Final Project - Step 3

Riaz Ahmed Tamim Ansari

2023-11-18

Introduction

The annual inflation rate for the United States was 3.7% for the 12 months ended September, according to U.S. Labor Department data published on Oct. 12, 2023. Nowadays, owning a vehicle has almost become mandatory. During these times, it becomes prudent for the consumers shopping for vehicles to go for used ones, instead of new vehicles to save some dollars.

Due to various factors like supply chain issues and unavailability of new vehicles, we have seen unstable used car price increase during the pandemic times and it is still hovering around there for a long time.

Addressed problem statement

This paper aims to find out the used car selling price trend and what drives it. It covers the following analysis,

1. Popular makes
2. Adoption of electric/hybrid cars
3. Relationship between selling price and other predictor variables
4. Analysis of selling price
5. Preference of different type of vehicles
6. Location where most of the cars were sold
7. Predicting the rates of the used car by using a basic model

Research questions

1. What determines the cost of the used car?
2. A car has its own specifications. How much contribution does each has to offer for the price?
3. Is the price driven by the latitude and longitude or location where it is sold?
4. Which brands were popular among the consumers?
5. Did consumers preferred to use smaller vehicles more, in order to have gas efficiency?

6. How was the sale for the vehicles other than that ran on Gas/Diesel?

7. How was the price trend during pandemic times?

8. When will the used car prices go down?

Approach to address the problem statement

From the available data, identify what drives the value of the used car, thereby its cost. We would be able to answer most of the above questions by manipulating the available data. Different datasets were combined together to get a holistic picture of the used car costs. After cleaning the dataset, various R libraries, plots and packages has been used and the following sections explains them in detail.

Analysis

This can be broadly broken down to gathering data, manipulating it and using the required packages/graphs/plots and models to uncover the hidden information.

Gathering Data

A total of 3 datasets has been referred to,

1. Used car listing dataset

Data collected year : 2023 This is a comprehensive dataset of used cars sold from 1921 to 2017 in U.S. This dataset contains two files, one just has latitude and longitude information with the region and other file containing the specifications of the car and the price it sold for. I would be using combining both of these files and lookup with latitude/longitude information to match with the region where it got sold.

```
usedcar_lat_lon <- read.delim("C:\\Users\\Riaz\\Desktop\\MSDS\\Introduction to Statistics\\Week8&9\\Data
str(usedcar_lat_lon)
```

```
## 'data.frame':    372 obs. of  3 variables:
##  $ region   : chr  "north jersey" "northern WI" "modesto" "susanville" ...
##  $ latitude : num  35.7 40.8 37.6 40.4 40.8 ...
##  $ longitude: num  -80.3 -124.2 -121 -120.7 -111.9 ...
```

```
head(usedcar_lat_lon)
```

```
##              region latitude  longitude
## 1      north jersey 35.73097  -80.31533
## 2       northern WI 40.79069 -124.16737
## 3           modesto 37.63910 -120.99688
## 4        susanville 40.42741 -120.65370
## 5    salt lake city 40.75962 -111.88680
## 6 gulfport / biloxi 29.42117  -94.69148
```

```
usedcar_train <- read.delim("C:\\Users\\Riaz\\Desktop\\MSDS\\Introduction to Statistics\\Week8&9\\Data\\
str(usedcar_train)
```

```
## 'data.frame':    27532 obs. of  16 variables:
##  $ id          : int  0 1 2 3 4 5 6 7 8 9 ...
##  $ region      : chr  "nashville" "state college" "wichita" "albany" ...
##  $ year        : int  1949 2013 1998 2014 2005 2013 2014 2006 2013 2012 ...
##  $ manufacturer: chr  "bmw" "toyota" "ford" "ford" ...
##  $ condition   : chr  "excellent" "fair" "good" "excellent" ...
##  $ cylinders   : chr  "6 cylinders" "8 cylinders" "6 cylinders" "4 cylinders" ...
##  $ fuel        : chr  "gas" "gas" "gas" "gas" ...
##  $ odometer    : int  115148 172038 152492 104118 144554 133208 131104 158776 127167 150319 ...
##  $ title_status: chr  "clean" "clean" "clean" "clean" ...
##  $ transmission: chr  "manual" "automatic" "automatic" "manual" ...
##  $ drive       : chr  "rwd" "rwd" "fwd" "fwd" ...
##  $ size        : chr  "mid-size" "full-size" "full-size" "mid-size" ...
##  $ type        : chr  "convertible" "sedan" "SUV" "SUV" ...
##  $ paint_color : chr  "orange" "silver" "silver" "blue" ...
##  $ state       : chr  NA "pa" "ks" "ny" ...
##  $ price       : int  27587 4724 10931 16553 5158 7941 5860 2799 6036 7341 ...
```

```
head(usedcar_train)
```

```
##   id                  region year manufacturer condition   cylinders fuel
## 1  0               nashville 1949          bmw excellent 6 cylinders  gas
## 2  1           state college 2013       toyota      fair 8 cylinders  gas
## 3  2                 wichita 1998         ford      good 6 cylinders  gas
## 4  3                  albany 2014         ford excellent 4 cylinders  gas
## 5  4                 redding 2005         ford excellent 6 cylinders  gas
## 6  5 florence / muscle shoals 2013       nissan      good 4 cylinders  gas
##   odometer title_status transmission drive      size        type paint_color
## 1   115148        clean       manual   rwd  mid-size convertible      orange
## 2   172038        clean    automatic   rwd full-size       sedan      silver
## 3   152492        clean    automatic   fwd full-size         SUV      silver
## 4   104118        clean       manual   fwd  mid-size         SUV        blue
## 5   144554        clean       manual   fwd  mid-size       sedan         red
## 6   133208        clean    automatic   4wd full-size         SUV       black
##   state price
## 1  <NA> 27587
## 2    pa  4724
## 3    ks 10931
## 4    ny 16553
## 5    ca  5158
## 6    al  7941
```

2. US car prices

Data collected year : 2019 This dataset contains the following variables which are the specifications of the car. I am planning to use this primarily as it has a lot more extra details about the car. For eg, car width/height,horsepower etc

