Assignment 2

Mike Whitley 55503405

21/03/2021

Multiple Linear Regression Extra code has been put at bottom after question 5 for ease of reading where possible to whilst still keeping meaning

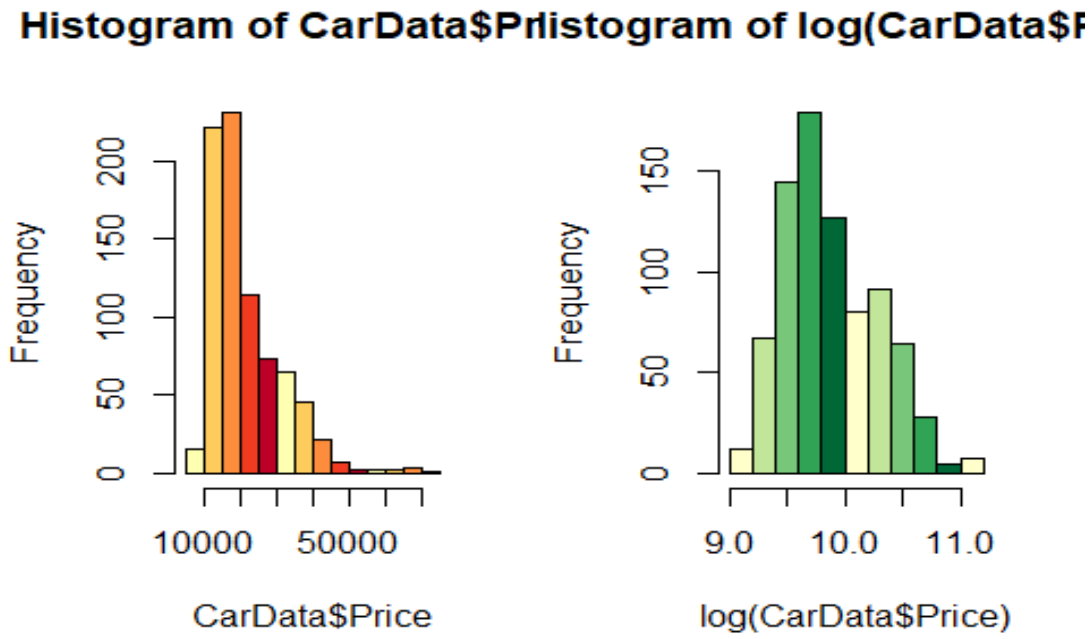
Full ordered explanation and full ordered R code file can be found at: https://github.com/Mike-Whitley/stat315.git under Assignment 2 markdown.Rmd

Use the dataset CarData.txt to investigate the relationship between the price of cars and a range of possible factors that might influence it. The factors recorded are as follows:

Price: suggested retail price of the used car still in excellent condition. Mileage: miles the car has been driven Make: manufacturer of the car Model: model of the car – ignore this for now Type: body type such as sedan, coupe, etc. Cylinder: number of cylinders in the engine Litre: a more specific measure of engine size Doors: number of doors Cruise: indicator variable representing whether the car has cruise control (1 = cruise) Sound: indicator variable representing whether the car has upgraded speakers (1 = upgraded) Leather: indicator variable representing whether the car has leather seats (1 = leather)

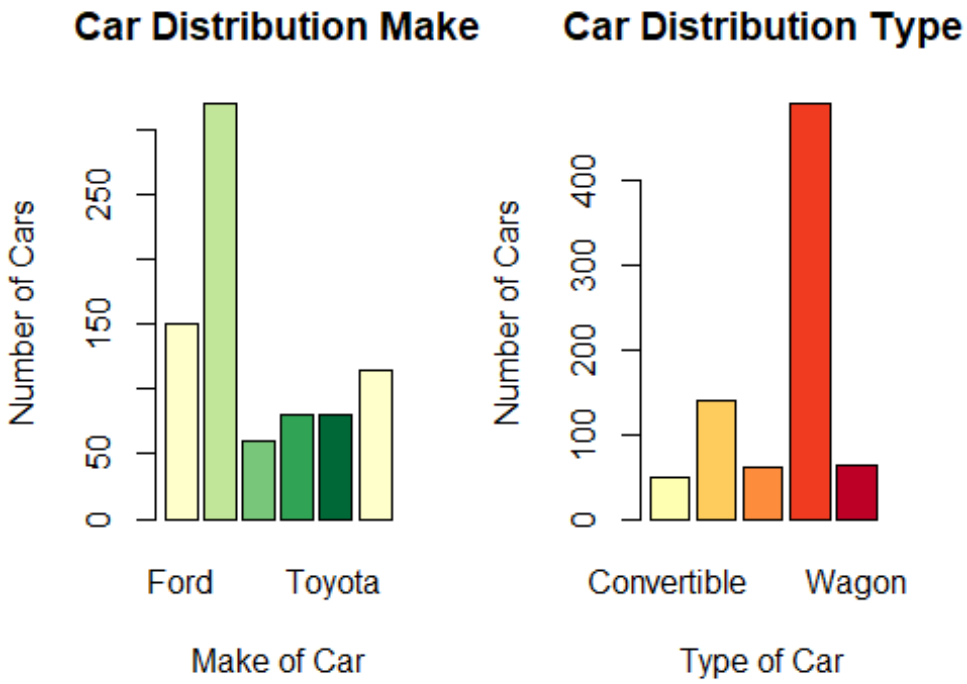
1. Start by exploring the data. Use R and create summary statistics and plots for each variable (ignore the variable for the model of car).

What I want to look for here is aspects such as possible outliers and trends I can immediately see see I think I want to have a quick look at all variables visually see if I can spot anything then do a bit more in-depth with anything I see as important and I want to check my response variable price is normally distributed

