Exam Template: Statistical Inference

21044874

Jan 2022: Sep21 run

# Instructions to students

You should only use the file Exam\_template.Rmd provided on blackboard and you should load this file from your scripts folder / directory.

Save this template as your studentID.Rmd; you will upload this file as your submission. Change the information on line 3 of this file – changing the author information to your **student ID**. Do not change the authorship to your name.

Ensure that you save your data into your data folder (as discussed in class). You may use the files mypackages.R and helperFunctions.R from blackboard. If you use these files, do not alter them. If you wish to create additional files for custom functions that you have prepared in advance, make sure that you upload these in addition to your .Rmd file and your compiled output file.

Your should knit this file to a document **Word** format.

Any changes that you make to the data (e.g. variable name changes) should be made entirely within R.

The subsubsections labelled **Answer:** indicate where you should put in your written Answers. The template also provides blank code chunks for you to complete your Answers; you may choose to add additional chunks if required.

# load dataset  
# using read.csv we load data  
Jan\_data <- read.csv("../data/Jan\_2022\_Exam\_Data.csv")  
head(Jan\_data)

brand model year price transmission mileage fuelType tax mpg engineSize  
1 Audi A1 2017 12500 Manual 15735 Petrol 150 55.4 1.4  
2 Audi A6 2016 16500 Automatic 36203 Diesel 20 64.2 2.0  
3 Audi A1 2016 11000 Manual 29946 Petrol 30 55.4 1.4  
4 Audi A4 2017 16800 Automatic 25952 Diesel 145 67.3 2.0  
5 Audi A3 2019 17300 Manual 1998 Petrol 145 49.6 1.0  
6 Audi A1 2016 13900 Automatic 32260 Petrol 30 58.9 1.4

str(Jan\_data)

'data.frame': 52533 obs. of 10 variables:  
 $ brand : chr "Audi" "Audi" "Audi" "Audi" ...  
 $ model : chr " A1" " A6" " A1" " A4" ...  
 $ year : int 2017 2016 2016 2017 2019 2016 2016 2016 2015 2016 ...  
 $ price : int 12500 16500 11000 16800 17300 13900 13250 11750 10200 12000 ...  
 $ transmission: chr "Manual" "Automatic" "Manual" "Automatic" ...  
 $ mileage : int 15735 36203 29946 25952 1998 32260 76788 75185 46112 22451 ...  
 $ fuelType : chr "Petrol" "Diesel" "Petrol" "Diesel" ...  
 $ tax : int 150 20 30 145 145 30 30 20 20 30 ...  
 $ mpg : num 55.4 64.2 55.4 67.3 49.6 58.9 61.4 70.6 60.1 55.4 ...  
 $ engineSize : num 1.4 2 1.4 2 1 1.4 2 2 1.4 1.4 ...

# Data description

This dataset is part of a larger dataset that has been collected to help to estimate the price of used cars.

It contains the following variables:

* brand (manufacturer)
* model (of car)
* year (of registration of the car)
* price (in GB pounds)
* transmission (type of gearbox)
* mileage (total distance covered by the car)
* fuelType (type of fuel used by the car)
* tax (annual cost of vehicle tax)
* mpg (miles per gallon - a measure of fuel efficiency)
* engineSize (size of the engine in litres)

# Question 1: Data Preparation (11 marks)

You are interested in modelling the price of vehicles that have all of the following properties:

* mileage less than 60000
* Manual transmission
* Petrol engine (fuelType)
* Costing less than £200 in annual Vehicle Tax.

Once you have selected the rows of data with these properties, then you must *use your studentID* to select a random sample of 2000 rows of the data to perform the rest of your analysis with.

You should remove any redundant variables (where only one value remains in that variable).

This subset of the data is what you should use for the rest of this assessment.

# we filter the data by specifying our required argument  
  
filtered\_data <- Jan\_data %>%  
 filter(mileage<60000)%>%  
 filter(transmission == "Manual")%>%  
 filter(fuelType == "Petrol")%>%  
 filter(tax<200)  
   
head(filtered\_data)

brand model year price transmission mileage fuelType tax mpg engineSize  
1 Audi A1 2017 12500 Manual 15735 Petrol 150 55.4 1.4  
2 Audi A1 2016 11000 Manual 29946 Petrol 30 55.4 1.4  
3 Audi A3 2019 17300 Manual 1998 Petrol 145 49.6 1.0  
4 Audi A3 2015 10200 Manual 46112 Petrol 20 60.1 1.4  
5 Audi A1 2016 12000 Manual 22451 Petrol 30 55.4 1.4  
6 Audi A3 2017 16100 Manual 28955 Petrol 145 58.9 1.4

## Set seed with my studentID

#using student id as set.seed, we select 2000 random samples from our df  
set.seed(21044874)  
rand\_filtered\_data <- filtered\_data[sample(nrow(filtered\_data), 2000, replace = FALSE, prob = NULL),]  
   
head(rand\_filtered\_data)

brand model year price transmission mileage fuelType tax mpg  
13434 Mercedes CL Class 2015 16600 Manual 34336 Petrol 145 48.7  
897 Audi A1 2020 20000 Manual 5000 Petrol 145 49.6  
4217 Ford Focus 2018 14998 Manual 10306 Petrol 145 58.9  
5260 Ford Fiesta 2017 8998 Manual 10602 Petrol 145 54.3  
5408 Ford Fiesta 2018 11000 Manual 33763 Petrol 145 56.5  
5359 Ford Fiesta 2014 7090 Manual 28877 Petrol 0 65.7  
 engineSize  
13434 1.6  
897 1.0  
4217 1.0  
5260 1.2  
5408 1.0  
5359 1.0

#first we use the duplicated() function from the dyplr library to check if we have any redundant record  
  
#duplicated(rand\_filtered\_data)  
  
#then we get rid of the duplicated variables  
rand\_filtered\_data\_new <- rand\_filtered\_data[!duplicated(rand\_filtered\_data), ]

1. Explain what data preparation is required in order for the data in Jan\_2022\_Exam\_Data.csv to be suitable for this analysis.

**(4 marks)**

### Answer:

1. Data Collection for the purpose we are trying to solve:

In this case, since we are only interested in data with the following properties, the mileage less than 60000, Manual transmission, Petrol engine (fuelType) and Costing less than £200 in annual Vehicle Tax.We have to filter and store them in a new data frame.

1. Metedata:

Here we read and understand better the data, what it contains, the kind of problem we are trying to solve and what we’ll be needing to solve this problem.

1. Data Cleansing:

Here we get rid of null values, replacing them with NA.