car_ID,symboling,CarName,fueltype,aspiration,doornumber,carbody,drivewheel,enginelocation,wheelbase,carlength carwidth,carheight,curbweight,enginetype,cylindernumber,enginesize,fuelsystem,boreratio,stroke,compressionratio horsepower,peakrpm,citympg,highwaympg,price

```
carprice <- read.delim("C:\\Users\\Riaz\\Desktop\\MSDS\\Introduction to Statistics\\Week8&9\\Data\\Data
str(carprice)
```

```
## 'data.frame':    205 obs. of  26 variables:
##  $ car_ID          : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ symboling       : int  3 3 1 2 2 2 1 1 1 0 ...
##  $ CarName         : chr  "alfa-romero giulia" "alfa-romero stelvio" "alfa-romero Quadrifoglio" "audi
##  $ fueltype        : chr  "gas" "gas" "gas" "gas" ...
##  $ aspiration      : chr  "std" "std" "std" "std" ...
##  $ doornumber      : chr  "two" "two" "two" "four" ...
##  $ carbody         : chr  "convertible" "convertible" "hatchback" "sedan" ...
##  $ drivewheel      : chr  "rwd" "rwd" "rwd" "fwd" ...
##  $ enginelocation  : chr  "front" "front" "front" "front" ...
##  $ wheelbase       : num  88.6 88.6 94.5 99.8 99.4 ...
##  $ carlength       : num  169 169 171 177 177 ...
##  $ carwidth        : num  64.1 64.1 65.5 66.2 66.4 66.3 71.4 71.4 71.4 67.9 ...
##  $ carheight       : num  48.8 48.8 52.4 54.3 54.3 53.1 55.7 55.7 55.9 52 ...
##  $ curbweight      : int  2548 2548 2823 2337 2824 2507 2844 2954 3086 3053 ...
##  $ enginetype      : chr  "dohc" "dohc" "ohcv" "ohc" ...
##  $ cylindernumber  : chr  "four" "four" "six" "four" ...
##  $ enginesize      : int  130 130 152 109 136 136 136 136 131 131 ...
##  $ fuelsystem      : chr  "mpfi" "mpfi" "mpfi" "mpfi" ...
##  $ boreratio       : num  3.47 3.47 2.68 3.19 3.19 3.19 3.19 3.19 3.13 3.13 ...
##  $ stroke          : num  2.68 2.68 3.47 3.4 3.4 3.4 3.4 3.4 3.4 3.4 ...
##  $ compressionratio: num  9 9 9 10 8 8.5 8.5 8.5 8.3 7 ...
##  $ horsepower      : int  111 111 154 102 115 110 110 110 140 160 ...
##  $ peakrpm         : int  5000 5000 5000 5500 5500 5500 5500 5500 5500 5500 ...
##  $ citympg         : int  21 21 19 24 18 19 19 19 17 16 ...
##  $ highwaympg      : int  27 27 26 30 22 25 25 25 20 22 ...
##  $ price           : num  13495 16500 16500 13950 17450 ...
```

```
head(carprice)
```

```
##   car_ID symboling                 CarName fueltype aspiration doornumber
## 1      1         3       alfa-romero giulia      gas        std        two
## 2      2         3      alfa-romero stelvio      gas        std        two
## 3      3         1 alfa-romero Quadrifoglio      gas        std        two
## 4      4         2              audi 100 ls      gas        std       four
## 5      5         2               audi 100ls      gas        std       four
## 6      6         2                 audi fox      gas        std        two
##       carbody drivewheel enginelocation wheelbase carlength carwidth carheight
## 1 convertible        rwd          front      88.6     168.8     64.1      48.8
## 2 convertible        rwd          front      88.6     168.8     64.1      48.8
## 3   hatchback        rwd          front      94.5     171.2     65.5      52.4
## 4       sedan        fwd          front      99.8     176.6     66.2      54.3
## 5       sedan        4wd          front      99.4     176.6     66.4      54.3
## 6       sedan        fwd          front      99.8     177.3     66.3      53.1
##   curbweight enginetype cylindernumber enginesize fuelsystem boreratio stroke
## 1       2548       dohc           four        130       mpfi      3.47   2.68
## 2       2548       dohc           four        130       mpfi      3.47   2.68
## 3       2823       ohcv            six        152       mpfi      2.68   3.47
## 4       2337        ohc           four        109       mpfi      3.19   3.40
```

```
## 5       2824        ohc           five        136       mpfi      3.19   3.40
## 6       2507        ohc           five        136       mpfi      3.19   3.40
##   compressionratio horsepower peakrpm citympg highwaympg price
## 1              9.0        111    5000      21         27 13495
## 2              9.0        111    5000      21         27 16500
## 3              9.0        154    5000      19         26 16500
## 4             10.0        102    5500      24         30 13950
## 5              8.0        115    5500      18         22 17450
## 6              8.5        110    5500      19         25 15250
```

3. US used car sales data

Data collected year : 2020 This is a data set for used car sales in the US. Approximately, 160k sales records over a period of 20 months in 2019 and 2020. I would be particularly interested in finding out the prices during the pandemic times, as most part of the data corresponds to the part of the time period. Following are the variables we have in here,

ID,pricesold,yearsold,zipcode,Mileage,Make,Model,Year,Trim,Engine,BodyType,NumCylinders,DriveType

```
carsales <- read.delim("C:\\Users\\Riaz\\Desktop\\MSDS\\Introduction to Statistics\\Week8&9\\Data\\Data3
str(carsales)
```

```
## 'data.frame':    122144 obs. of  13 variables:
##  $ ID          : int  137178 96705 119660 80773 64287 132695 132829 5250 29023 80293 ...
##  $ pricesold   : int  7500 15000 8750 11600 44000 950 950 70000 1330 25200 ...
##  $ yearsold    : int  2020 2019 2020 2019 2019 2020 2020 2019 2019 2019 ...
##  $ zipcode     : chr  "786**" "81006" "33449" "07852" ...
##  $ Mileage     : int  84430 0 55000 97200 40703 71300 71300 6500 167000 15000 ...
##  $ Make        : chr  "Ford" "Replica/Kit Makes" "Jaguar" "Ford" ...
##  $ Model       : chr  "Mustang" "Jaguar Beck Lister" "XJS" "Mustang" ...
##  $ Year        : int  1988 1958 1995 1968 2002 1965 1965 1997 2001 1970 ...
##  $ Trim        : chr  "LX" NA "2+2 Cabriolet" "Stock" ...
##  $ Engine      : chr  "5.0L Gas V8" "383 Fuel injected" "4.0L In-Line 6 Cylinder" "289 cu. in. V8" .
##  $ BodyType    : chr  "Sedan" "Convertible" "Convertible" "Coupe" ...
##  $ NumCylinders: int  0 8 6 8 6 0 0 0 4 0 ...
##  $ DriveType   : chr  "RWD" "RWD" "RWD" "RWD" ...
```

```
head(carsales)
```

```
##       ID pricesold yearsold zipcode Mileage              Make
## 1 137178      7500     2020   786**   84430              Ford
## 2  96705     15000     2019   81006       0 Replica/Kit Makes
## 3 119660      8750     2020   33449   55000            Jaguar
## 4  80773     11600     2019   07852   97200              Ford
## 5  64287     44000     2019   07728   40703           Porsche
## 6 132695       950     2020   462**   71300           Mercury
##                Model Year          Trim                Engine    BodyType
## 1            Mustang 1988            LX            5.0L Gas V8       Sedan
## 2 Jaguar Beck Lister 1958          <NA>      383 Fuel injected Convertible
## 3                XJS 1995 2+2 Cabriolet 4.0L In-Line 6 Cylinder Convertible
## 4            Mustang 1968         Stock          289 cu. in. V8       Coupe
## 5                911 2002    Turbo X-50                   3.6L       Coupe
## 6          Montclair 1965          <NA>              NO ENGINE       Sedan
```

```
##   NumCylinders DriveType
## 1            0       RWD
## 2            8       RWD
## 3            6       RWD
## 4            8       RWD
## 5            6       AWD
## 6            0       RWD
```

We are able to observe some missing data, which would be dropped before computing. Reason being, it would be misleading for some missing variables like the manufactured year or the fuel type to be derived from the measure of central tendency.

Required Packages

BaseR, readr,fread,readxl - To read the datasets and load to dataframe.
ggplot2,ggally - For graphing capabilities.
tidyverse,plyr,dplyr,purrr,stringr - For data munging capabilities and to slice and dice
lubridate - For manipulation of dates.
QuantPsyc,Metrics - For data analysis.

Plots and Table Needs

Base R plots ggplot2,lattice - These packages will be used for doing Exploratory Data Analysis and plotting graphs. For example, we can plot a bar plot between sale of gas vs non gas vehicles and see which one sold the most. Manipulation can be done by grouping the year column and check whether it got increased over the next years.

Scatter plots - Scatterplots will help in determining how a variable changes with respect to another variable. For example using this, we can identify how the car price trend varies with manufactured year.

Histograms - Histograms will be used to identify the frequency of the variable and to understand how the data is distributed. It will also say what values occur the most. Say for example, what would be the model of cars which got sold most.

Box plots - Will be used to identify the InterQuartile Range. This will be useful in determining where the values are the most. Say for example, we can check where most of the car prices lie. One another feature is to identify the outliers.

QQ plots - Determining if the data is normally distributed.

Questions for future steps

1. How to do data cleansing.
2. How to handle the missing values.
3. What are all the relevant variables to be chosen for plotting purposes.
4. How to join data from different dataframes.

Manipulating the data - Importing and cleaning data

As seen in the above commands, we have already imported the data using read.delim function. Proceeding to clean the data.

