Primer momento de retroalimentación

Samantha Daniela Guanipa Ugas A01703936

2023-08-23

Planteamiento del problema

Una empresa automovilística china aspira a entrar en el mercado estadounidense. Desea establecer allí una unidad de fabricación y producir automóviles localmente para competir con sus contrapartes estadounidenses y europeas. Contrataron una empresa de consultoría de automóviles para identificar los principales factores de los que depende el precio de los automóviles, específicamente, en el mercado estadounidense, ya que pueden ser muy diferentes del mercado chino. Esencialmente, la empresa quiere saber:

1. Qué variables son significativas para predecir el precio de un automóvil
2. Qué tan bien describen esas variables el precio de un automóvil

```
M=read.csv("precios_autos.csv") #leer la base de datos
```

Teniendo la base de datos proporcionada, se subirá y será leída

Análisis de la base de datos

- Exploración de la base de datos

1. Cálculo de las medidas estadísticas de las variables cuantitativas y cualitativas

```
# Se crea un data frame con las variables numericas

numerical_df <- M %>% select(wheelbase, carlength, carwidth, carheight, curbweight, enginesize, stroke,
head(numerical_df)
```

a) Cálculo de las medidas estadísticas de las variables cuantitativas

wheelbase	carlength	carwidth	carheight	curbweight	enginesize	stroke	compressionratio	horsepower	peakrpm	citympg	highwaympg	price
88.6	168.8	64.1	48.8	2548	130	2.68	9.0	111	5000	21	27	13495
88.6	168.8	64.1	48.8	2548	130	2.68	9.0	111	5000	21	27	16500
94.5	171.2	65.5	52.4	2823	152	3.47	9.0	154	5000	19	26	16500
99.8	176.6	66.2	54.3	2337	109	3.40	10.0	102	5500	24	30	13950
99.4	176.6	66.4	54.3	2824	136	3.40	8.0	115	5500	18	22	17450
99.8	177.3	66.3	53.1	2507	136	3.40	8.5	110	5500	19	25	15250

```
#Se calcula el resumen de la variable, su desviacion estandar, su varianza, su kurtosis y su sesgo

m0 <- round(c(as.numeric(summary(numerical_df$wheelbase)), sd(numerical_df$wheelbase), var(numerical_df$
```

```r
m1 <- round(c(as.numeric(summary(numerical_df$carlength)), sd(numerical_df$carlength), var(numerical_df$

m2 <- round(c(as.numeric(summary(numerical_df$carwidth)), sd(numerical_df$carwidth), var(numerical_df$ca
m3 <- round(c(as.numeric(summary(numerical_df$carheight)), sd(numerical_df$carheight), var(numerical_df$

m4 <- round(c(as.numeric(summary(numerical_df$curbweight)), sd(numerical_df$curbweight), var(numerical_d

m5 <- round(c(as.numeric(summary(numerical_df$enginesize)), sd(numerical_df$enginesize), var(numerical_d

m6 <- round(c(as.numeric(summary(numerical_df$stroke)), sd(numerical_df$stroke), var(numerical_df$stroke

m7 <- round(c(as.numeric(summary(numerical_df$compressionratio)), sd(numerical_df$compressionratio), var

m8 <- round(c(as.numeric(summary(numerical_df$horsepower)), sd(numerical_df$horsepower), var(numerical_d

m9 <- round(c(as.numeric(summary(numerical_df$peakrpm)), sd(numerical_df$peakrpm), var(numerical_df$peak

m10 <- round(c(as.numeric(summary(numerical_df$citympg)), sd(numerical_df$citympg), var(numerical_df$city

m11 <- round(c(as.numeric(summary(numerical_df$highwaympg)), sd(numerical_df$highwaympg), var(numerical_

m12 <- round(c(as.numeric(summary(numerical_df$price)), sd(numerical_df$price), var(numerical_df$price),
```

```r
m<-as.data.frame(rbind(m0,m1,m2,m3,m4,m5,m6,m7,m8,m9,m10,m11,m12))
row.names(m)=c("Wheelbase","Car length","Car width", "Car height", "Curb weight", "Engine size", "Stroke
names(m)=c("Minimo","Q1","Mediana","Media","Q3","Máximo","Desviación Estándar", "Varianza", "Curtosis","
m
```