4 Understand the structure of the data.

1. Implement the required data preparation in the code chunk below:

**(7 marks)**

### Answer:

All, of the code chunks above my explanation are part of data preparation, which involves understanding the data and selecting the variables we’ll be needing to solve our problem.. Here we selected certain columns, we filtered them and then selected a random sample from our filtered data which we’ll be needing to proceed to our EDA.

# in addition to the data preparation we have done above we can do some more to make sure our data is ready for EDA  
  
  
#function to replace missing values with NA  
  
replace\_with\_na <- function(column, na\_value){  
 column[column %in% na\_value] <- NA  
 return((column))  
}  
  
# now let's call the function on our data for each quantitative variable column  
  
rand\_filtered\_data\_new$year <- replace\_with\_na(rand\_filtered\_data\_new$year, 0)  
  
rand\_filtered\_data\_new$price <- replace\_with\_na(rand\_filtered\_data\_new$price, 0)  
  
rand\_filtered\_data\_new$mileage <- replace\_with\_na(rand\_filtered\_data\_new$mileage, 0)  
  
rand\_filtered\_data\_new$transmission <- replace\_with\_na(rand\_filtered\_data\_new$transmission, 0)  
  
rand\_filtered\_data\_new$fuelType <- replace\_with\_na(rand\_filtered\_data\_new$fuelType, 0)  
  
rand\_filtered\_data\_new$tax <- replace\_with\_na(rand\_filtered\_data\_new$tax, 0)  
  
rand\_filtered\_data\_new$mpg <- replace\_with\_na(rand\_filtered\_data\_new$mpg, 0)  
  
rand\_filtered\_data\_new$engineSize <- replace\_with\_na(rand\_filtered\_data\_new$engineSize, 0)  
  
#let's see if changes were made by printing our new data frame  
  
head(rand\_filtered\_data\_new)

brand model year price transmission mileage fuelType tax mpg  
13434 Mercedes CL Class 2015 16600 Manual 34336 Petrol 145 48.7  
897 Audi A1 2020 20000 Manual 5000 Petrol 145 49.6  
4217 Ford Focus 2018 14998 Manual 10306 Petrol 145 58.9  
5260 Ford Fiesta 2017 8998 Manual 10602 Petrol 145 54.3  
5408 Ford Fiesta 2018 11000 Manual 33763 Petrol 145 56.5  
5359 Ford Fiesta 2014 7090 Manual 28877 Petrol NA 65.7  
 engineSize  
13434 1.6  
897 1.0  
4217 1.0  
5260 1.2  
5408 1.0  
5359 1.0

# Question 2: Exploratory Data Analysis (22 marks)

## Descriptive Statistics

1. What descriptive statistics would be appropriate for this dataset? Explain why these are useful in this context.

**(2 marks)**

### Answer:

1. We can start by getting a quick summary of the variables, mean, min, max and quantile as well as count of NA values.
2. We can also look at the range of distribution of the variables.
3. Finally we can use the describe function to get the group statistics which includes the sd.
4. Produce those descriptive statistics in the code chunk below:

**(4 marks)**

### Answer:

# summary  
summary(rand\_filtered\_data\_new)

brand model year price   
 Length:1990 Length:1990 Min. :2001 Min. : 2995   
 Class :character Class :character 1st Qu.:2016 1st Qu.: 9303   
 Mode :character Mode :character Median :2017 Median :11898   
 Mean :2017 Mean :12949   
 3rd Qu.:2018 3rd Qu.:15952   
 Max. :2020 Max. :48000   
   
 transmission mileage fuelType tax   
 Length:1990 Min. : 1 Length:1990 Min. : 20.0   
 Class :character 1st Qu.: 8468 Class :character 1st Qu.:135.0   
 Mode :character Median :15872 Mode :character Median :145.0   
 Mean :18327 Mean :126.5   
 3rd Qu.:26286 3rd Qu.:145.0   
 Max. :59964 Max. :165.0   
 NA's :183   
 mpg engineSize   
 Min. :20.90 Min. :1.000   
 1st Qu.:51.40 1st Qu.:1.000   
 Median :56.50 Median :1.000   
 Mean :55.73 Mean :1.204   
 3rd Qu.:61.40 3rd Qu.:1.400   
 Max. :70.60 Max. :5.000   
 NA's :10

#str  
  
str(rand\_filtered\_data\_new)

'data.frame': 1990 obs. of 10 variables:  
 $ brand : chr "Mercedes" "Audi" "Ford" "Ford" ...  
 $ model : chr " CL Class" " A1" " Focus" " Fiesta" ...  
 $ year : int 2015 2020 2018 2017 2018 2014 2019 2013 2017 2017 ...  
 $ price : int 16600 20000 14998 8998 11000 7090 18500 8698 11498 9620 ...  
 $ transmission: chr "Manual" "Manual" "Manual" "Manual" ...  
 $ mileage : int 34336 5000 10306 10602 33763 28877 19638 9734 17897 22431 ...  
 $ fuelType : chr "Petrol" "Petrol" "Petrol" "Petrol" ...  
 $ tax : int 145 145 145 145 145 NA 150 NA 30 145 ...  
 $ mpg : num 48.7 49.6 58.9 54.3 56.5 65.7 40.9 65.7 54.3 53.3 ...  
 $ engineSize : num 1.6 1 1 1.2 1 1 1.6 1 1 1 ...

#describe  
  
describe(rand\_filtered\_data\_new)

vars n mean sd median trimmed mad min  
brand\* 1 1990 2.68 0.77 3.0 2.80 0.00 1.0  
model\* 2 1990 16.00 6.19 17.0 16.19 1.48 1.0  
year 3 1990 2017.20 1.72 2017.0 2017.38 1.48 2001.0  
price 4 1990 12948.82 4992.32 11898.5 12440.66 4590.87 2995.0  
transmission\* 5 1990 1.00 0.00 1.0 1.00 0.00 1.0  
mileage 6 1990 18326.78 12779.53 15871.5 17123.76 12517.59 1.0  
fuelType\* 7 1990 1.00 0.00 1.0 1.00 0.00 1.0  
tax 8 1807 126.48 42.15 145.0 136.04 0.00 20.0  
mpg 9 1990 55.73 7.91 56.5 56.43 7.56 20.9  
engineSize 10 1980 1.20 0.31 1.0 1.15 0.00 1.0  
 max range skew kurtosis se  
brand\* 4.0 3.0 -1.35 0.75 0.02  
model\* 33.0 32.0 -0.42 0.65 0.14  
year 2020.0 19.0 -1.29 4.91 0.04  
price 48000.0 45005.0 1.05 1.69 111.91  
transmission\* 1.0 0.0 NaN NaN 0.00  
mileage 59964.0 59963.0 0.80 0.14 286.48  
fuelType\* 1.0 0.0 NaN NaN 0.00  
tax 165.0 145.0 -1.90 1.81 0.99  
mpg 70.6 49.7 -0.64 0.10 0.18  
engineSize 5.0 4.0 3.10 23.95 0.01

#standard deviation of tax and engineSize  
sd(rand\_filtered\_data\_new$tax, na.rm = TRUE) #solves issues with NA'S by ignoring in calculation

[1] 42.14668

sd(rand\_filtered\_data\_new$engineSize, na.rm = TRUE) #solves issues with NA'S by ignoring in calculation

[1] 0.3095028

1. What have those descriptive statistics told you – and how does this inform the analysis that you would undertake on this data or any additional data cleaning requirements?