Finding if the data is having any NA values,

```
sum(is.na(usedcar_lat_lon))
```

```
## [1] 0
```

```
sum(is.na(usedcar_train))
```

```
## [1] 5455
```

```
sum(is.na(carprice))
```

```
## [1] 0
```

```
sum(is.na(carsales))
```

```
## [1] 123046
```

There are few NA values in above datasets. Finding which columns are having NA values
Dropping columns Trim as we are not interested in using that variable and confirming no NA values are there.
Removing the years which doesnt make any sense like less than 4 digit year values and choosing a more appropriate values between 1950 to present.

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
sapply(carsales, function(x) sum(is.na(x)))
```

```
##           ID    pricesold     yearsold      zipcode      Mileage         Make
##            0            0            0          909            0            0
##        Model         Year         Trim       Engine     BodyType NumCylinders
##          573            0        48893        27051        20782            0
##    DriveType
##        24838
```

```r
sapply(usedcar_train, function(x) sum(is.na(x)))
```

```
##           id       region         year manufacturer    condition    cylinders
##            0            0            0            0            0            0
##         fuel     odometer title_status transmission        drive         size
##         1239            0          456            0            0            0
##         type  paint_color        state        price
##          456            0         3304            0
```

```r
carsales <- carsales[-c(9)]
carsales_exclude <- na.exclude(carsales)
usedcar_train <- na.exclude(usedcar_train)
carsales <- na.exclude(carsales)
sapply(carsales_exclude, function(x) sum(is.na(x)))
```

```
##           ID    pricesold     yearsold      zipcode      Mileage         Make
##            0            0            0            0            0            0
##        Model         Year       Engine     BodyType NumCylinders    DriveType
##            0            0            0            0            0            0
```

```r
sapply(usedcar_train, function(x) sum(is.na(x)))
```

```
##           id       region         year manufacturer    condition    cylinders
##            0            0            0            0            0            0
##         fuel     odometer title_status transmission        drive         size
##            0            0            0            0            0            0
##         type  paint_color        state        price
##            0            0            0            0
```

```r
sapply(carsales, function(x) sum(is.na(x)))
```

```
##           ID    pricesold     yearsold      zipcode      Mileage         Make
##            0            0            0            0            0            0
##        Model         Year       Engine     BodyType NumCylinders    DriveType
##            0            0            0            0            0            0
```

```r
carsales_exclude_year <- carsales_exclude %>% filter(nchar(as.character(Year)) == 4 & Year > 1950 & Yea
```

Final data set

After cleansing the dataset, it looks like,

```r
head(carsales_exclude_year)
```

```
##       ID pricesold yearsold zipcode Mileage              Make
## 1 137178      7500     2020   786**   84430              Ford
## 2  96705     15000     2019   81006       0 Replica/Kit Makes
## 3 119660      8750     2020   33449   55000            Jaguar
## 4  80773     11600     2019   07852   97200              Ford
```

```
## 5  64287     44000     2019   07728   40703           Porsche
## 6 132695       950     2020   462**   71300           Mercury
##                Model Year                  Engine    BodyType NumCylinders
## 1            Mustang 1988              5.0L Gas V8       Sedan            0
## 2 Jaguar Beck Lister 1958       383 Fuel injected Convertible            8
## 3                XJS 1995 4.0L In-Line 6 Cylinder Convertible            6
## 4            Mustang 1968          289 cu. in. V8       Coupe            8
## 5                911 2002                   3.6L       Coupe            6
## 6          Montclair 1965               NO ENGINE       Sedan            0
##   DriveType
## 1       RWD
## 2       RWD
## 3       RWD
## 4       RWD
## 5       AWD
## 6       RWD
```

```
head(usedcar_lat_lon)
```

```
##              region latitude  longitude
## 1      north jersey 35.73097  -80.31533
## 2       northern WI 40.79069 -124.16737
## 3           modesto 37.63910 -120.99688
## 4        susanville 40.42741 -120.65370
## 5    salt lake city 40.75962 -111.88680
## 6 gulfport / biloxi 29.42117  -94.69148
```

```
head(usedcar_train)
```

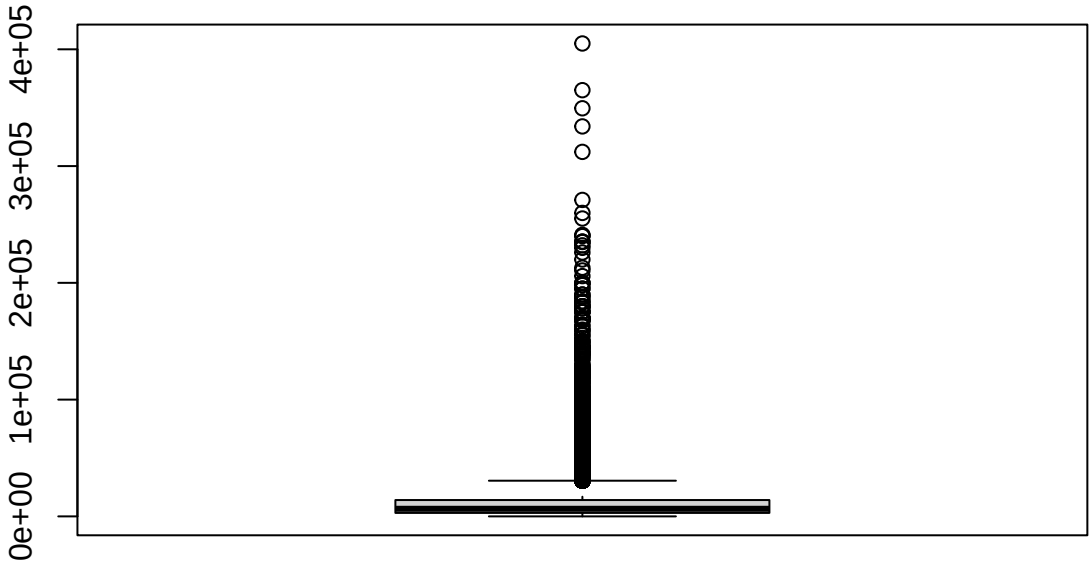
```
##   id                   region year manufacturer condition   cylinders fuel
## 2  1            state college 2013       toyota      fair 8 cylinders  gas
## 3  2                  wichita 1998         ford      good 6 cylinders  gas
## 4  3                   albany 2014         ford excellent 4 cylinders  gas
## 5  4                  redding 2005         ford excellent 6 cylinders  gas
## 6  5 florence / muscle shoals 2013       nissan      good 4 cylinders  gas
## 7  6             oregon coast 2014   volkswagen excellent 6 cylinders  gas
##   odometer title_status transmission drive      size  type paint_color state
## 2   172038        clean    automatic   rwd full-size sedan      silver    pa
## 3   152492        clean    automatic   fwd full-size   SUV      silver    ks
## 4   104118        clean       manual   fwd  mid-size   SUV        blue    ny
## 5   144554        clean       manual   fwd  mid-size sedan         red    ca
## 6   133208        clean    automatic   4wd full-size   SUV       black    al
## 7   131104        clean    automatic   fwd  mid-size sedan        blue    or
##   price
## 2  4724
## 3 10931
## 4 16553
## 5  5158
## 6  7941
## 7  5860
```

Analysis of the dataset to answer the questions,

Slicing and dicing of data

Doing a boxplot analysis to see how the data is looking for pricesold,

```
boxplot(carsales_exclude_year$pricesold)
```



Looking at the graph there are a lot of outliers above 100K$. Finding the outliers in pricesold and mileage columns of the data.

```
range(carsales_exclude_year$pricesold)
```

```
## [1]      0 404990
```

Removing the values more than 100K and storing in a different dataframe for further processing, I am considering these as outliers as the vehicles above 100K in cost are only 165 which is a very small percentage of the total values.

I am also considering outliers for mileage as > 300K, as that data is only 1449 rows which is a very small amount as percentage of total values.