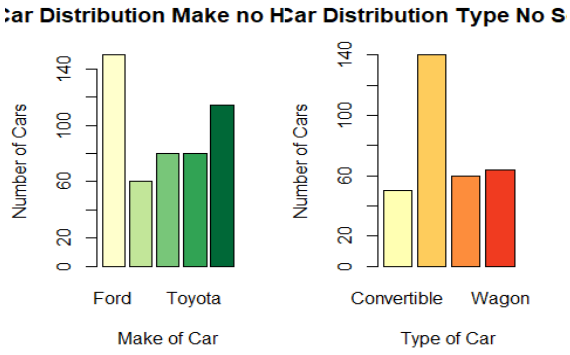


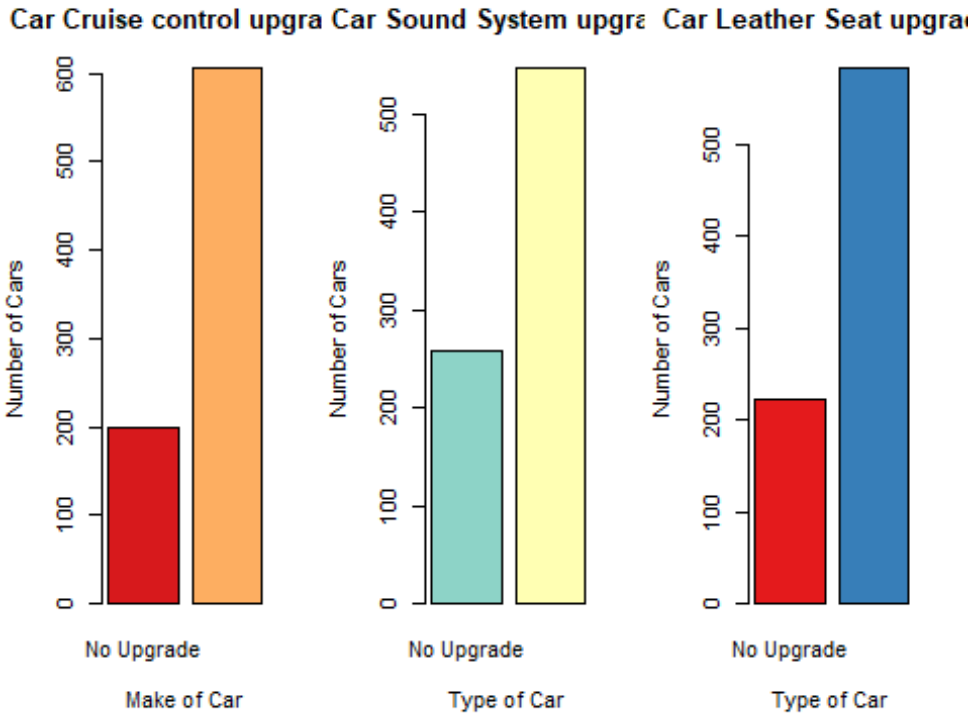
As I want to make sure my distribution of errors/residuals is normal the easiest way to ensure this is by having a normally distributed response variable therefor based on this I will change price to its logged equivalent which fits better

Make, Model, Type = categorical #ignore Model Cruise, Sound, Leather = Binary Price, Mileage, Cylinder, Litre, Doors = numerical



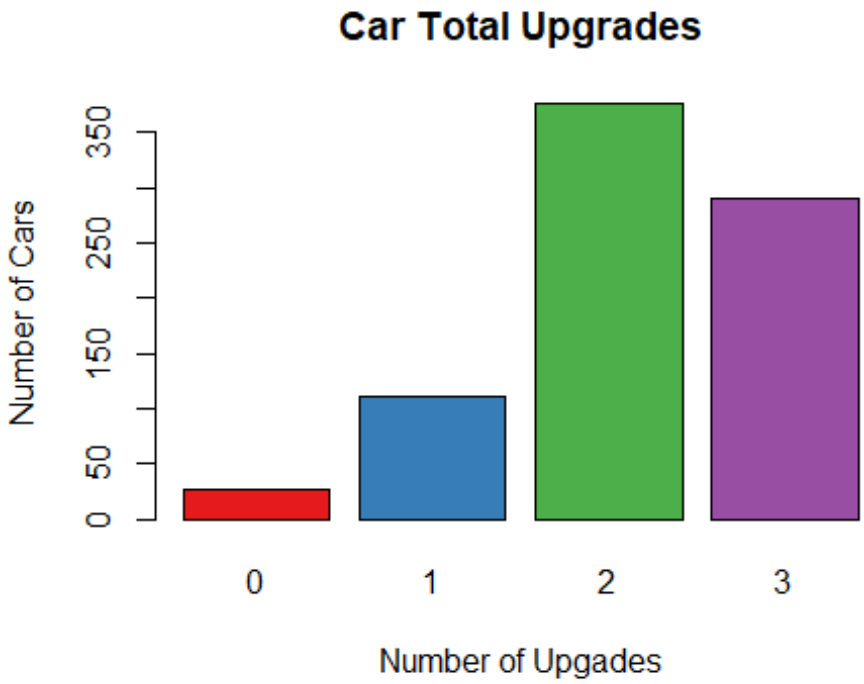
Can immediately see that there is more Hondas than almost all the other 5 combined and More sedans than the other 4 combined so the question is would this skew any other data we are trying to analyze later so might have to analyze both with honda/sedan and without for a fair picture especially if features differ? this could be relevant if Hondas are cheaper or more expensive could infer that Make has a big influence on price have to look at that when doing multiple variables





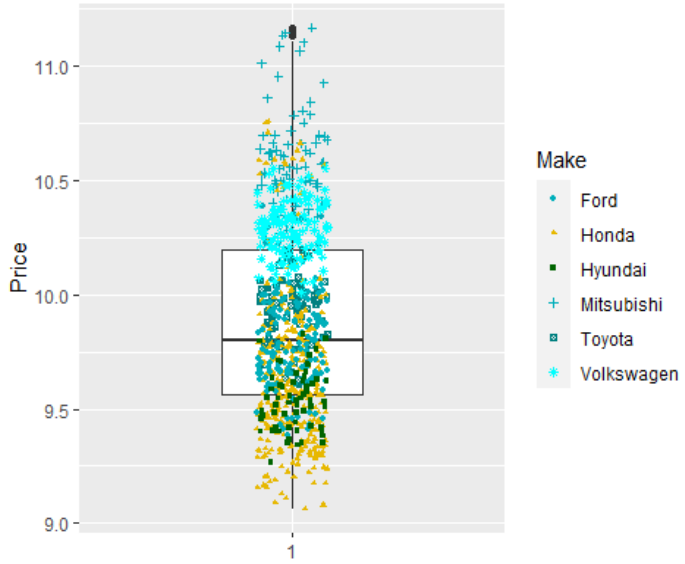
around 2/3 of cars have an upgrade in all types Unable to tell from this if the data we have got if it is a mix and match or all 3 have the upgrade or 2/3 etc wonder how good the spread if for cars with 0 - 3 upgrades

```
##      Price Mileage   Make Model  Type Cylinder Litre Doors Cruise Sound Le
ather
## 1 9.759277    8221 Toyota Camry Sedan        6   3.1     4      1     1
1
## 2 9.772355    9135 Toyota Camry Sedan        6   3.1     4      1     1
0
## 3 9.693929   13196 Toyota Camry Sedan        6   3.1     4      1     1
0
## 4 9.701182   16342 Toyota Camry Sedan        6   3.1     4      1     0
0
## 5 9.701321   19832 Toyota Camry Sedan        6   3.1     4      1     0
1
## 6 9.661992   22236 Toyota Camry Sedan        6   3.1     4      1     1
0
##   Total_upgrades
## 1              3
## 2              2
## 3              2
## 4              1
## 5              2
## 6              2
```