**(4 marks)**

### Answer:

1. Our summary statistics shows us our mean,min,max value, 1st and 3rd quantile and also the number of NA’s. This can be really helpful to know the range of values of our variables.

* The summary of the price tells us that the maximum price of a used car in our dataset is 48000 and the minimum price is 2995, with a mean of 12949.
* Because we filtered our mileage to be < 6000, we are not expecting to get any variable with a higher mileage than that, if not we might have to go back to our data to fix this issue.
* Also for our qualitative variables fuelType and transmission,brand and model we should be expecting characters for our class and also no statistical calculations.
* However, our tax has 183 NA’s and engineSize has 10 NA’s, which means we might have to carry out further statistical calculations on it, as our sd might be wrong if we use the describe function.

1. The str function helps us view the internal structure of our data.
2. The describe function gives us the number of observations, our mean and sd.
3. Confirming the sd of tax and engineSize. As we can see our sd from the single sd is the same as that from the describe function so we are in the clear.

From our observation, we can say there’s a slightly even distribution in the price and mileage. But then mpg proves to me evenly distributed with a min value of 20.90, a max value of 70.60 and a mean almost in between this two values at 56.50. Now We can go ahead and perform some exploratory graphs and gain more insight.

## Exploratory Graphs

1. What exploratory graphs would be appropriate for this dataset? Explain why these are useful in this context.

**(2 marks)**

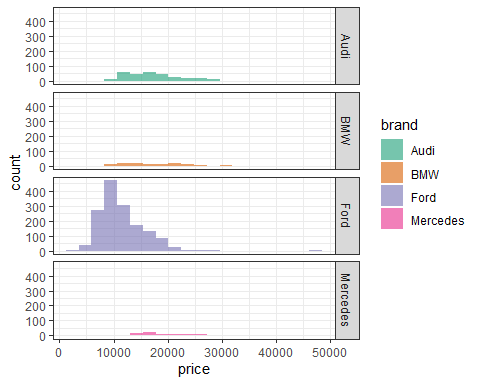
### Answer:

1. A histogram will be appropriate to view the distribution of our quantitative variables.
2. A boxplot will also come in handly to see if the mean is evenly distributed.
3. Now produce those exploratory graphs in the code chunk below:

**(4 marks)**

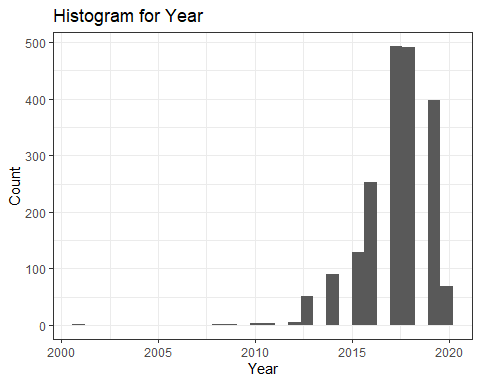
### Answer:

#Histogram for Brand against the price  
ggplot(rand\_filtered\_data\_new,aes(x=price,fill=brand))+  
 geom\_histogram(alpha=0.6, position = 'identity',bins=20) +  
 scale\_fill\_brewer(palette="Dark2")+theme\_bw() + facet\_grid(rows = vars(brand))

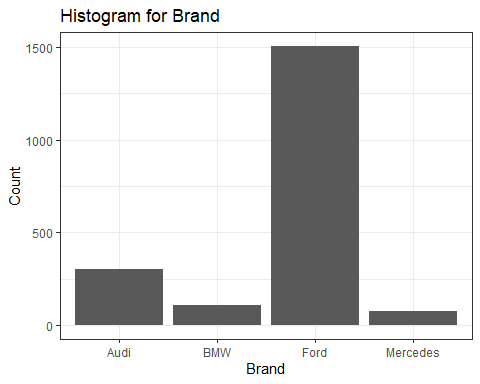


#Histogram for Year  
rand\_filtered\_data\_new %>%  
 ggplot(aes(x=year)) +   
 geom\_histogram() +   
 labs(title = "Histogram for Year") + xlab("Year") + ylab("Count") + theme\_bw()

`stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

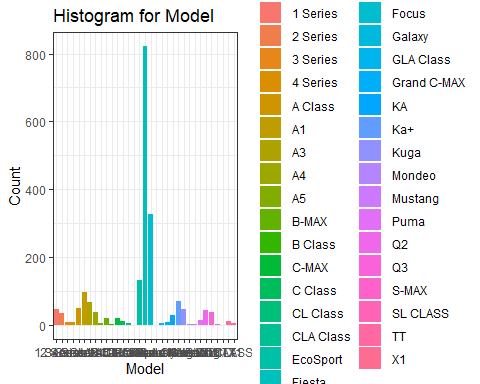


#Histogram for Brand  
rand\_filtered\_data\_new %>%  
 ggplot(aes(x=brand)) +   
 geom\_bar() +   
 labs(title = "Histogram for Brand") + xlab("Brand") + ylab("Count") + theme\_bw()



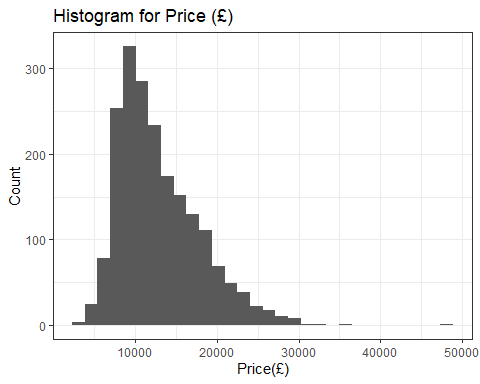
#Histogram for Model  
rand\_filtered\_data\_new %>%  
 ggplot(aes(x=model, fill=model)) +   
 geom\_bar(bin=.5) +   
 labs(title = "Histogram for Model") + xlab("Model") + ylab("Count") + theme\_bw()