```
carsales_exclude_morethan100k <- carsales_exclude_year %>% filter(pricesold > 100000)
nrow(carsales_exclude_morethan100k)
```

```
## [1] 165
```

```
nrow(carsales_exclude_year %>% filter(Mileage > 300000))
```

```
## [1] 1449
```

```
carsales_exclude <- carsales_exclude_year %>% filter(Mileage < 300000)
carsales_exclude <- carsales_exclude_year %>% filter(pricesold < 100000)

nrow(carsales_exclude)
```

```
## [1] 77902
```

Different ways of looking at this data

I have merged with another dataset and tried to get the lat/lon information of the region where the vehicles are sold the most. Also, various plots and graphs has been used to extract the information. I have seen the data through various variables like manufactured year, make/model, mileage, popularity of the makes, premium makes which command more than 100k, adoption of electric/hybrid vehicles, mean value of the price sold etc.

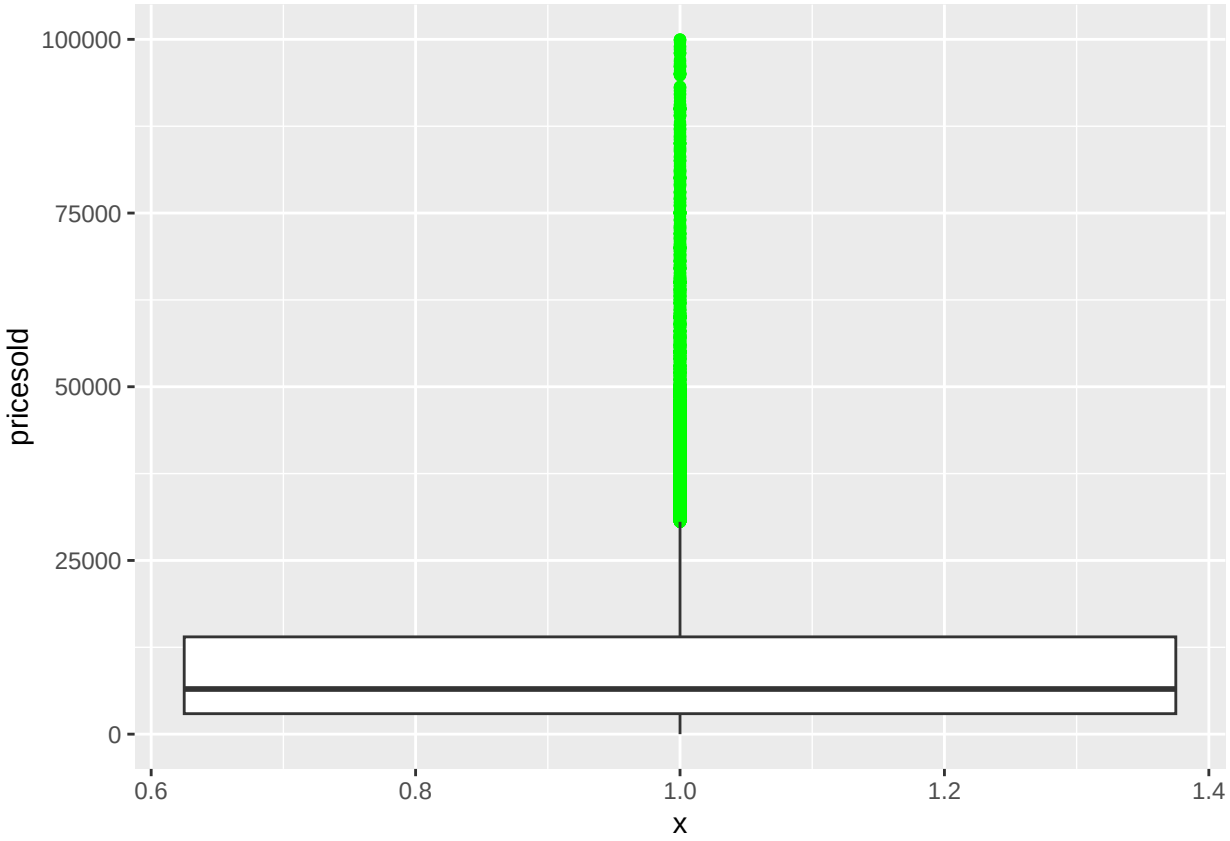
I have also used various functions to arrive at my analysis like groupby, unique, arrange, str_detect, regex, filter conditions etc.

Plots and tables

I have used a combination of different plots like boxplots, barplots and scatter plots to explain the analysis, which could be seen throughout this document.

Now we have the dataset with car prices less than 100K, doing a boxplot on price sold,

```
library(ggplot2)
ggplot(carsales_exclude, aes(y=pricesold, x=1)) + geom_boxplot(outlier.colour="green") +  scale_y_conti
```



The box plot says that the mean of price sold is around $6500 and 75% of vehicles lying within the cost of $13,000. Also the box plot upper whisker indicates that the price ranges from $13000 to $32000.

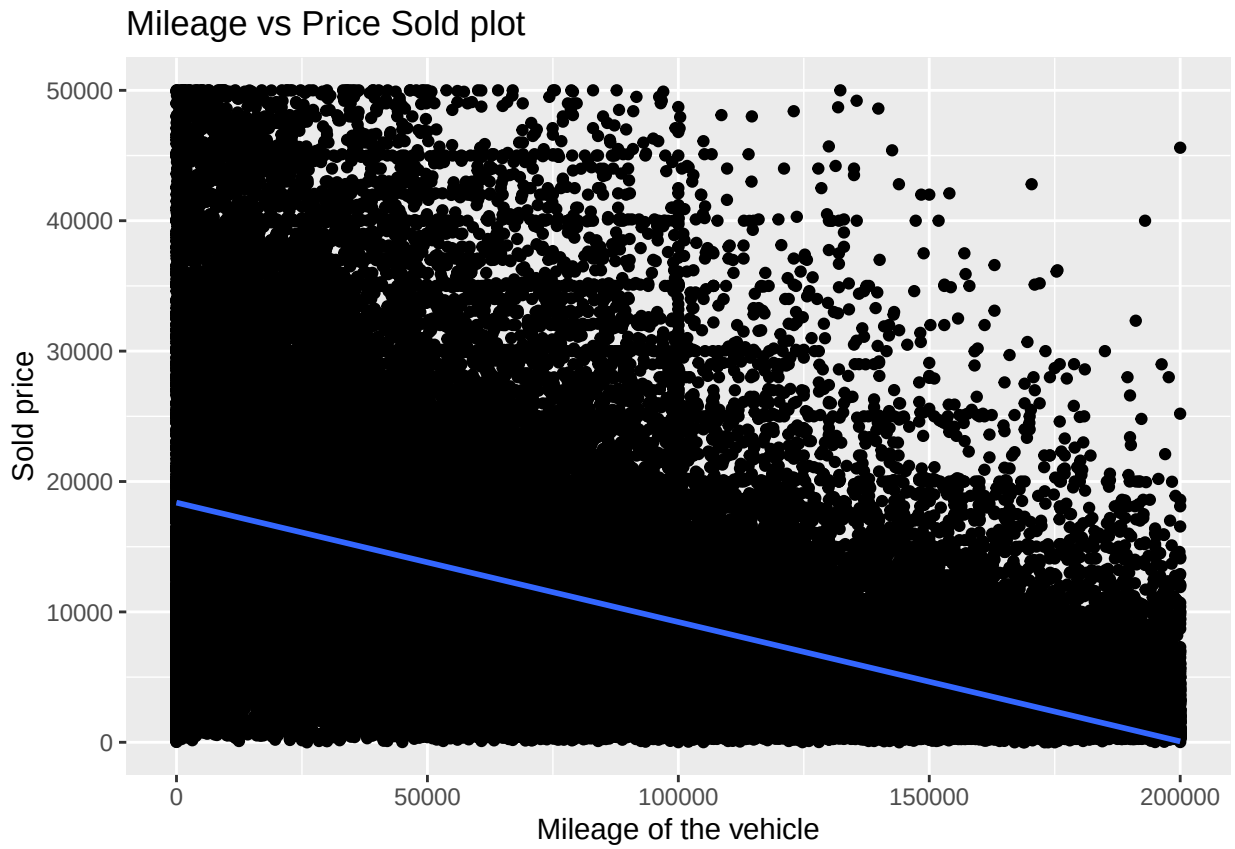
From the below scatter plot, it is clearly evident that as the mileage of the vehicle increases, the selling price comes down.