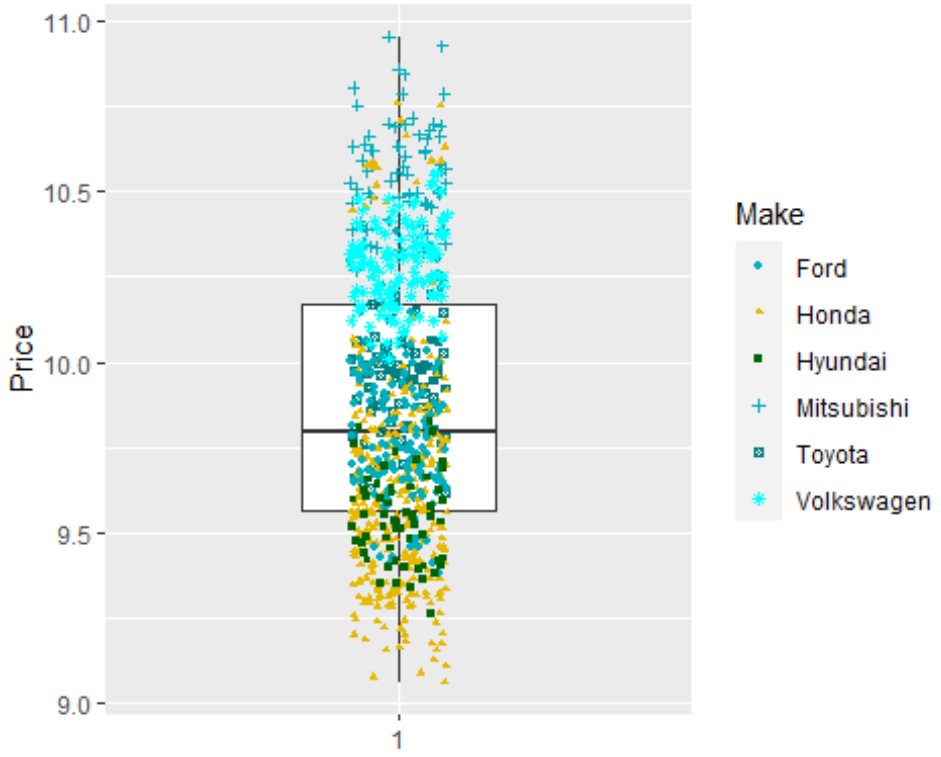
So most cars have at least 1 upgrade with the majority having 2 upgrade if we analyze between the upgrades can we find a trend in upgrades to price increase

Price, Mileage, Cylinder, Litre, Doors = numerical



We can see that the majority of cars fall under the 40k mark and the majority of Hondas are below 10.5 this could be indicative of why many Honda are in the data and few Mitsubishi due to Honda being cheaper There are quite a few values higher than the very next different car type I cant see anything indicative in the data to explain why Mitsubishi is getting near over 11.5 where the highest next Make is just under 10.8 I think dropping Mitsubishi values of 11 and higher will provide better

insight in to the data for reliability to price as after looking at the table data all cars over the 11 mark are Mitsubishi Lancer Convertibles 8 cylinders 4.6 liters with 3 upgrades its possible this is correct but after a bit of research unable to find any cars that are even a convertible Mitsubishi Lancer let alone 8 cylinders 4.6 liters] So I will remove them as I believe them to be in error outliers however can double check later to see if it changes anything significantly

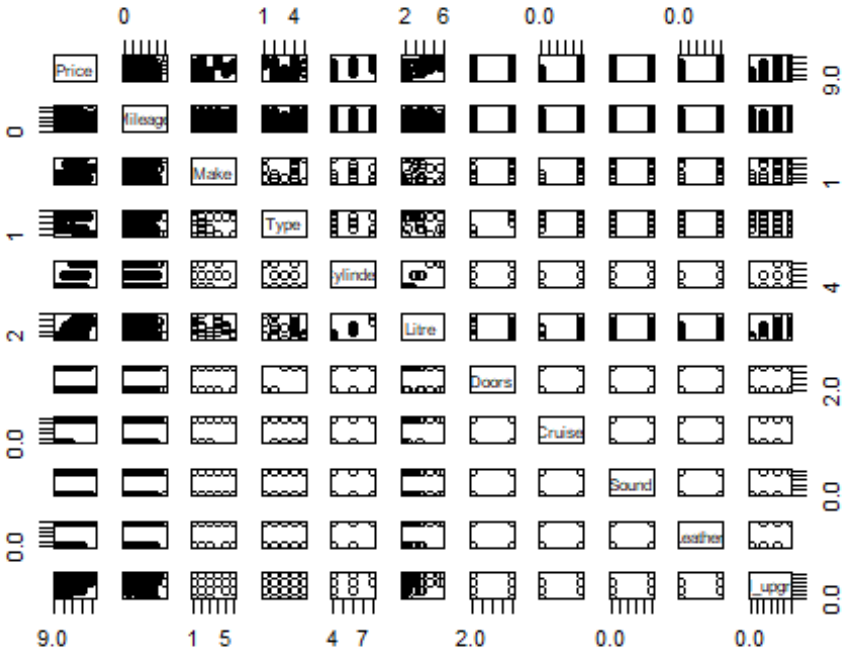


This looks like a much better spread of data.

2. Use suitable graphs and plots to explore the relationship between variables

Based on the individual variables explored some interesting aspects to look at would be total car upgrades variable that was made to see if this is a good measure does price increase for cars with upgrades?

Do a plot of all variables see if anything sticks out for a good relationship



Looking at that data there are not any nice relationships in relation to price the closet would be Litre and cylinders looking purely visually for positive relationships and mileage for negative relationship so might be indicative as Mileage decreases price increases and that fits with what I would assume would happen.

I want to look at relationships between the explanatory variables so my X variables to see if they are highly correlated will I see them in the final model for car price. so first I'll use a correlation matrix between all the variables

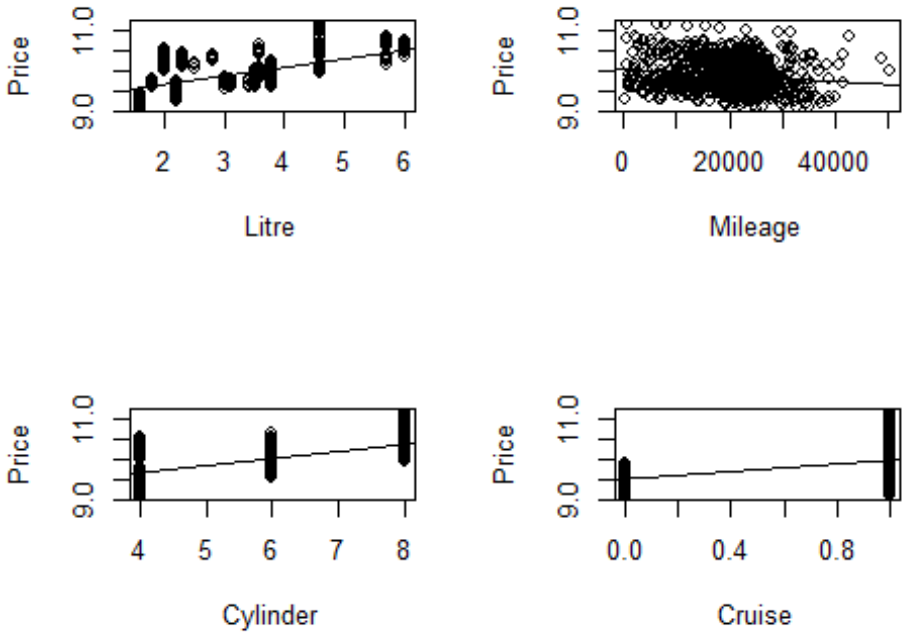
```
round(cor(CarData[, -(3:5)]), 2) #need to skip categorical data so Make model
type
```