Warning: Ignoring unknown parameters: bin



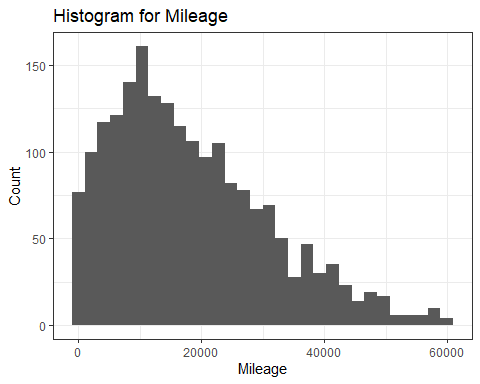
#Histogram for Price  
rand\_filtered\_data\_new %>%  
 ggplot(aes(x=price)) +   
 geom\_histogram() +   
 labs(title = "Histogram for Price (£)") + xlab("Price(£)") + ylab("Count") + theme\_bw()

`stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



#Histogram for Mileage  
rand\_filtered\_data\_new %>%  
 ggplot(aes(x=mileage)) +   
 geom\_histogram() +   
 labs(title = "Histogram for Mileage") + xlab("Mileage") + ylab("Count") + theme\_bw()

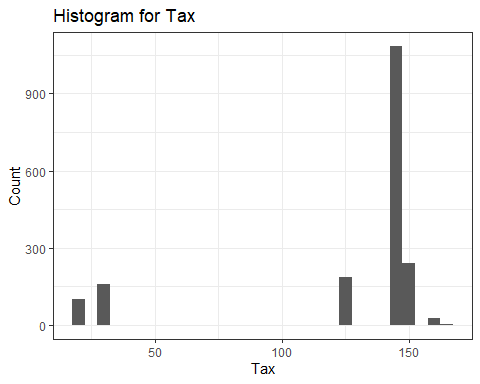
`stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



#Histogram for Tax  
rand\_filtered\_data\_new %>%  
 ggplot(aes(x=tax)) +   
 geom\_histogram() +   
 labs(title = "Histogram for Tax") + xlab("Tax") + ylab("Count") + theme\_bw()

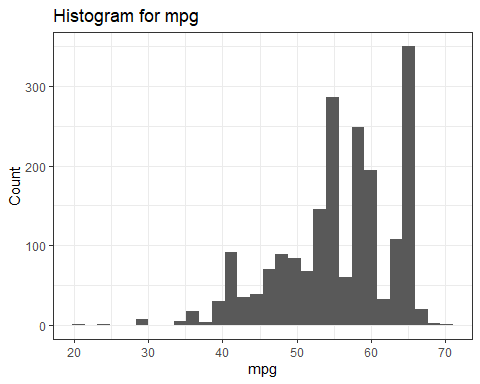
`stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

Warning: Removed 183 rows containing non-finite values (stat\_bin).



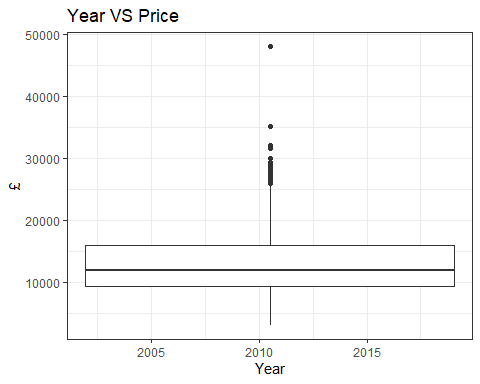
#Histogram for mpg  
rand\_filtered\_data\_new %>%  
 ggplot(aes(x=mpg)) +   
 geom\_histogram() +   
 labs(title = "Histogram for mpg") + xlab("mpg") + ylab("Count") + theme\_bw()

`stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



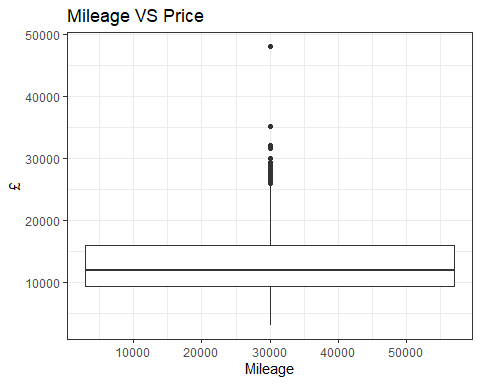
rand\_filtered\_data\_new %>%  
 ggplot(aes(x=year, y = price)) +   
 geom\_boxplot() +   
 labs(title = "Year VS Price") +   
 xlab("Year") +   
 ylab("£") +   
 theme\_bw()

Warning: Continuous x aesthetic -- did you forget aes(group=...)?



rand\_filtered\_data\_new %>%  
 ggplot(aes(x=mileage, y = price)) +   
 geom\_boxplot() +   
 labs(title = "Mileage VS Price") +   
 xlab("Mileage") +   
 ylab("£") +   
 theme\_bw()

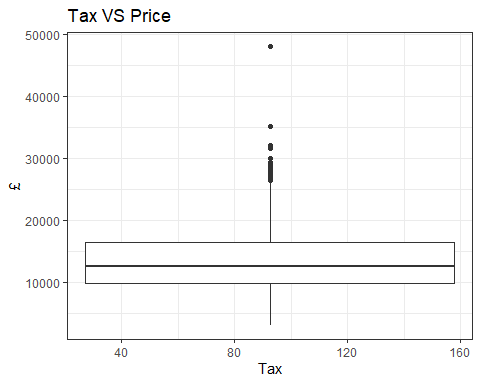
Warning: Continuous x aesthetic -- did you forget aes(group=...)?



rand\_filtered\_data\_new %>%  
 ggplot(aes(x=tax, y = price)) +   
 geom\_boxplot() +   
 labs(title = "Tax VS Price") +   
 xlab("Tax") +   
 ylab("£") +   
 theme\_bw()

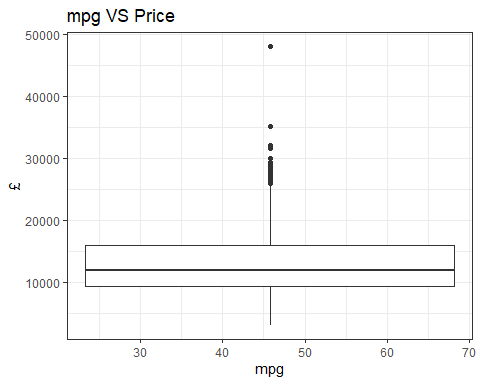
Warning: Continuous x aesthetic -- did you forget aes(group=...)?