```
scatter <- ggplot(carsales_exclude, aes(Mileage,pricesold)) + geom_point() +
  labs(title = "Mileage vs Price Sold plot",x = "Mileage of the vehicle", y = "Sold price") + geom_smoo
scatter
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

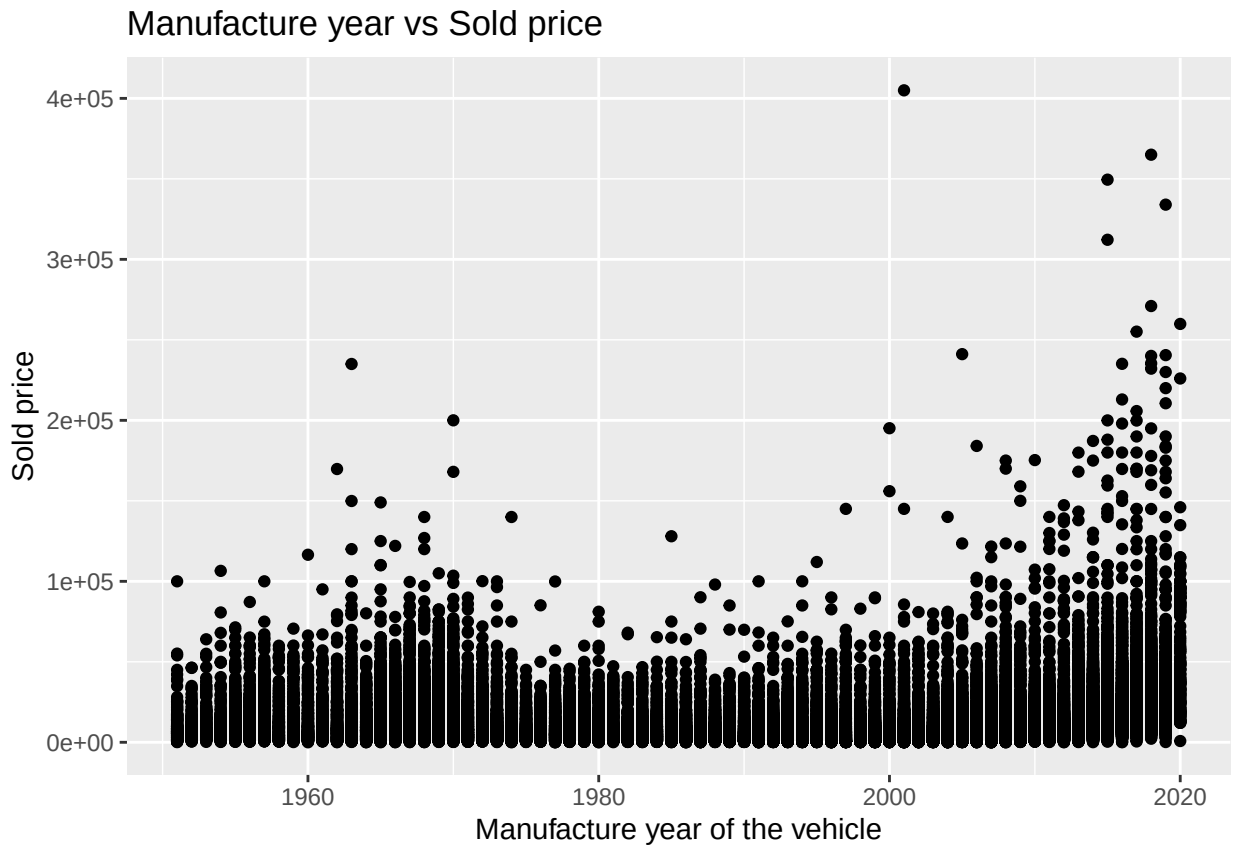
```
## Warning: Removed 8019 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 8019 rows containing missing values (`geom_point()`).
```



From the below scatter plot, it is showing clearly newer the manufactured year, more is the sold price.
Data which was hidden and that got uncovered is that the cars manufactured from 1965 to 1975 commanded higher price as well.
One reason that could be because, they are considered as vintage cars which are still in driving condition.

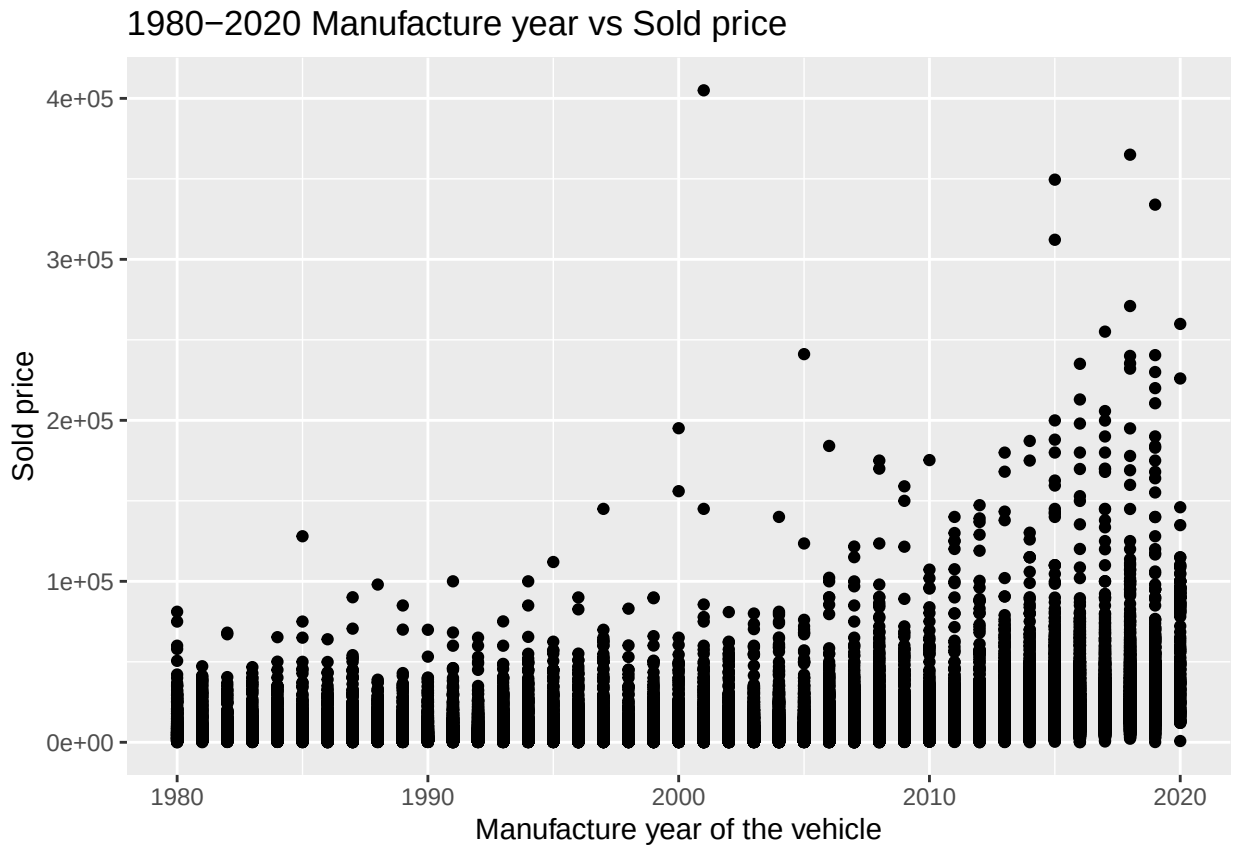
```
scatter_year <- ggplot(carsales_exclude_year, aes(Year,pricesold)) + geom_point() +
  labs(title = "Manufacture year vs Sold price",x = "Manufacture year of the vehicle", y = "Sold price")
scatter_year
```