	Minimo	Q1	Mediana	Media	Q3	Máximo	Desviación Estándar	Varianza	Curtosis	Sesgo
Wheelbase	86.60	94.50	97.00	98.757	102.40	120.90	6.022	36.262	0.925	1.035
Car length	141.10	166.30	173.20	174.049	183.10	208.10	12.337	152.209	-0.138	0.154
Car width	60.30	64.10	65.50	65.908	66.90	72.30	2.145	4.602	0.621	0.891
Car height	47.80	52.00	54.10	53.725	55.50	59.80	2.444	5.971	-0.487	0.062
Curb weight	1488.00	2145.00	2414.00	2555.566	2935.00	4066.00	520.680	271107.874	-0.099	0.671
Engine size	61.00	97.00	120.00	126.907	141.00	326.00	41.643	1734.114	5.069	1.919
Stroke	2.07	3.11	3.29	3.255	3.41	4.17	0.314	0.098	2.043	-0.680
Compression ratio	7.00	8.60	9.00	10.143	9.40	23.00	3.972	15.777	4.999	2.573
Horse power	48.00	70.00	95.00	104.117	116.00	288.00	39.544	1563.741	2.535	1.385
Peak rpm	4150.00	4800.00	5200.00	5125.122	5500.00	6600.00	476.986	227515.304	0.026	0.074
City mpg	13.00	19.00	24.00	25.220	30.00	49.00	6.542	42.800	0.501	0.654
Highway mpg	16.00	25.00	30.00	30.751	34.00	54.00	6.886	47.423	0.367	0.532
Price	5118.00	7788.00	10295.00	13276.711	16503.00	45400.00	7988.852	63821761.578	2.891	1.752

Se puede observar el resumen de las medidas estadísticas de cada variable cuantitativa. Como ejemplo, explicaré el resumen de la variable "Wheelbase", que tiene como valor mínimo 86.60, como valor máximo 120.90, y el promedio de medidas es de 98.757 entre todos los elementos de esta clase. Sin embargo, el 25% de

los valores son iguales o menores a 94.50, y el 75% de los valores son iguales o menores a 102.40. Además, se tiene una desviación estándar de 6.22 que indica que los valores están agrupados cerca de la media. Se puede observar que la curtosis es mayor que cero, por lo que es positiva y sugiere una distribución más puntiaguda. Por último, se tiene un sesgo positivo, lo que indica que es una distribución asimétrica a la derecha. Así, cada variable tiene su significado en el presente resumen para su análisis a lo largo de este entregable.

```
library(dplyr)

# Se crea un data frame con las variables categoricas
categorical_df <- M %>% select(symboling, CarName, fueltype, carbody, drivewheel, enginelocation, engin
head(categorical_df)
```

b) Cálculo de las medidas estadísticas de las variables cualitativas

symboling	CarName	fueltype	carbody	drivewheel	enginelocation	enginetype	cylindernumber
3	alfa-romero giulia	gas	convertible	rwd	front	dohc	four
3	alfa-romero stelvio	gas	convertible	rwd	front	dohc	four
1	alfa-romero Quadrifoglio	gas	hatchback	rwd	front	ohcv	six
2	audi 100 ls	gas	sedan	fwd	front	ohc	four
2	audi 100ls	gas	sedan	4wd	front	ohc	five
2	audi fox	gas	sedan	fwd	front	ohc	five

```
# Se crean los inputs
input1 <- categorical_df$symboling
input2 <- categorical_df$carbody

# Se crea un dataframe con los inputs
df_related <- data.frame(Input1 = input1, Input2 = input2)

# Se cuenta la frecuencia de los inputs
tabla_frecuencias <- df_related %>%
  group_by(Input1, Input2) %>%
  summarise(Frequency = n()) %>%
  ungroup()
```

1) Se generarán las tablas de frecuencias con dos inputs

```
## `summarise()` has grouped output by 'Input1'. You can override using the
## `.groups` argument.
```

```
# Se nombran las columnas
namess <- c("Symboling", "Car body", "Frequency")
colnames(tabla_frecuencias) <- namess

# Tabla de frecuencias
tabla_frecuencias
```