Warning: Removed 183 rows containing missing values (stat\_boxplot).



rand\_filtered\_data\_new %>%  
 ggplot(aes(x=mpg, y = price)) +   
 geom\_boxplot() +   
 labs(title = "mpg VS Price") +   
 xlab("mpg") +   
 ylab("£") +   
 theme\_bw()

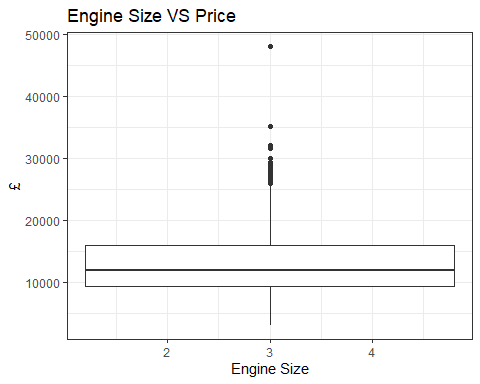
Warning: Continuous x aesthetic -- did you forget aes(group=...)?



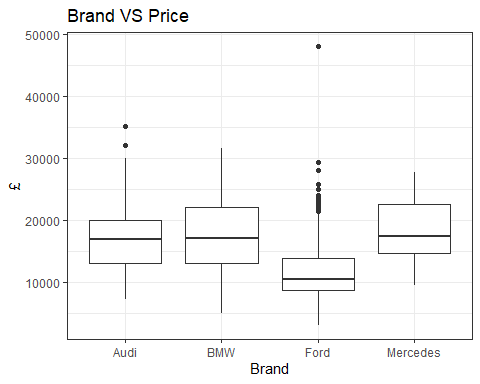
rand\_filtered\_data\_new %>%  
 ggplot(aes(x=engineSize, y = price)) +   
 geom\_boxplot() +   
 labs(title = "Engine Size VS Price") +   
 xlab("Engine Size") +   
 ylab("£") +   
 theme\_bw()

Warning: Continuous x aesthetic -- did you forget aes(group=...)?

Warning: Removed 10 rows containing missing values (stat\_boxplot).



rand\_filtered\_data\_new %>%  
 ggplot(aes(x=brand, y = price)) +   
 geom\_boxplot() +   
 labs(title = "Brand VS Price") +   
 xlab("Brand") +   
 ylab("£") +   
 theme\_bw()



1. Interpret these exploratory graphs. How do these graphs inform your subsequent analysis?

**(4 marks)**

### Answer:

Our first histogram Brand against the price shows us a lot of things about the brand of cars and how they are priced. Mercedes, have more expensive cars, resulting in less purchase of their used cars, same goes for Audi and BMW in between. But Ford has more purchases compared to the other 3 brands because the pricing of the car is fair compared to them.

Year is distributed to the right, which shows there is a demand for latest cars, this is also the same for mpg.

For our price, between 0 and 30000 the demand for moderately priced cars is high.

For mileage, it shows that more people bought cars with lower mileage as there is generally less wear and tear with lower mileage cars.

We can see that price, year and mpg are slightly correlated, as one increases the other reduces, which could mean a possible correlation. And as for the tax histogram.

Also from our box plot it is evident that our mean is evenly distributed.

## Correlations

1. What linear correlations are present within this data?

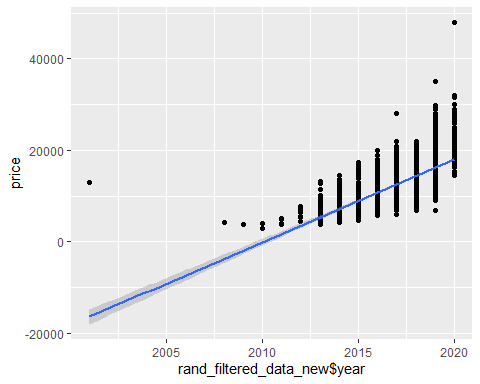
**(2 marks)**

### Answer:

#scatter plot checking for linear correlation between Year and price  
ggplot(data=rand\_filtered\_data\_new,  
 aes(x=rand\_filtered\_data\_new$year, y=price)) +geom\_point()+geom\_smooth(method="lm")

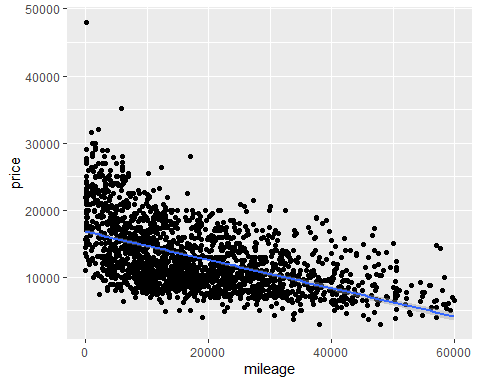
Warning: Use of `rand\_filtered\_data\_new$year` is discouraged. Use `year`  
instead.  
  
Warning: Use of `rand\_filtered\_data\_new$year` is discouraged. Use `year`  
instead.

`geom\_smooth()` using formula 'y ~ x'



#scatter plot checking for linear correlation between mileage and price  
ggplot(data=rand\_filtered\_data\_new,  
 aes(x=mileage, y=price)) +geom\_point()+geom\_smooth(method="lm")

`geom\_smooth()` using formula 'y ~ x'

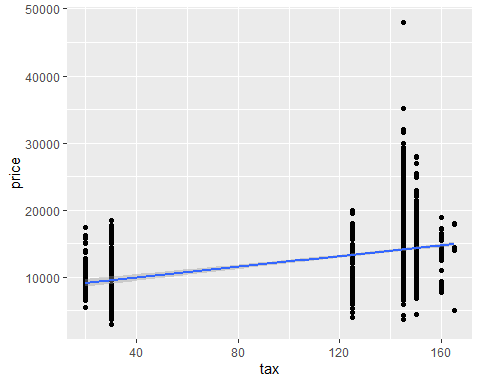


#scatter plot checking for linear correlation between price and tax  
ggplot(data=rand\_filtered\_data\_new,  
 aes(x=tax, y=price)) +geom\_point()+geom\_smooth(method="lm")

`geom\_smooth()` using formula 'y ~ x'

Warning: Removed 183 rows containing non-finite values (stat\_smooth).

Warning: Removed 183 rows containing missing values (geom\_point).