Following scatter plot through the years 1980 to 2020, shows more clearly as newer the manufactured year, more is the sold price.

```
scatter_year_1980_2020 <- ggplot(carsales_exclude_year, aes(Year,pricesold)) + geom_point() +
  labs(title = "1980-2020 Manufacture year vs Sold price", x = "Manufacture year of the vehicle", y = "S
scatter_year_1980_2020
```

```
## Warning: Removed 15247 rows containing missing values (`geom_point()`).
```



Finding the most preferred make of the car by analysing how many times a make is sold and Cleaning the dataset to exclude the sold cars less than 50,
Converting the table to dataframe and renaming the columns,

```
popular_make <- table(carsales_exclude$Make)
popular_make <- popular_make[popular_make > 50]
names(popular_make)
```

```
##  [1] "Acura"                 "Alfa Romeo"
##  [3] "AMC"                   "Audi"
##  [5] "Austin Healey"         "Bentley"
##  [7] "BMW"                   "Buick"
##  [9] "Cadillac"              "Chevrolet"
## [11] "Chrysler"              "Datsun"
## [13] "Dodge"                 "Ferrari"
## [15] "Fiat"                  "Ford"
## [17] "GMC"                   "Honda"
## [19] "Hummer"                "Hyundai"
## [21] "Infiniti"              "International Harvester"
## [23] "Isuzu"                 "Jaguar"
## [25] "Jeep"                  "Kia"
## [27] "Land Rover"            "Lexus"
## [29] "Lincoln"               "Maserati"
## [31] "Mazda"                 "Mercedes-Benz"
## [33] "Mercury"               "MG"
## [35] "Mini"                  "Mitsubishi"
```

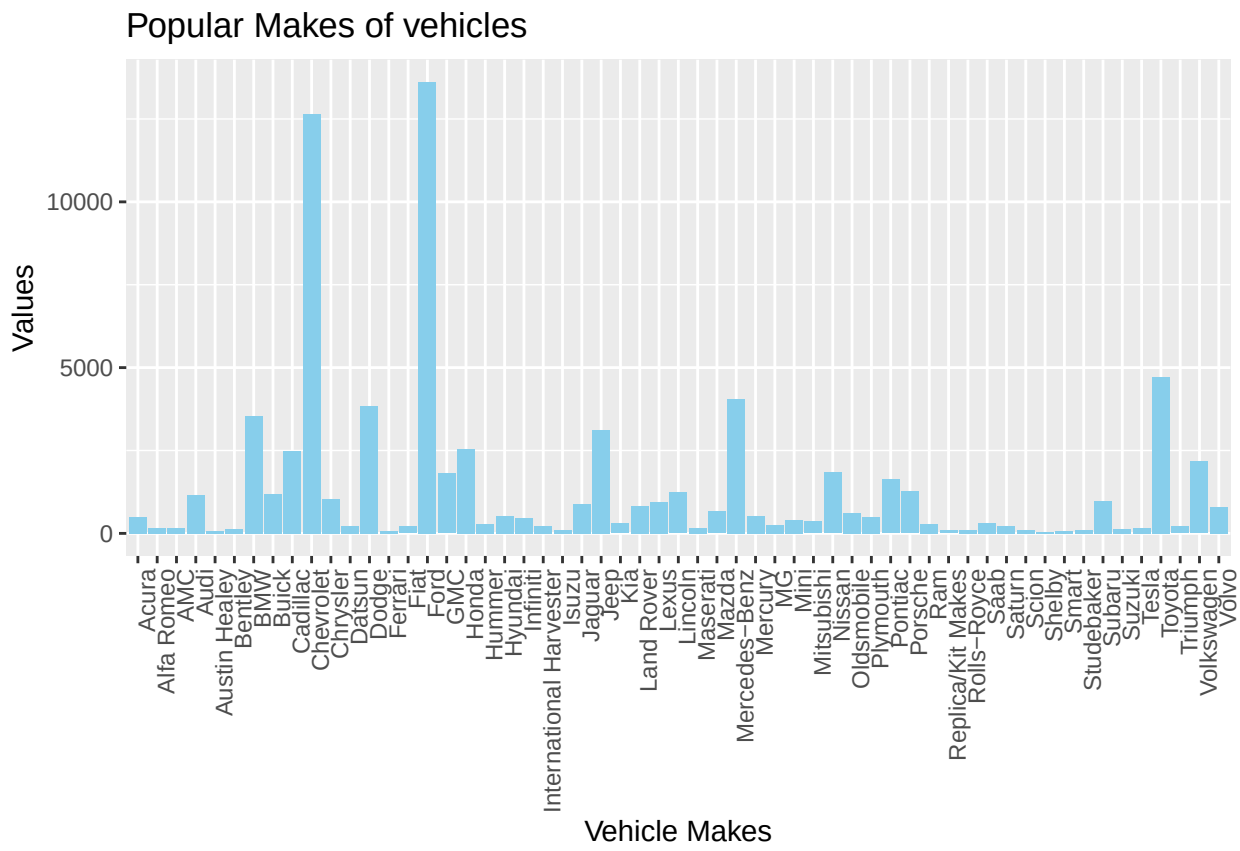
```
## [37] "Nissan"                  "Oldsmobile"
## [39] "Plymouth"                "Pontiac"
## [41] "Porsche"                 "Ram"
## [43] "Replica/Kit Makes"       "Rolls-Royce"
## [45] "Saab"                    "Saturn"
## [47] "Scion"                   "Shelby"
## [49] "Smart"                   "Studebaker"
## [51] "Subaru"                  "Suzuki"
## [53] "Tesla"                   "Toyota"
## [55] "Triumph"                 "Volkswagen"
## [57] "Volvo"
```

```
popular_make <- as.data.frame(popular_make)
popular_make <- popular_make %>% rename(Make=Var1, Count=Freq)
head(popular_make)
```

```
##            Make Count
## 1         Acura   498
## 2    Alfa Romeo   166
## 3           AMC   165
## 4          Audi  1164
## 5 Austin Healey    66
## 6       Bentley   143
```

Plotting the barplot for the different makes of the vehicles,

```
ggplot(popular_make, aes(Make,Count)) + geom_bar(stat = "identity", fill = "skyblue") +
  labs(title = "Popular Makes of vehicles", x = "Vehicle Makes", y = "Values") + theme(axis.text.x = el
```



From the above plot, it is a very clear indicator that Ford tops the list, closely followed by Chevrolet. Third most popular brand are Toyota/Mercedes Benz,Jeep and BMW.

