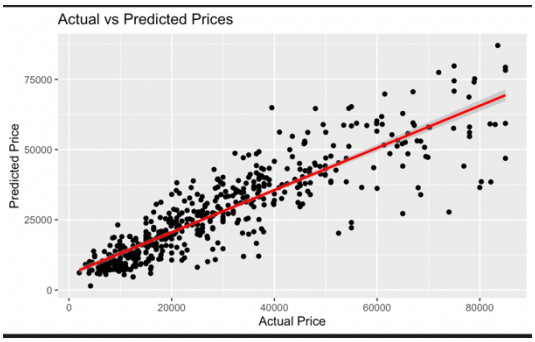






The most important feature in prediction is Horsepower, followed by fuel type, mileage, car age, interior color, car accidents, transmission, and brand. The model has an R-squared value of 0.887.

Prediction Model 4: SVM



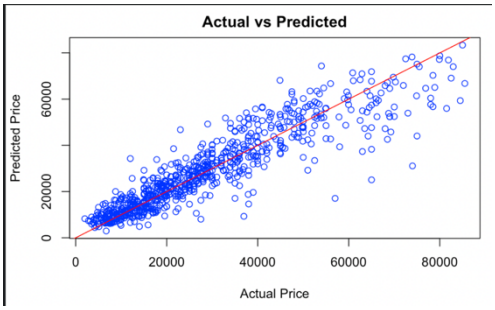
To fit a SVM model, I first split the data into training and test sets with 80 percent of data used for training. Next, I set a tuning grid for the radial basis function kernel SVM. The two parameters I tunned was C (the regularization parameter that balances bias and variance), and Sigma (that controls the width of radial basic function kernel). The expand.grid() function generated a grid of these values and uses 5-fold cross-validation to find the best combination. RMSE was used to select the optimal model using the smallest value. The final values used for the model were sigma = 0.04 and C = 10. It has a RMSE = 8578.87 and MAE = 5894.756. The model has an r-squared value of 0.7685.

Predictive Model 5: Gaussian Boosting Regression

To fit this Gaussian boosting regression, I tuned this model to find nrounds, to find the number of boosting iterations and determine how many trees get added to the model. Second, I tuned max_depth, to find the maximum depth of the trees and controls complexity of the individual trees. Third, I tuned for eta (learning rate), that controls each tree's contribution. Lower values reduce the impact of each tree but requires more rounds to converge. Fourth, I tuned gamma, which is the regularization term that controls for the minimum loss reduction required to make further partitions on a leaf node. Fifth, I tuned colsample_by_tree, which the fraction of features to be randomly sampled for each tree. Sixth, I tuned min_child_weight, which is the minimum sum of instance weights (Hessian) needed in a child. Lastly, I tuned subsample, which is the fraction of samples to be randomly selected for each tree. I used 5-fold cross validation and set ranges of possible values 4, 6, and 8 for max_depth. For eta, 0.01, 0.1, and 0.3. For subsample 0.6, 0.8, and 1.

	nrounds <dbl>	max_depth <dbl>	eta <dbl>	gamma <dbl>	colsample_bytree <dbl>	min_child_weight <dbl>	subsample <dbl>
221	100	6	0.1	0	0.6	2	0.8

Using these optimal values (above figure), I used the train model to predict the test data using 5-fold cross validation. The RMSE is 7523.21 and R-squared value is 0.8322. This model fit very accurately, according to the residual plot (right).



OLS regression produced a 0.71 r-squared value, regression tree produced 0.748, random forest produced 0.887, SVM produced 0.7685, and Gaussian Boosting regression produced 0.8322. The random forest was the best preforming model in terms of prediction.

Open-Ended Question - Predicting Original Prices:

Most cars depreciate in value an exponential rate, meaning the older the car gets the less money it is worth. It depends on a variety of different factors: mileage, age, etc. Many cars have a depreciation rate in the range of 10 - 20 percent every year. The goal is to calculate the original price, when it was brand new, of each used car in the dataset. The formula is: **price/(1-depreciation_rate/100) ^car_age = estimated_original_price.** Now, we can then use the depreciation formula to extrapolate the original price.

The biggest challenge and limitation are that the rate of depreciation varies depending on the brand and make of the car. A Mercedes may depreciate at a rate of 20 percent every year because of its high maintenance cost, but a Toyota might depreciate by only 10 percent since it is a more reliable car and doesn't have as much maintenance. Since there are too many unique makes in this dataset, I researched the depreciate rate based on each brand in the data since there is a finite amount of them in this dataset. Another challenge is inflation, which skews the price prediction model because when the used cars in this dataset were new, they could be depreciating at different rates every year depending on inflation.

The **2019 Porsche** cayenne base model has a price of $42,499 at 59,000 miles and an average depreciation rate of 8 percent. The predicted original price calculated using the formula above turned out to be $70,089. When searching the price of this car new, in 2019 the car new cost between $66,950 and $84,150. Since this porshe is a base model it most likely cost closer to 68k, which is relatively close to 70k.

The **2014 Lexus LS 460 L** has 106,731 miles, a 10 % average depreciation rate, and costs 30k. The estimated original price of this car turned out to be around 95k. When searching for the price of this car, the MSRP starts at 80k. But this Lexus has the highest horsepower package this model offers, which is going to make it significantly more expensive, since horsepower is a significant predictor in price. Therefore, the original price makes sense.

The **2011 Volvo XC90** 3.3 has 88,300 miles, an average depreciation rate of 12 percent, and costs $11,500. The formula produced an original cost of $86,855. This car's MSRP starts at $57,400, but it has a bigger engine and horsepower, meaning that it should more expensive than the MSRP. It also does not mention what kind of additional packages or customizations this car could potentially have. Overall, the model slightly over-predicted the price of this car.

General Requirements:

My report is easy to read as it highlights important figures in the text, so readers can easily follow along. All model fittings are accurately reported and resented with provided reasoning for all decisions made. It is written in a manner where readers can easily understand what data I am referring to and how I use it to formulate and explain my models. All of my plots and tables are compact and clearly incorporated throughout my report. All irrelevant code and output are not included in my report.