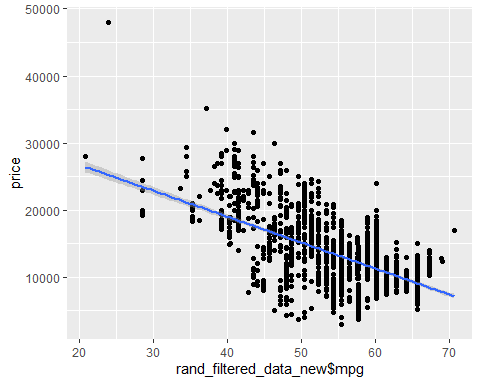


#scatter plot checking for linear correlation between mpg and price  
ggplot(data=rand\_filtered\_data\_new,  
 aes(x=rand\_filtered\_data\_new$mpg, y=price)) +geom\_point()+geom\_smooth(method="lm")

Warning: Use of `rand\_filtered\_data\_new$mpg` is discouraged. Use `mpg` instead.

Warning: Use of `rand\_filtered\_data\_new$mpg` is discouraged. Use `mpg` instead.

`geom\_smooth()` using formula 'y ~ x'

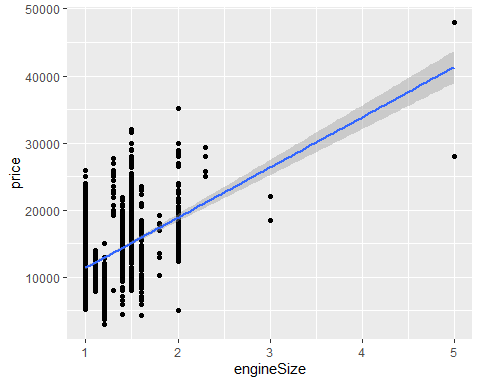


#scatter plot checking for linear correlation between engineSize and price  
ggplot(data=rand\_filtered\_data\_new,  
 aes(x=engineSize, y=price)) +geom\_point()+geom\_smooth(method="lm")

`geom\_smooth()` using formula 'y ~ x'

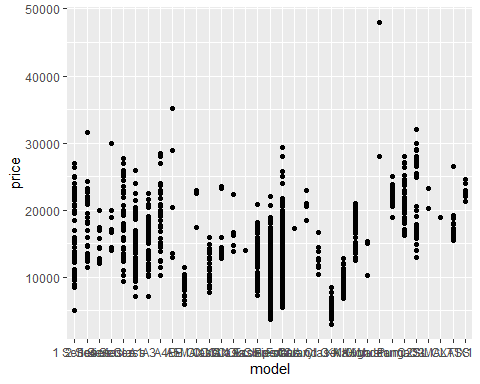
Warning: Removed 10 rows containing non-finite values (stat\_smooth).

Warning: Removed 10 rows containing missing values (geom\_point).



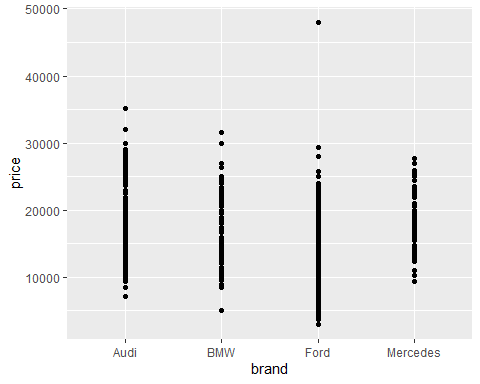
#scatter plot checking for linear correlation between Model and price  
ggplot(data=rand\_filtered\_data\_new,  
 aes(x=model, y=price)) +geom\_point()+geom\_smooth(method="lm")

`geom\_smooth()` using formula 'y ~ x'



#scatter plot checking for linear correlation between Brand and price  
ggplot(data=rand\_filtered\_data\_new,  
 aes(x=brand, y=price)) +geom\_point()+geom\_smooth(method="lm")

`geom\_smooth()` using formula 'y ~ x'

 From our scatter plot there is a negative linear relationship between mileage and mpg to price.Also there is a positive relationship between year and price. And also our qualitative variables don’t show any patterns of course.

# let's further analyse which of the explanatory variables has a correlation with y using cor()  
cor(rand\_filtered\_data\_new$mpg, rand\_filtered\_data\_new$price)

[1] -0.6133162

cor(rand\_filtered\_data\_new$mileage, rand\_filtered\_data\_new$price)

[1] -0.5381298

cor(rand\_filtered\_data\_new$year, rand\_filtered\_data\_new$price)

[1] 0.6232788

cor(rand\_filtered\_data\_new$tax, rand\_filtered\_data\_new$price)

[1] NA

cor(rand\_filtered\_data\_new$engineSize, rand\_filtered\_data\_new$price)

[1] NA

Our cor() shows that mileage, mpg and year have a correlation with y.

# Question 3: Bivariate relationship (14 marks)

1. Which of the potential explanatory variables has the strongest linear relationship with the dependent variable?

**(1 mark)**

### Answer:

# let's use cor.test() to validate our correlation  
cor.test(rand\_filtered\_data\_new$mpg, rand\_filtered\_data\_new$price)

Pearson's product-moment correlation  
  
data: rand\_filtered\_data\_new$mpg and rand\_filtered\_data\_new$price  
t = -34.622, df = 1988, p-value < 2.2e-16  
alternative hypothesis: true correlation is not equal to 0  
95 percent confidence interval:  
 -0.6400091 -0.5851448  
sample estimates:  
 cor   
-0.6133162

cor.test(rand\_filtered\_data\_new$mileage, rand\_filtered\_data\_new$price)

Pearson's product-moment correlation  
  
data: rand\_filtered\_data\_new$mileage and rand\_filtered\_data\_new$price  
t = -28.467, df = 1988, p-value < 2.2e-16  
alternative hypothesis: true correlation is not equal to 0  
95 percent confidence interval:  
 -0.5686251 -0.5061574  
sample estimates:  
 cor   
-0.5381298

cor.test(rand\_filtered\_data\_new$year, rand\_filtered\_data\_new$price)

Pearson's product-moment correlation  
  
data: rand\_filtered\_data\_new$year and rand\_filtered\_data\_new$price  
t = 35.537, df = 1988, p-value < 2.2e-16  
alternative hypothesis: true correlation is not equal to 0  
95 percent confidence interval:  
 0.5956512 0.6494334  
sample estimates:  
 cor   
0.6232788

From our cor.test(), the year is the explanatory variable with the strongest relationship to our dependent variable(price). It shows a positive relationship and has a strong confidence interval compared to the other variables.