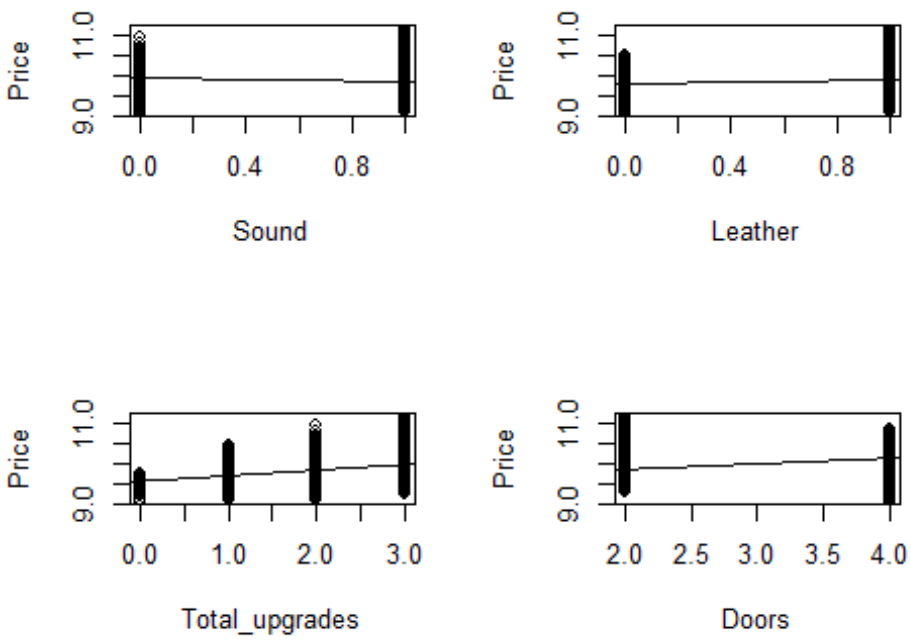
```
##                Price Mileage Cylinder Litre Doors Cruise Sound Leather
## Price           1.00   -0.15     0.58  0.59 -0.09   0.49 -0.14    0.13
## Mileage        -0.15    1.00    -0.03 -0.02 -0.02   0.03 -0.03    0.00
## Cylinder        0.58   -0.03     1.00  0.96  0.00   0.35 -0.09    0.08
## Litre           0.59   -0.02     0.96  1.00 -0.08   0.38 -0.07    0.09
## Doors          -0.09   -0.02     0.00 -0.08  1.00  -0.05 -0.06   -0.06
## Cruise          0.49    0.03     0.35  0.38 -0.05   1.00 -0.09   -0.07
## Sound          -0.14   -0.03    -0.09 -0.07 -0.06  -0.09  1.00    0.17
## Leather         0.13    0.00     0.08  0.09 -0.06  -0.07  0.17    1.00
## Total_upgrades  0.26    0.00     0.19  0.22 -0.10   0.46  0.64    0.63
##                Total_upgrades
## Price                    0.26
## Mileage                  0.00
## Cylinder                 0.19
## Litre                    0.22
```

```
## Doors                      -0.10
## Cruise                      0.46
## Sound                       0.64
## Leather                     0.63
## Total_upgrades              1.00
```

for the correlation between explanatory variable we can see that there is an overlap of 0.96 for cylinder and Litre so my end modal I would assume for one of these to drop out as they are so similar this indicated multicollinearity yet we can see almost no relationship between some of the other variables such as Mileage and Cylinder / Litre

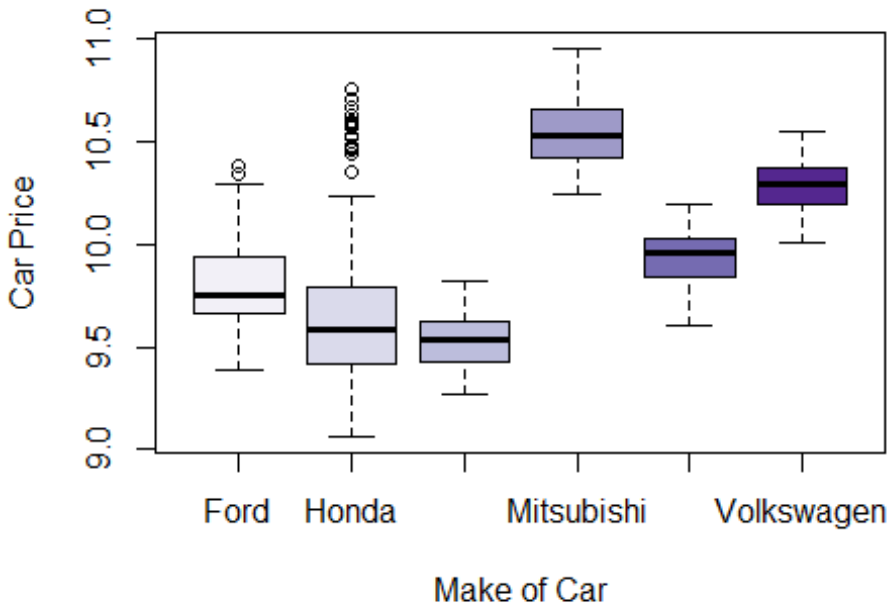
I also want to look a bit further between at Price and Mileage Price and Make Price and Type Price and Litre Price and Cylinder Price and Total upgrades and the 3 upgrades





Judging from these graphs there is a relationship between all these variables looked at and price to find out how strong looking at the summary apart from doors this is an odd variable we can see a good indication with Litre and cylinder

Judging from this data (refer to code below of all summary's) looking at the R-squared values there is no strong relationships evident currently for prediction of price meaning we will need multiple predictors for prediction of price.



We can see from this analysis of car Price to make of Car that Mitsubishis are the most expensive cars with hyundai being the cheapest we can determine from this that the make of the car influences the price to some degree if I was to go more in-depth with this I would split by each make of Car or possibly type of car

3. Next, fit a linear model starting with all the main effects. Reduce your model to the most parsimonious final model. Make sure you look at the residuals

first up start with my full model we want to drop variables to get lowest AIC I will use AIC value and anvoa f values to pick variables to drop. I will be using anova due to categorical variables I will drop doors first due to the fact that its full of NA values in summary as its an odd data of 2 or 4 doors it behaves as numerical but is closer to categorical

I have chosen to not use Total upgrades in this first model as it was added and adds little value based on all being NA