We have discovered the car make and models which got sold for more than 100K,

```
carsales_exclude_morethan100k %>% select_('Make','Model') %>%  group_by(Model) %>% unique() %>% arrange
```

```
## Warning: `select_()` was deprecated in dplyr 0.7.0.
## i Please use `select()` instead.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
## # A tibble: 64 x 2
## # Groups:   Model [63]
##    Make         Model
##    <chr>        <chr>
##  1 Acura        NSX
##  2 Aston Martin DBS
##  3 Aston Martin Vanquish
##  4 Aston Martin Vantage
##  5 Audi         R8
##  6 BMW          Z8
##  7 BMW          8-Series
##  8 Bentley      Continental GT
##  9 Bentley      Bentayga
## 10 Bentley      Flying Spur
## # i 54 more rows
```

Using regex expression to find the Electric/Hybrid models from the column Engine. There are a total of 611,

```
library(stringr)
```

```
## Warning: package 'stringr' was built under R version 4.3.2
```

```
sum(str_detect(carsales_exclude$Engine, regex("Elec|hyb",ignore_case = TRUE)))
```

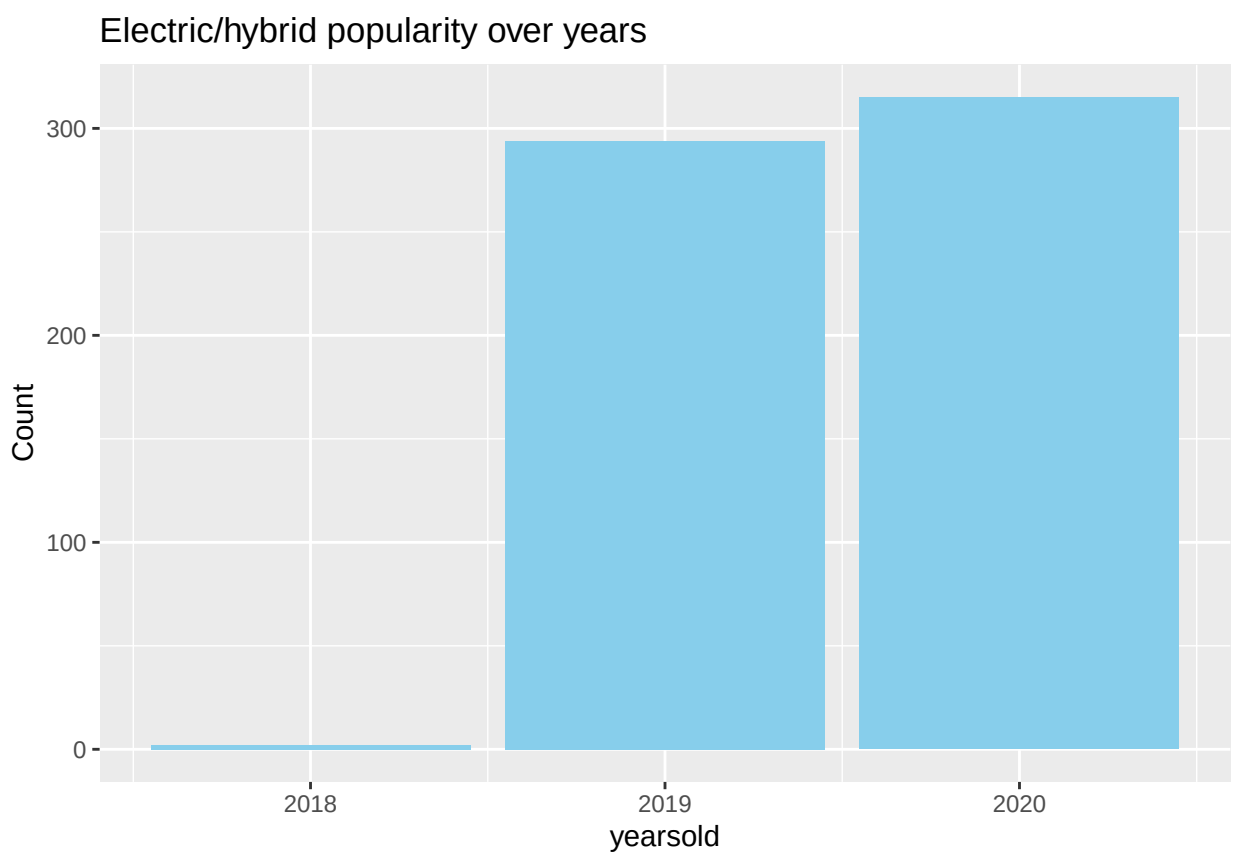
```
## [1] 611
```

```
elect_hybrid_vehicles <- carsales_exclude[str_detect(carsales_exclude$Engine, regex("Elec|hyb",ignore_ca

elect_hybrid_vehicles_count <- elect_hybrid_vehicles %>% group_by(yearsold) %>% summarise(Count = n())
elect_hybrid_vehicles_count
```

```
## # A tibble: 3 x 2
##   yearsold Count
##      <int> <int>
## 1     2018     2
## 2     2019   294
## 3     2020   315
```

```
ggplot(elect_hybrid_vehicles_count, aes(yearsold,Count)) + geom_bar(stat = "identity", fill = "skyblue"
  labs(title = "Electric/hybrid popularity over years", x = "yearsold", y = "Count")
```



From the above graph we are seeing a steady increase of electric/hybrid cars over the years.

Using two different datasets usedcar_lat_lon and usedcar_train, we are going to merge the latitude and longitude and do data analysis on the latitude and longitude and the sales.
To achieve this, I have joined the two datasets using merge and by specifying the primary key as region.
I have rounded off the latitude and longitude to 0 decimals for convenience by using the round function.
arranged the count by descending.

```
usedcarwithlatlong <- merge(usedcar_lat_lon, usedcar_train, by = "region")
usedcarwithlatlong_count <- usedcarwithlatlong %>% group_by(region,round(latitude,digits=0),round(longi
```

```
## `summarise()` has grouped output by 'region', 'round(latitude, digits = 0)'.
## You can override using the `.groups` argument.
```

```
usedcarwithlatlong_count %>% arrange(desc(Count))
```

```
## # A tibble: 368 x 4
## # Groups:   region, round(latitude, digits = 0) [368]
##    region              round(latitude, digits = 0~1 round(longitude, dig~2 Count
##    <chr>                                      <dbl>                  <dbl> <int>
##  1 central NJ                                    40                    -75   881
##  2 rhode island                                  42                    -72   585
##  3 rochester                                     43                    -78   430
##  4 albany                                        41                     20   409
##  5 washington, DC                                39                    -77   392
##  6 las vegas                                     36                   -115   345
##  7 tampa bay area                                28                    -83   329
##  8 los angeles                                   34                   -118   328
##  9 richmond                                      38                    -77   296
## 10 dallas / fort worth                           33                    -97   281
## # i 358 more rows
## # i abbreviated names: 1: `round(latitude, digits = 0)`,
## #   2: `round(longitude, digits = 0)`
```

From the above data, we are able to see that lat/lon of 40 and -75 lead the sales significantly by 881 total sales, closely followed by 585 for lat/lon of 42 and -72.

To answer the question of did consumers preferred to use smaller vehicles more, in order to have gas efficiency.
Smaller vehicles has been filtered by using regex Sedan. All the other vehicles, I have classified as more fuel guzzling vehicles.
It has been found out that only 20% of the total sold vehicles are fuel efficient vehicles.
This indicates that people are not into buying the vehicles which offer more gas efficiency.

```
Totalcarsold <- nrow(carsales_exclude)
Totalcarsold
```

```
## [1] 77902
```

```
Sedan_count <- count(carsales_exclude[str_detect(carsales_exclude$BodyType, regex("sed",ignore_case = TR
Sedan_count
```

```
##       n
## 1 14349
```

Summarizing the data to answer key questions

1. Mean of price sold is around $6500 and 75% of vehicles lying within the cost of $13,000. Also the box plot upper whisker indicates that the price ranges from $13000 to $32000.

2. From the scatter plot, it is clearly evident that as the mileage of the vehicle increases, the selling price comes down.

3. From scatter plot, it is showing clearly newer the manufactured year, more is the sold price.

4. Analysing the popular makes, it is a very clear indicator that Ford tops the list, closely followed by Chevrolet. Third most popular brand are Toyota/Mercedes Benz,Jeep and BMW.

5. We are seeing a steady increase of adoption of electric/hybrid cars over the years.

6. From the data, we are able to see that lat/lon of 40 and -75 lead the sales significantly by 881 total sales, closely followed by 585 for lat/lon of 42 and -72.

7. It has been found out that only 20% of the total vehicles sold are fuel efficient vehicles. This indicates that people are not into buying the vehicles which offer more gas efficiency.