1. Create a linear model to model this relationship.

**(2 marks)**

### Answer:

#linear model  
model1 <- lm(rand\_filtered\_data\_new$price ~ rand\_filtered\_data\_new$year, data = rand\_filtered\_data\_new)  
summary(model1)

Call:  
lm(formula = rand\_filtered\_data\_new$price ~ rand\_filtered\_data\_new$year,   
 data = rand\_filtered\_data\_new)  
  
Residuals:  
 Min 1Q Median 3Q Max   
 -9311 -2793 -898 2241 29976   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -3.645e+06 1.029e+05 -35.41 <2e-16 \*\*\*  
rand\_filtered\_data\_new$year 1.813e+03 5.102e+01 35.54 <2e-16 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 3905 on 1988 degrees of freedom  
Multiple R-squared: 0.3885, Adjusted R-squared: 0.3882   
F-statistic: 1263 on 1 and 1988 DF, p-value: < 2.2e-16

1. Explain and interpret the model:

**(3 marks)**

### Answer:

From our model our R-squared shows 39% variation within our dependent variable, which shows that the year a car was produced explains some degree of variation in the price of used cars.

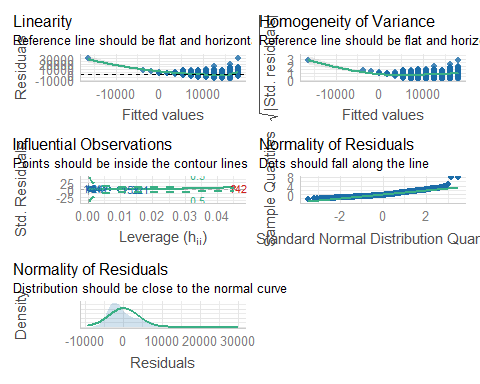
1. Comment on the performance of this model, including comments on overall model fit and the validity of model assumptions. Include any additional code required for you to make these comments in the code chunk below.

**(4 marks)**

### Answer:

check\_model(model1)

Warning: Using `$` in model formulas can produce unexpected results. Specify your model  
 using the `data` argument instead.  
 Try: price ~ year, data =  
 rand\_filtered\_data\_new

 From our model fit, the linearity assumption is met with the residuals evenly distributed following the normal curves, all also falling along the line. Are influential observations points are inside the contour lines. With this we can say our assumptions are met, but we can further prove this.

## Bootstrap

1. Use bootstrapping on this model to obtain a 95% confidence interval of the estimate of the slope parameter.

**(4 marks)**

### Answer:

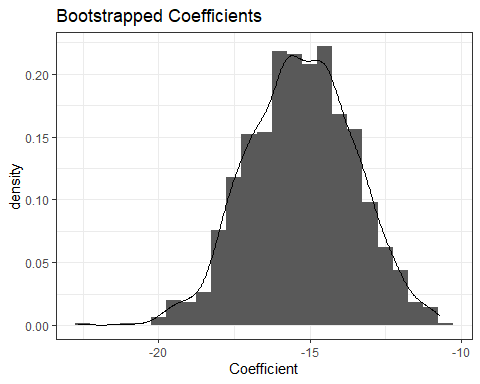
# Set up bootstrap  
Nbootstrap<- 1000 # 1000 takes half a minute  
coeff\_tax <-rep(NA,Nbootstrap)  
  
# Perform bootstrap  
set.seed(21044874)  
for(i in seq\_len(Nbootstrap)){  
 usevalues<-sample(seq\_along(rand\_filtered\_data\_new$tax),size=length(rand\_filtered\_data\_new$tax),replace=TRUE)  
   
 bootstrap.sample<- rand\_filtered\_data\_new[usevalues,]  
 model\_full <- lm(price ~ mileage + mpg + tax + engineSize + year, data=bootstrap.sample)  
 model\_boot <- stepAIC(model\_full, trace=FALSE)  
 coeff\_tax[i] <- model\_boot$coef["tax"]  
}  
  
# Where the coefficient is NA it means the tax variable wasn't selected  
# Set NA's to 0  
mean(is.na(coeff\_tax))

[1] 0

coeff\_tax[is.na(coeff\_tax)] <- 0  
mean(coeff\_tax==0)

[1] 0

# Histogram  
Bootstrap <- data.frame(Coefficient=coeff\_tax)  
ggplot(Bootstrap, aes(x=Coefficient)) +  
geom\_histogram(aes(y = ..density..),binwidth = .5)+  
 geom\_density()+  
ggtitle("Bootstrapped Coefficients")+  
theme\_bw()



# 95% bootstrap confidence interval  
quantile(coeff\_tax,c(.025,0.975))

2.5% 97.5%   
-18.68032 -12.05654

# Question 4: Multivariable relationship (10 marks)

Create a model with all of the appropriate remaining explanatory variables included:

model2 <- lm(price ~ mileage + mpg + tax + engineSize + year, data = rand\_filtered\_data\_new)  
summary(model2)

Call:  
lm(formula = price ~ mileage + mpg + tax + engineSize + year,   
 data = rand\_filtered\_data\_new)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-11294 -1548 11 1464 11107   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -3.715e+06 1.297e+05 -28.648 < 2e-16 \*\*\*  
mileage -5.196e-02 7.472e-03 -6.953 4.99e-12 \*\*\*  
mpg -1.685e+02 1.247e+01 -13.506 < 2e-16 \*\*\*  
tax -1.514e+01 1.900e+00 -7.970 2.80e-15 \*\*\*  
engineSize 5.626e+03 2.976e+02 18.903 < 2e-16 \*\*\*  
year 1.851e+03 6.421e+01 28.821 < 2e-16 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 2782 on 1791 degrees of freedom  
 (193 observations deleted due to missingness)  
Multiple R-squared: 0.6905, Adjusted R-squared: 0.6896   
F-statistic: 799.2 on 5 and 1791 DF, p-value: < 2.2e-16

1. Explain and interpret the model:

**(4 marks)**

### Answer:

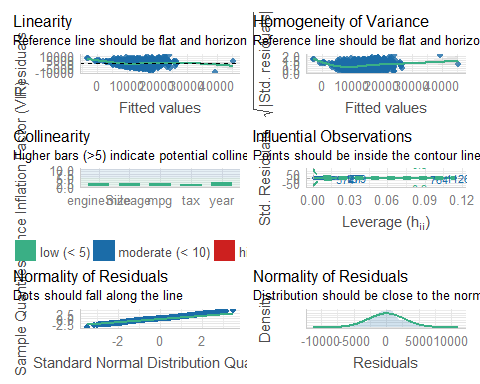
Compared to our model created with our bivariate relationship. This model produces an R-squared of 69%, showing a higher variation within our dependent variable.

1. Comment on the performance of this model, including comments on overall model fit and the validity of model assumptions. Include any additional code required for you to make these comments in the code chunk below.

**(4 marks)**

### Answer:

check\_model(model2)



1. What general concerns do you have regarding this model?

**(2 marks)**

### Answer:

Our linearity and homogeneity is scattered and does not follow the line which could mean overfitting. So it’s advisible to drop some of the independent variables and work with the one with the strongest correlation to the dependent variable.