Start up by fitting the Maximal model

```
#this is backward selection
model1 = lm(Price ~ Mileage + Make + Type + Cylinder + Litre + Doors + Cruise
+ Sound + Leather, data = Car_Price_Update)
anova(model1) #using anova as want to be bit more aggressive with removing va
riables and because I have categorical variables I will use this to drop out
of the variables
```

```
## Analysis of Variance Table
##
## Response: Price
##            Df Sum Sq Mean Sq   F value  Pr(>F)
## Mileage     1  2.041  2.0413  253.1320 < 2e-16 ***
## Make        5 77.741 15.5483 1928.0790 < 2e-16 ***
## Type        4  8.038  2.0095  249.1844 < 2e-16 ***
## Cylinder    1 26.810 26.8100 3324.6048 < 2e-16 ***
## Litre       1  3.498  3.4981  433.7793 < 2e-16 ***
## Cruise      1  0.027  0.0275    3.4093 0.06521 .
## Sound       1  0.011  0.0111    1.3717 0.24188
## Leather     1  0.018  0.0178    2.2050 0.13796
## Residuals 781  6.298  0.0081
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(model1)
```

```
##
## Call:
## lm(formula = Price ~ Mileage + Make + Type + Cylinder + Litre +
##     Doors + Cruise + Sound + Leather, data = Car_Price_Update)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.33687 -0.05850  0.00501  0.06067  0.27978
##
## Coefficients: (1 not defined because of singularities)
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     9.540e+00  3.510e-02 271.818  < 2e-16 ***
## Mileage        -8.096e-06  3.912e-07 -20.692  < 2e-16 ***
## MakeHonda      -3.505e-02  1.023e-02  -3.427 0.000642 ***
## MakeHyundai    -8.371e-03  1.514e-02  -0.553 0.580566
## MakeMitsubishi  5.473e-01  1.593e-02  34.365  < 2e-16 ***
## MakeToyota      9.310e-02  1.314e-02   7.087 3.06e-12 ***
## MakeVolkswagen  6.420e-01  1.439e-02  44.609  < 2e-16 ***
## TypeCoupe      -3.005e-01  1.787e-02 -16.822  < 2e-16 ***
## TypeHatchback  -3.333e-01  2.071e-02 -16.093  < 2e-16 ***
## TypeSedan      -3.031e-01  1.594e-02 -19.010  < 2e-16 ***
## TypeWagon      -1.491e-01  1.894e-02  -7.871 1.17e-14 ***
## Cylinder       -3.237e-02  1.133e-02  -2.857 0.004394 **
## Litre           2.554e-01  1.271e-02  20.100  < 2e-16 ***
## Doors                  NA         NA      NA       NA
## Cruise          1.849e-02  9.159e-03   2.019 0.043862 *
## Sound           6.997e-03  7.308e-03   0.957 0.338639
## Leather         1.160e-02  7.814e-03   1.485 0.137964
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0898 on 781 degrees of freedom
```

```
## Multiple R-squared:  0.9494, Adjusted R-squared:  0.9484
## F-statistic:   977 on 15 and 781 DF,  p-value: < 2.2e-16
```

```
model2<-update(model1,~.-Doors) #removed doors variable
drop1(model2, test="F") #looking at what varibles I can remove if I take out cruise we will get a lower AIC overall and due to the F value being tiny
```

```
## Single term deletions
##
## Model:
## Price ~ Mileage + Make + Type + Cylinder + Litre + Cruise + Sound +
##     Leather
##          Df Sum of Sq    RSS     AIC  F value    Pr(>F)
## <none>                 6.298 -3826.0
## Mileage   1     3.453  9.751 -3479.6 428.1735 < 2.2e-16 ***
## Make      5    34.612 40.910 -2344.7 858.4148 < 2.2e-16 ***
## Type      4     3.469  9.767 -3484.3 107.5340 < 2.2e-16 ***
## Cylinder  1     0.066  6.364 -3819.7   8.1608  0.004394 **
## Litre     1     3.258  9.556 -3495.7 404.0186 < 2.2e-16 ***
## Cruise    1     0.033  6.331 -3823.8   4.0751  0.043862 *
## Sound     1     0.007  6.305 -3827.0   0.9167  0.338639
## Leather   1     0.018  6.316 -3825.7   2.2050  0.137964
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The next variables I will drop will be Cruise sound leather as these have small F values and I believe it to be a more aggressive approach based on F value expecially since doing this st ep by step

```
model3<-update(model2,~.-Cruise) #removed Cruise variable
model3<-update(model3,~.-Leather) #removed Leather variable
model3<-update(model3,~.-Sound) #removed Sound variable
anova(model3)
```

```
## Analysis of Variance Table
##
## Response: Price
##            Df Sum Sq Mean Sq F value    Pr(>F)
## Mileage     1  2.041  2.0413  251.85 < 2.2e-16 ***
## Make        5 77.741 15.5483 1918.33 < 2.2e-16 ***
## Type        4  8.038  2.0095  247.92 < 2.2e-16 ***
## Cylinder    1 26.810 26.8100 3307.79 < 2.2e-16 ***
## Litre       1  3.498  3.4981  431.59 < 2.2e-16 ***
## Residuals 784  6.354  0.0081
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Now I've dropped all the super low F values I will start using AIC to reduce further

```
drop1(model3) #looking at what varibles I can remove if I take out cruise we will get a lower AIC overall and due to the F value being tiny
```

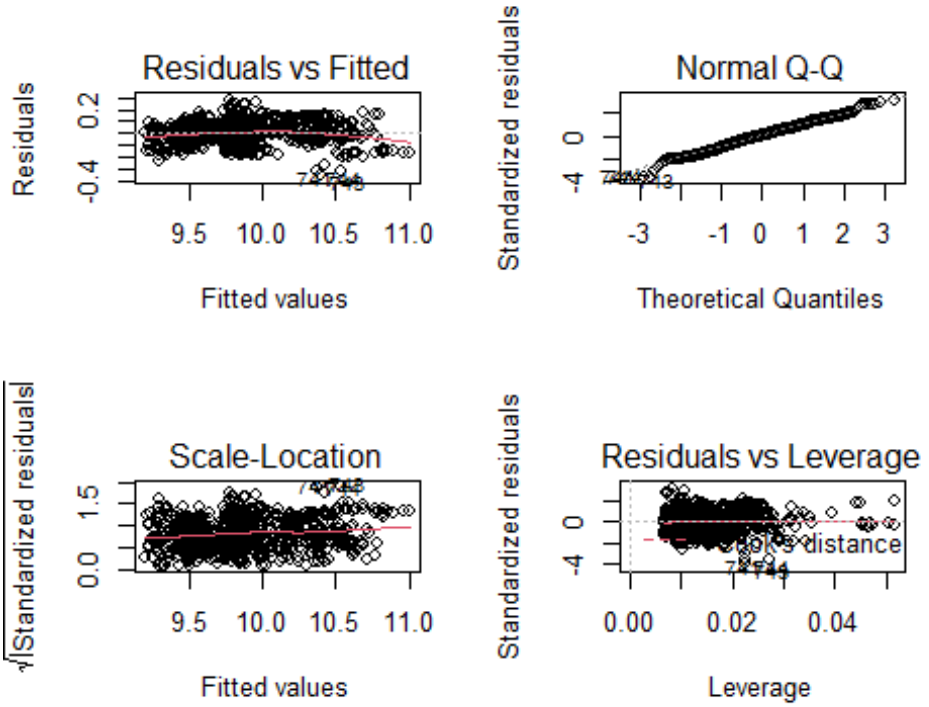
```
## Single term deletions
##
## Model:
## Price ~ Mileage + Make + Type + Cylinder + Litre
##          Df Sum of Sq    RSS     AIC
## <none>                  6.354 -3824.9
## Mileage   1     3.440  9.795 -3482.0
## Make      5    41.851 48.206 -2219.9
## Type      4     3.437  9.792 -3488.3
## Cylinder  1     0.077  6.431 -3817.3
## Litre     1     3.498  9.852 -3477.3
```