Uncovering new data from the dataset, which was not evident

1. We have discovered the car make and models which got sold for more than 100K.

2. The vehicles manufactured from 1965 to 1975 also commanded higher price as well. One reason that could be because, they are considered as vintage cars which are still in driving condition.

Implications to the consumer

From the above summary, we are now clear on what determines the cost of the used car. Our analysis can be divided to two segments of users.

From buyers standpoint, if anyone wants to buy a used car on a tight budget, it is wise to keep the following in mind,

1. Go for a vehicle which is relatively older and has a little bit more miles on it.

2. Stay away from vintage cars, which are still on road from 1965 to 1975.

3. Avoid the luxury models which were identified above. For example Ferrari etc.

4. Aim to spend around $6500 or less, as that is the median of the sold price.

From the sellers standpoint,

1. If you have vehicles built from 1965 to 1975, they command a higher selling price.

2. If you are having vehicles from the popular makes identified from above, there would be a lot of interested people to buy in.

3. The more miles you put on your car and the more it ages, selling price is going to decrease.

4. If you are planning to sell hybrid/electric vehicles, more interested buyers would be there.

5. If you are in and around NJ area, you can be lucky, as more transactions happen in that area.

Questions for future steps and usage of any machine learning techniques

We can still work on further to find out what are the additional variables that might contribute to the price of the car. Few variables like mileage and the year manufactured has been pointed out in this document. But there are more to be discovered.

Model creation

A very basic model has been created with year and mileage as independent variables and the car price as dependent variable. These variables has been chosen depending upon the common understanding that these drives the prices. However, this model can still be improved by considering the other variables like location sold, model of the car, body type etc.

From the summary of the model, Rsquared is 0.007, which means these variables just explains 0.7% of the price. Adjusted Rsquared, the predictive power is almost same which means that if this is derived from population rather than sample there would be no variance in Rsquared values. As the Pr(>|t|) is above .05 we are not considering Mileage as a factor. We are taking Year as an important predictor as it is less than .05 and it has positive correlation with the price sold. Which means, that for every 1 year increase, we are seeing a $65 increase in the cost.

```
carsales_exclude_year <- carsales_exclude_year %>% filter(Mileage > 1000) %>% filter(pricesold > 1000)
cor(carsales_exclude_year[,c("Year","Mileage","pricesold")])
```

```
##                  Year      Mileage    pricesold
## Year       1.00000000 -0.048171875  0.084162532
## Mileage   -0.04817187  1.000000000 -0.002387487
## pricesold  0.08416253 -0.002387487  1.000000000
```

```
model <- lm(pricesold ~ Year + Mileage , data=carsales_exclude_year,header=TRUE)
```

```
## Warning: In lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) :
##  extra argument 'header' will be disregarded
```

```
summary(model)
```

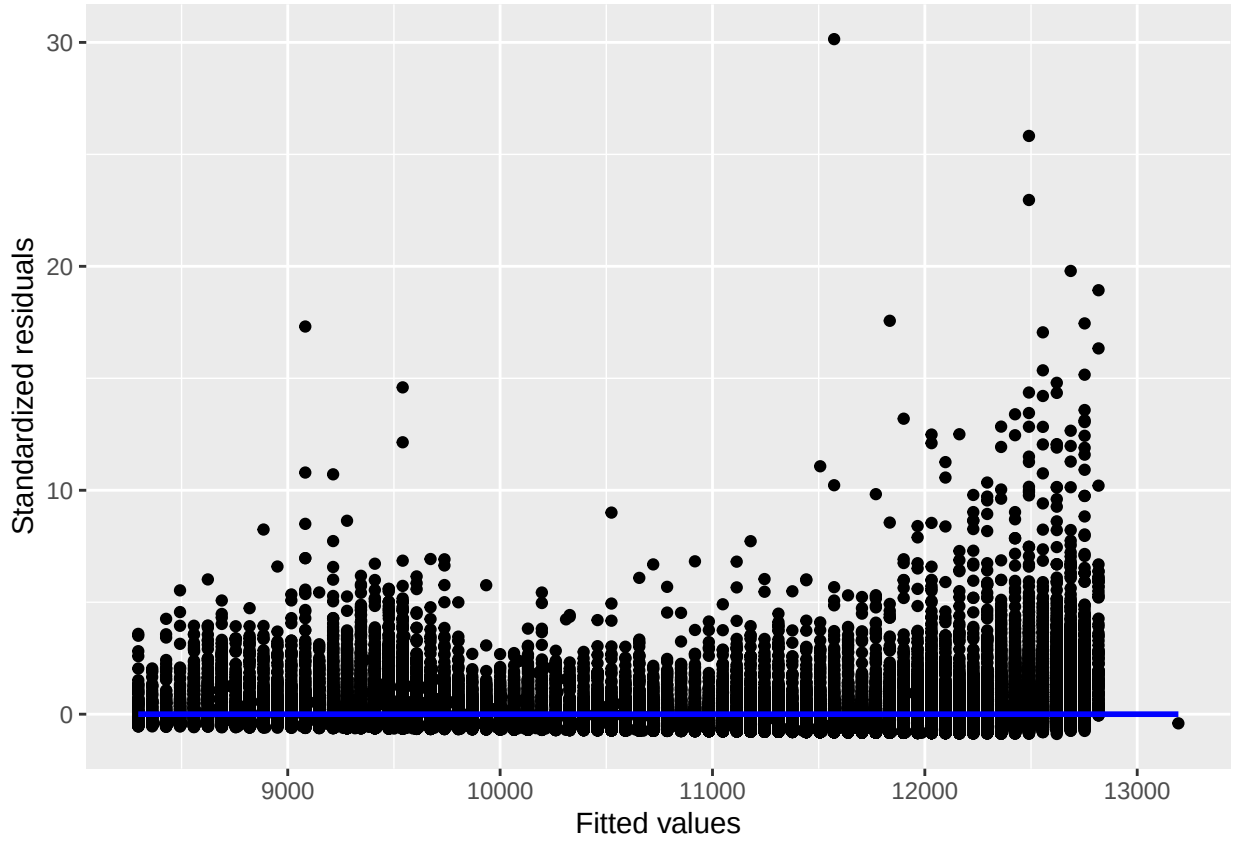
```
##
## Call:
## lm(formula = pricesold ~ Year + Mileage, data = carsales_exclude_year,
##     header = TRUE)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -11521  -7751  -4063   3196 393417
##
## Coefficients:
```

```
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.196e+05  5.877e+03 -20.344   <2e-16 ***
## Year         6.553e+01  2.942e+00  22.277   <2e-16 ***
## Mileage      8.347e-07  1.889e-06   0.442    0.659
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13050 on 69595 degrees of freedom
## Multiple R-squared:  0.007086,   Adjusted R-squared:  0.007058
## F-statistic: 248.3 on 2 and 69595 DF,  p-value: < 2.2e-16
```

```
carsales_exclude_year$residuals <- resid(model)
carsales_exclude_year$standardized.residuals <- rstandard(model)
carsales_exclude_year$studentized.residuals <- rstudent(model)
carsales_exclude_year$fitted <- model$fitted.values

library(ggplot2)
scatter <- ggplot(carsales_exclude_year,aes(fitted,standardized.residuals))
scatter + geom_point() + geom_smooth(method = "lm", color = "Blue") +
labs(x="Fitted values", y="Standardized residuals")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



Limitations

The model still needs a lot of improvement. We can get very good results, if all the high end cars are removed and some sort of standardization done. It can be further tweaked by considering the other variables like location sold, model of the car, body type etc.

However, there would be still a open question of when will the used card prices go down. This is quite unpredictable based on the existing conditions, as it depends on a lot of other external uncontrollable issues like,

1. New vehicle production volume.
2. Chipset and labour availability.
3. Bilateral relationships between countries.
4. How much a country is interested to ramp up the support required for the production.
5. How many manufacturers are willing to build the bare base models etc.

Concluding remarks

Using the available data, we have done a comprehensive analysis of the used car segment and have uncovered interesting information as mentioned in this paper.