# Question 5: Model simplification (8 marks)

1. What approaches for model simplification would you consider implementing and why?

**(4 marks)**

### Answer:

* We can use an appropriate variable selection method like AIC/BIC or considering the size of coefficients.
* We can split our data into training and test data for more validation.
* Regularization methods can also be used which involves, Lasso and Ridge regression.

1. What are the potential advantages of simplifying a model?

**(2 marks)**

### Answer:

Model simplification helps us understand the main data without altering it, thereby making closely accurate predictions without altering our data.

1. What are the potential disadvantages of simplifying a model?

**(2 marks)**

### Answer:

* Over fitting
* Under fitting

# Question 6: Reporting (35 marks)

A client is looking to purchase a used Skoda Superb (registration year either 2018 or 2019, manual transmission, petrol engine) and wants to understand what factors influence the expected price of a used car, (and how they influence the price).

Write a short report of 300-500 words for the client.

Furthermore, include an explanation as to which model you would recommend, and why you have selected that model.

Comment on any suggestions for alterations to the model that would be appropriate to consider.

Highlight what may or may not be directly transferable from the scenario analysed in Questions 1 to 5.

### Answer:

## Report for a Client

In my recent analysis of the factors that could affect the estimated price of a used petrol car, I found out that miles per gallon, mileage and year are considerably high factors that affect the price of a car.

The price of a car increases as the year of production of the car increases. This is common knowledge, especially when it comes to a used car. A car made in 2010 must have been used longer than a car made in 2018, meaning that the car made in 2018 is new and should be in good shape compared to a car made in an earlier year. Some parts of the older car could have been changed over the course of the year, which could mean you are getting a less durable car compared to the newer ones. Putting this into consideration, the price of newer cars would cost more than an older one because it is “more durable”, has been used for a shorter time and in good shape.

Another thing to consider is the mileage. This is a significant factor you should consider. From my analysis, people bought cars with lower mileage, this is because it means the car has been driven fewer miles on average compared to how old the car is. This could also mean that cars with lower mileage would cost more as they are more efficient and newer compared to cars with higher mileage.

And then the last thing to consider is the miles per gallon. From my analysis, more people bought cars with higher miles per gallon(mpg). A high mpg means the car consumes less fuel, and this could be a good thing as it reduces the cost of maintaining the car. My analysis of manual transmission cars showed that cars with higher mpg are cheaper than those with lower mpg, and it could be an advantage as you can buy a car with a high mpg at a cheaper rate.

Furthermore, I would suggest a Ford Fiesta, as my recently conducted analysis shows that they are cheaper, and more people bought them compared to the other brand of cars that I analysed. Also, you would want to consider the year of the model as well, as it means you would get a more durable car and 2018 or 2019 would be fine. Another thing to consider when selecting this model is mpg, as this would influence the cost of maintenance of the car. And finally a very important thing to consider the mileage because for used cars, the condition of the car is measured by the mileage and we would be considering a car with a low mileage average compared to the year of production here.

# Session Information

Do not edit this part. Make sure that you compile your document so that the information about your session (including software / package versions) is included in your submission.

sessionInfo()

R version 4.0.5 (2021-03-31)  
Platform: x86\_64-w64-mingw32/x64 (64-bit)  
Running under: Windows 10 x64 (build 19042)  
  
Matrix products: default  
  
locale:  
[1] LC\_COLLATE=English\_United Kingdom.1252   
[2] LC\_CTYPE=English\_United Kingdom.1252   
[3] LC\_MONETARY=English\_United Kingdom.1252  
[4] LC\_NUMERIC=C   
[5] LC\_TIME=English\_United Kingdom.1252   
  
attached base packages:  
[1] stats graphics grDevices utils datasets methods base   
  
other attached packages:  
 [1] glmnet\_4.1-3 Matrix\_1.3-2 performance\_0.8.0 corrplot\_0.92   
 [5] patchwork\_1.1.1 psych\_2.1.9 summarytools\_1.0.0 MASS\_7.3-53.1   
 [9] ggplot2\_3.3.5 lubridate\_1.8.0 dplyr\_1.0.7 readxl\_1.3.1   
[13] readr\_2.1.0 kableExtra\_1.3.4   
  
loaded via a namespace (and not attached):  
 [1] nlme\_3.1-152 matrixStats\_0.61.0 insight\_0.14.5 webshot\_0.5.2   
 [5] RColorBrewer\_1.1-2 httr\_1.4.2 tools\_4.0.5 backports\_1.4.1   
 [9] utf8\_1.2.2 R6\_2.5.1 DBI\_1.1.1 mgcv\_1.8-34   
[13] colorspace\_2.0-2 withr\_2.4.2 tidyselect\_1.1.1 mnormt\_2.0.2   
[17] compiler\_4.0.5 rvest\_1.0.2 see\_0.6.8 xml2\_1.3.2   
[21] labeling\_0.4.2 bayestestR\_0.11.5 scales\_1.1.1 checkmate\_2.0.0   
[25] systemfonts\_1.0.3 stringr\_1.4.0 digest\_0.6.28 rmarkdown\_2.11   
[29] svglite\_2.0.0 base64enc\_0.1-3 pkgconfig\_2.0.3 htmltools\_0.5.2   
[33] fastmap\_1.1.0 highr\_0.9 rlang\_0.4.11 rstudioapi\_0.13   
[37] pryr\_0.1.5 shape\_1.4.6 generics\_0.1.0 farver\_2.1.0   
[41] magrittr\_2.0.1 rapportools\_1.0 Rcpp\_1.0.7 munsell\_0.5.0   
[45] fansi\_0.5.0 lifecycle\_1.0.1 stringi\_1.7.5 yaml\_2.2.1   
[49] plyr\_1.8.6 grid\_4.0.5 parallel\_4.0.5 crayon\_1.4.1   
[53] lattice\_0.20-41 splines\_4.0.5 pander\_0.6.4 hms\_1.1.1   
[57] magick\_2.7.3 tmvnsim\_1.0-2 knitr\_1.36 pillar\_1.6.3   
[61] tcltk\_4.0.5 codetools\_0.2-18 glue\_1.4.2 evaluate\_0.14   
[65] vctrs\_0.3.8 tzdb\_0.2.0 foreach\_1.5.1 cellranger\_1.1.0   
[69] gtable\_0.3.0 purrr\_0.3.4 tidyr\_1.1.4 assertthat\_0.2.1   
[73] datawizard\_0.2.1 xfun\_0.26 survival\_3.2-10 viridisLite\_0.4.0   
[77] tibble\_3.1.5 iterators\_1.0.13 ellipsis\_0.3.2