There are no more changes that can be made here as current AIC at -3824.9 as nothing I remove can make it lower I could have removed 1 to many AIC values I can run the auto step to double check later however I will remove cylinder manually as the multicollinearity identified above with Litre cylinder has lower correlation with price this will give me a simpler model overall in turn leaving me with AIC of -3817.3

this leaves me with mileage make type and Litre as my variabels

I want to check the assumptions i've made using the residuals

```
par(mfrow=c(2,2))
plot(model3)
```



Looking at the first plot there is curvature starting at around 10.5 and prior to 9.5 however as these are minor deviations based on all the models they are reasonable enough for standard assumptions of

linear regression due to a even enough spread of data points below and above the red line. However a quadratic model may be a better fit here as we can see curvature in the first plot

I can see from the Normal QQ plot a reasonably tight fit so he assumption of normality applies here

Cooks distance is not showing any outliers that impact negatively on my residuals in regards to leverage

4. Try some different model fitting methods such as starting with a minimal model and adding terms to it (forward selection), or using the step-both-ways option in R. Compare your final models with your backwards selection model. You can also try some different ways to make decisions, such as compare your final, best model from using AIC, BIC, or, if you have done STAT202, Mallows Cp.

```
#first try automatic backwards model
step(model1, direction = "backward")
```

```
## Start:  AIC=-3825.97
## Price ~ Mileage + Make + Type + Cylinder + Litre + Doors + Cruise +
##     Sound + Leather
##
##
## - Mileage   1     3.457  9.762 -3480.7
## - Make      5    34.778 41.083 -2343.3

##
## Call:
## lm(formula = Price ~ Mileage + Make + Type + Cylinder + Litre +
##     Cruise + Leather, data = Car_Price_Update)
##
## Coefficients:
##    (Intercept)         Mileage       MakeHonda     MakeHyundai  MakeMitsub
ishi
##      9.5458752      -0.0000081      -0.0343213      -0.0102908       0.546
9821
##     MakeToyota  MakeVolkswagen       TypeCoupe   TypeHatchback       TypeS
edan
##      0.0935206       0.6414880      -0.2994658      -0.3324698      -0.302
1937
##      TypeWagon        Cylinder           Litre          Cruise         Lea
ther
##     -0.1501306      -0.0333033       0.2560582       0.0183397       0.012
6242
```

so the variables removed are Doors and Cruise as compared to my step by step one the only difference is I also removed Leather and Cruise now to try it both ways final AIC of -3827.03 Now with manual removal of Cylinder my final AIC is -3820.2

```r
step(model1, direction = "both")
```

```
## Start:  AIC=-3825.97
## Price ~ Mileage + Make + Type + Cylinder + Litre + Doors + Cruise +
##     Sound + Leather
##
##
## Step:  AIC=-3825.97
## Price ~ Mileage + Make + Type + Cylinder + Litre + Cruise + Sound +
##     Leather
##
##            Df Sum of Sq    RSS     AIC
## - Sound     1     0.007  6.305 -3827.0
## <none>                   6.298 -3826.0
## - Leather   1     0.018  6.316 -3825.7
## - Cruise    1     0.033  6.331 -3823.8
## - Cylinder  1     0.066  6.364 -3819.7
## - Litre     1     3.258  9.556 -3495.7
## - Type      4     3.469  9.767 -3484.3
## - Mileage   1     3.453  9.751 -3479.6
# - Make      5    34.612 40.910 -2344.7
## Call:
## lm(formula = Price ~ Mileage + Make + Type + Cylinder + Litre +
##     Cruise + Leather, data = Car_Price_Update)
```

this made no difference to the backwards way so now I'll try with a a minimal model and adding terms to it (forward selection)

```r
#min model forward selection cylinder has been pre removed

minmod_with_totalupgrades = lm(Price ~ 1, data = Car_Price_Update)
step(minmod_with_totalupgrades, direction = "forward", # forwards
 scope = list(lower = ~ 1,
 upper = ~ Mileage + Make + Type +
 Litre + Doors + Cruise + Sound + Leather + Total_upgrades))
```

```
## Start:  AIC=-1477.78
## Price ~ 1
##
##                  Df Sum of Sq     RSS     AIC
## + Make            5    77.033  47.450 -2236.5
## + Litre           1    42.094  82.389 -1804.7
## + Cruise          1    31.025  93.458 -1704.2
## + Type            4    27.919  96.564 -1672.2
## + Total_upgrades  1     7.592 116.891 -1525.9
## + Sound           1     3.354 121.129 -1497.5
## + Mileage         1     2.041 122.442 -1489.0
## + Leather         1     1.757 122.726 -1487.1
## <none>                        124.483 -1477.8
## + Doors           1     0.275 124.208 -1477.5
##
```

```
##
## Step:  AIC=-3817.28
## Price ~ Make + Litre + Type + Mileage
##
##                   Df Sum of Sq    RSS     AIC
## + Total_upgrades  1  0.063129 6.3682 -3823.1
## + Leather         1  0.028489 6.4028 -3818.8
## + Cruise          1  0.021022 6.4103 -3817.9
## + Sound           1  0.016731 6.4146 -3817.4
## <none>                       6.4313 -3817.3
##
## Step:  AIC=-3823.15
## Price ~ Make + Litre + Type + Mileage + Total_upgrades
##
##           Df Sum of Sq    RSS     AIC
## <none>                  6.3682 -3823.1
## + Sound    1 0.0038707 6.3643 -3821.6
## + Cruise   1 0.0022724 6.3659 -3821.4
## + Leather  1 0.0001964 6.3680 -3821.2

##
## Call:
## lm(formula = Price ~ Make + Litre + Type + Mileage + Total_upgrades,
##     data = Car_Price_Update)
##
## Coefficients:
##    (Intercept)       MakeHonda     MakeHyundai  MakeMitsubishi      MakeTo
yota
##      9.467e+00      -2.948e-02      -4.266e-04       5.271e-01       9.590
e-02
## MakeVolkswagen           Litre       TypeCoupe   TypeHatchback       TypeS
edan
##      6.574e-01       2.218e-01      -2.943e-01      -3.458e-01      -3.028
e-01
##      TypeWagon         Mileage  Total_upgrades
##     -1.437e-01      -8.096e-06       1.253e-02
```

```r
#min model forward selection  with cylinder removed already

minmod = lm(Price ~ 1, data = Car_Price_Update)
step(minmod, direction = "forward", # forwards
 scope = list(lower = ~ 1,
 upper = ~ Mileage + Make + Type +
```

```
##
## Step:  AIC=-3818.82
## Price ~ Make + Litre + Type + Mileage + Leather
##
##          Df Sum of Sq    RSS     AIC
```

```
## + Cruise  1  0.027152 6.3757 -3820.2
## <none>                6.4028 -3818.8
## + Sound   1  0.010985 6.3918 -3818.2
##
## Step:  AIC=-3820.21
## Price ~ Make + Litre + Type + Mileage + Leather + Cruise
##
##         Df Sum of Sq    RSS     AIC
## <none>               6.3757 -3820.2
## + Sound  1  0.011764 6.3639 -3819.7

##
## Call:
## lm(formula = Price ~ Make + Litre + Type + Mileage + Leather +
##     Cruise, data = Car_Price_Update)
##
## Coefficients:
##    (Intercept)       MakeHonda     MakeHyundai  MakeMitsubishi      MakeTo
yota
##      9.473e+00      -2.771e-02      -2.404e-03       5.246e-01       9.626
e-02
## MakeVolkswagen           Litre       TypeCoupe   TypeHatchback       TypeS
edan
##      6.549e-01       2.205e-01      -2.927e-01      -3.444e-01      -3.015
e-01
##      TypeWagon         Mileage         Leather          Cruise
##     -1.450e-01      -8.108e-06       1.587e-02       1.678e-02
```

-3820.21 = forward step model without total_upgades -3823.15 = forwards step model with total_upgrades -3820.2- backward step model The forward step model has a better AIC value as I've added Total_upgrades to it :) making my best model

lm(formula = Price ~ Make + Litre + Type + Mileage + Cylinder + Total_upgrades, data = Car_Price_Update)

```
anova(model1, minmod_with_totalupgrades)
```

```
## Analysis of Variance Table
##
## Model 1: Price ~ Mileage + Make + Type + Cylinder + Litre + Doors + Cruise
+
##     Sound + Leather
## Model 2: Price ~ 1
##   Res.Df     RSS  Df Sum of Sq      F    Pr(>F)
## 1    781   6.298
## 2    796 124.483 -15   -118.19 977.04 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
best_model <- lm(formula = Price ~ Make + Litre + Type + Mileage +
    Total_upgrades, data = Car_Price_Update)
anova(best_model)
```

```
## Analysis of Variance Table
##
## Response: Price
##                 Df Sum Sq Mean Sq   F value    Pr(>F)
## Make             5 77.033  15.407 1896.7419 < 2.2e-16 ***
## Litre            1 34.421  34.421 4237.7049 < 2.2e-16 ***
## Type             4  3.152   0.788   97.0025 < 2.2e-16 ***
## Mileage          1  3.446   3.446  424.1907 < 2.2e-16 ***
## Total_upgrades   1  0.063   0.063    7.7719  0.005435 **
## Residuals      784  6.368   0.008
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(best_model)
```

```
##
## Call:
## lm(formula = Price ~ Make + Litre + Type + Mileage + Total_upgrades,
##     data = Car_Price_Update)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.33041 -0.05817  0.00098  0.05999  0.28571
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     9.467e+00  2.463e-02 384.375  < 2e-16 ***
## MakeHonda      -2.948e-02  9.946e-03  -2.964  0.00313 **
## MakeHyundai    -4.266e-04  1.490e-02  -0.029  0.97716
## MakeMitsubishi  5.271e-01  1.384e-02  38.083  < 2e-16 ***
## MakeToyota      9.590e-02  1.299e-02   7.382  4.0e-13 ***
## MakeVolkswagen  6.574e-01  1.323e-02  49.679  < 2e-16 ***
## Litre           2.218e-01  3.712e-03  59.741  < 2e-16 ***
## TypeCoupe      -2.943e-01  1.780e-02 -16.539  < 2e-16 ***
## TypeHatchback  -3.458e-01  2.027e-02 -17.064  < 2e-16 ***
## TypeSedan      -3.028e-01  1.600e-02 -18.926  < 2e-16 ***
## TypeWagon      -1.437e-01  1.886e-02  -7.620  7.3e-14 ***
## Mileage        -8.096e-06  3.926e-07 -20.625  < 2e-16 ***
## Total_upgrades  1.253e-02  4.494e-03   2.788  0.00543 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.09013 on 784 degrees of freedom
## Multiple R-squared:  0.9488, Adjusted R-squared:  0.9481
## F-statistic:  1212 on 12 and 784 DF,  p-value: < 2.2e-16
```

```
#dopping cylinder and total_upgrades to see any change based on F value being
low for them
```

5. Discuss what your final, best model means – what effects car price and in what way?

My final best model for predicting the price of cars is: Price = Make + Litre + Type + Mileage + Total_upgrades with the associated coefficient being the Estimates shown within the summary above. The largest predictors that effect care price are Litre and make of the car followed by Mileage where as type and Total_upgrades added extra strength to the model but are not as important. Overall my best model is a balance between having a smallish number of predictors whilst adding enough variables to make good predictions. If I was to build on this Make of cars would be split due to this being an influence identified earlier that this has a big impact on the price. The model has a 0.9488 R squared value and a low P value and a Residual standard error pf 0.09013 indicating to me that overall this is a good model for predicting price

Full ordered explanation and full R file can be found at: https://github.com/Mike-Whitley/stat315.git under Assignment 2 markdown.Rmd

Extra code that's been used but easier to read if removed:

```
CarData <- read.delim("CarData.txt") #read in the CarData.txt textfile
head(CarData)

#there are lots of NA columns that hold no relevance so will drop all na colu
mns
CarData <- CarData[,colSums(is.na(CarData))<nrow(CarData)]
head(CarData)

par(mfrow=c(1,2))
library(RColorBrewer)
coul2 <- brewer.pal(5, "YlOrRd")
coul <- brewer.pal(5, "YlGn")

hist(CarData$Price,col = coul2)
hist(log(CarData$Price), col=coul)

CarData[,c(1)] <- log(CarData[,c(1)])

#first up looking at categorical data
par(mfrow=c(1,2))
counts <- table(CarData$Make)
barplot(counts, main="Car Distribution Make",
```

```
   xlab="Make of Car",ylab="Number of Cars",col=coul)

counts <- table(CarData$Type)
barplot(counts, main="Car Distribution Type",
   xlab="Type of Car",  ylab="Number of Cars", col=coul2)

library(dplyr)

par(mfrow=c(1,2))
Data_Without_Honda <- subset(CarData, Make != "Honda") #give me a subset of d
ata without Honda for later use if needed
Data_Without_Sedan <- subset(CarData, Type != "Sedan") #same for data with ou
t sedans

coul2 <- brewer.pal(5, "YlOrRd")
coul <- brewer.pal(5, "YlGn")

counts <- table(Data_Without_Honda$Make)
barplot(counts, main="Car Distribution Make no Honda",
   xlab="Make of Car",ylab="Number of Cars",col=coul)

counts <- table(Data_Without_Sedan$Type)
barplot(counts, main="Car Distribution Type No Sedan",
   xlab="Type of Car",  ylab="Number of Cars", col=coul2)

#lots more evenish data types might be useful later unsure currently
```

do most cars have cruise control, upgraded speakers and leather seats??

```
library(RColorBrewer)
coul3 <- brewer.pal(5, "Set1")
coul2 <- brewer.pal(5, "Set3")
coul <- brewer.pal(5, "Spectral")

par(mfrow=c(1,3))

counts <- table(CarData$Cruise)
barplot(counts, main="Car Cruise control upgrade",
   xlab="Make of Car",ylab="Number of Cars",col=coul,names=c("No Upgrade","Up
graded" ))

counts <- table(CarData$Sound)
barplot(counts, main="Car Sound System upgrade",
   xlab="Type of Car",  ylab="Number of Cars", col=coul2,names=c("No Upgrade"
,"Upgraded" ))


counts <- table(CarData$Leather)
barplot(counts, main="Car Leather Seat upgrade",
```

```
   xlab="Type of Car",  ylab="Number of Cars", col=coul3,names=c("No Upgrade"
,"Upgraded" ))

colnms=c("Cruise", "Sound", "Leather")
CarData$Total_upgrades<-rowSums(CarData[,colnms])
head(CarData)

counts <- table(CarData$Total_upgrades)
barplot(counts, main="Car Total Upgrades",
   xlab="Number of Upgades",  ylab="Number of Cars", col=coul3,names=c("0","1
","2","3" ))

attach(CarData)

library(ggplot2)

ggplot(CarData, aes(x = factor(1), y = Price)) +
  geom_boxplot(width = 0.4, fill = "white") +
  geom_jitter(aes(color = Make, shape = Make),
              width = 0.1, size = 1) +
  scale_color_manual(values = c("#00AFBB", "#E7B800","#006400","#00AFBB","#00
8080","#00FFFF","#330000")) +
  labs(x = NULL)   # Remove x axis label

attach(CarData)

library(ggplot2)

ggplot(CarData, aes(x = factor(1), y = Price)) +
  geom_boxplot(width = 0.4, fill = "white") +
  geom_jitter(aes(color = Make, shape = Make),
              width = 0.1, size = 1) +
  scale_color_manual(values = c("#00AFBB", "#E7B800","#006400","#00AFBB","#00
8080","#00FFFF","#330000")) +
  labs(x = NULL)   # Remove x axis label

Car_Price_Update <- subset(CarData, Price <= 11)
head(Car_Price_Update)

#start by plotting all the data we don't want model of car variable as told t
o ignore it as column 4 is model we drop it
plot(CarData[,-4]) #this plots each variable against each the variable

par(mfrow=c(2,2))
plot(Litre, Price) #plot(x, y, …)
LitrePrice <- lm(Price ~ Litre) #put it in own variable for Lm to use later
abline(LitrePrice) #abline(v = y)

plot(Mileage, Price)
MileagePrice <- lm(Price ~ Mileage)
abline(MileagePrice)
```

```r
plot(Cylinder, Price)
CylinderPrice <- lm(Price ~ Cylinder)
abline(CylinderPrice)

plot(Cruise, Price)
CruisePrice <- lm(Price ~ Cruise)
abline(CruisePrice)

plot(Sound, Price)
SoundPrice <- lm(Price ~ Sound)
abline(SoundPrice)

plot(Leather, Price)
LeatherPrice <- lm(Price ~ Leather)
abline(LeatherPrice)

plot(Total_upgrades, Price)
Total_upgradesPrice <- lm(Price ~ Total_upgrades)
abline(Total_upgradesPrice)

plot(Doors, Price)

DoorsPrice <- lm(Price ~ Total_upgrades)
abline(DoorsPrice)

summary(LitrePrice)
```

```
##
## Call:
## lm(formula = Price ~ Litre)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.5002 -0.2381 -0.1210  0.1382  0.9457
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  9.21384    0.03417  269.65   <2e-16 ***
## Litre        0.21901    0.01057   20.72   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3312 on 802 degrees of freedom
## Multiple R-squared:  0.3486, Adjusted R-squared:  0.3478
## F-statistic: 429.2 on 1 and 802 DF,  p-value: < 2.2e-16
```

```r
summary(MileagePrice)
```

```
##
## Call:
## lm(formula = Price ~ Mileage)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.77514 -0.32412 -0.09456  0.30831  1.17632
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.003e+01  3.749e-02 267.406  < 2e-16 ***
## Mileage     -7.406e-06  1.747e-06  -4.239 2.51e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4058 on 802 degrees of freedom
## Multiple R-squared:  0.02191,    Adjusted R-squared:  0.02069
## F-statistic: 17.97 on 1 and 802 DF,  p-value: 2.511e-05
```

```
summary(CylinderPrice)
```

```
##
## Call:
## lm(formula = Price ~ Cylinder)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.59630 -0.24149 -0.09723  0.14384  0.89351
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 8.970750   0.046187  194.23   <2e-16 ***
## Cylinder    0.172397   0.008478   20.34   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3333 on 802 degrees of freedom
## Multiple R-squared:  0.3402, Adjusted R-squared:  0.3394
## F-statistic: 413.5 on 1 and 802 DF,  p-value: < 2.2e-16
```

```
summary(CruisePrice)
```

```
##
## Call:
## lm(formula = Price ~ Cruise)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.9045 -0.2380 -0.0127  0.2292  1.1719
##
## Coefficients:
```

```
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  9.52629    0.02530  376.55   <2e-16 ***
## Cruise       0.46879    0.02916   16.07   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3569 on 802 degrees of freedom
## Multiple R-squared:  0.2437, Adjusted R-squared:  0.2427
## F-statistic: 258.4 on 1 and 802 DF,  p-value: < 2.2e-16
```

```
summary(SoundPrice)
```

```
##
## Call:
## lm(formula = Price ~ Sound)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.89804 -0.30293 -0.07689  0.29777  1.32716
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   9.9621     0.0253 393.766  < 2e-16 ***
## Sound        -0.1222     0.0307  -3.982 7.45e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4064 on 802 degrees of freedom
## Multiple R-squared:  0.01939,    Adjusted R-squared:  0.01817
## F-statistic: 15.86 on 1 and 802 DF,  p-value: 7.452e-05
```

```
summary(LeatherPrice)
```

```
##
## Call:
## lm(formula = Price ~ Leather)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.80235 -0.32575 -0.06218  0.31810  1.25501
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  9.79274    0.02731 358.602  < 2e-16 ***
## Leather      0.11924    0.03210   3.715 0.000217 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4069 on 802 degrees of freedom
## Multiple R-squared:  0.01692,    Adjusted R-squared:  0.01569
## F-statistic:  13.8 on 1 and 802 DF,  p-value: 0.0002173
```

```r
summary(Total_upgradesPrice)

##
## Call:
## lm(formula = Price ~ Total_upgrades)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.7669 -0.2977 -0.0397  0.2885  1.1707
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)     9.57992    0.04101 233.628  < 2e-16 ***
## Total_upgrades  0.13878    0.01789   7.758 2.62e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3958 on 802 degrees of freedom
## Multiple R-squared:  0.06981,    Adjusted R-squared:  0.06865
## F-statistic: 60.19 on 1 and 802 DF,  p-value: 2.615e-14

coul <- brewer.pal(6, "Purples")
boxplot(Car_Price_Update$Price ~ Car_Price_Update$Make, col=coul, ylab = 'Car
Price',xlab = 'Make of Car')
```