Symboling	Car body	Frequency
-2	sedan	3
-1	hatchback	2
-1	sedan	13

Symboling	Car body	Frequency
-1	wagon	7
0	hardtop	1
0	hatchback	8
0	sedan	43
0	wagon	15
1	hardtop	1
1	hatchback	27
1	sedan	23
1	wagon	3
2	convertible	1
2	hardtop	4
2	hatchback	13
2	sedan	14
3	convertible	5
3	hardtop	2
3	hatchback	20

Como se puede observar, hay valores que no tienen un patrón como la categoría de riesgo 2 y el tipo de coche convertible que solo se repite 1 vez. Pero, sí se puede encontrar un patrón en la categoría de riesgo "0", y el tipo de coche "sedán" ya que se repite 43 veces, o por ejemplo la categoría de riesgo "1" y el tipo de coche "hatchback" que se repite 27 veces.

Esta tabla puede ayudar a tener un conocimiento de qué categoría de riesgo puede tener cada tipo de coche mayormente, ya que se sigue un patrón con los datos que tienen una frecuencia alta.

```
# Se crean los inputs
input3 <- categorical_df$symboling
input4 <- categorical_df$fueltype

# Se crea un dataframe con los inputs
df_related1 <- data.frame(Input1 = input3, Input2 = input4)

# Se cuenta la frecuencia de los inputs
tabla_frecuencias1 <- df_related1 %>%
  group_by(Input1, Input2) %>%
  summarise(Frequency = n()) %>%
  ungroup()
```

```
## `summarise()` has grouped output by 'Input1'. You can override using the
## `.groups` argument.
```

```
# Se nombran las columnas
namess1 <- c("Symboling", "Fuel type", "Frequency")
colnames(tabla_frecuencias1) <- namess1

# Tabla de frecuencias
print(tabla_frecuencias1)
```

```
## # A tibble: 10 x 3
##    Symboling `Fuel type` Frequency
##        <int> <chr>           <int>
##  1        -2 gas                 3
##  2        -1 diesel              5
##  3        -1 gas                17
```

```
##  4         0 diesel             11
##  5         0 gas                56
##  6         1 diesel              1
##  7         1 gas                53
##  8         2 diesel              3
##  9         2 gas                29
## 10         3 gas                27
```

Se puede encontrar un patrón en la siguiente tabla de frecuencias ya que se obtiene que los vehículos con categoría de riesgo "0" suelen utilizar gas con una frecuencia de 56. Por otra parte, los vehículos de una categoría de riesgo "1" también suelen usar gas con una frecuencia de 53. Por lo que se puede asumir que los carros que por su categoría se asumen como seguros suelen utilizar gas. Pero esto no es del todo claro ya que la categoría 3 se asume como riesgosa y hay 27 autos de esa categoría que utilizan gas.

Si se ve de otro punto de vista, hay más frecuencia entre los autos que usan gas de los que usan diésel sin importar su categoría de riesgo.

```r
# Se crean los inputs
input5 <- categorical_df$drivewheel
input6 <- categorical_df$cylindernumber

# Se crea un dataframe con los inputs
df_related2 <- data.frame(Input1 = input5, Input2 = input6)

# Se cuenta la frecuencia de los inputs
tabla_frecuencias2 <- df_related2 %>%
  group_by(Input1, Input2) %>%
  summarise(Frequency = n()) %>%
  ungroup()
```

```
## `summarise()` has grouped output by 'Input1'. You can override using the
## `.groups` argument.
```

```r
# Se nombran las columnas
namess2 <- c("Drive wheel", "Cylinder number", "Frequency")
colnames(tabla_frecuencias2) <- namess2

# Tabla de frecuencias
print(tabla_frecuencias2)
```

```
## # A tibble: 12 x 3
##    `Drive wheel` `Cylinder number` Frequency
##    <chr>         <chr>                 <int>
##  1 4wd           five                      2
##  2 4wd           four                      7
##  3 fwd           five                      5
##  4 fwd           four                    111
##  5 fwd           six                       3
##  6 fwd           three                     1
##  7 rwd           eight                     5
##  8 rwd           five                      4
##  9 rwd           four                     41
## 10 rwd           six                      21
## 11 rwd           twelve                    1
## 12 rwd           two                       4
```

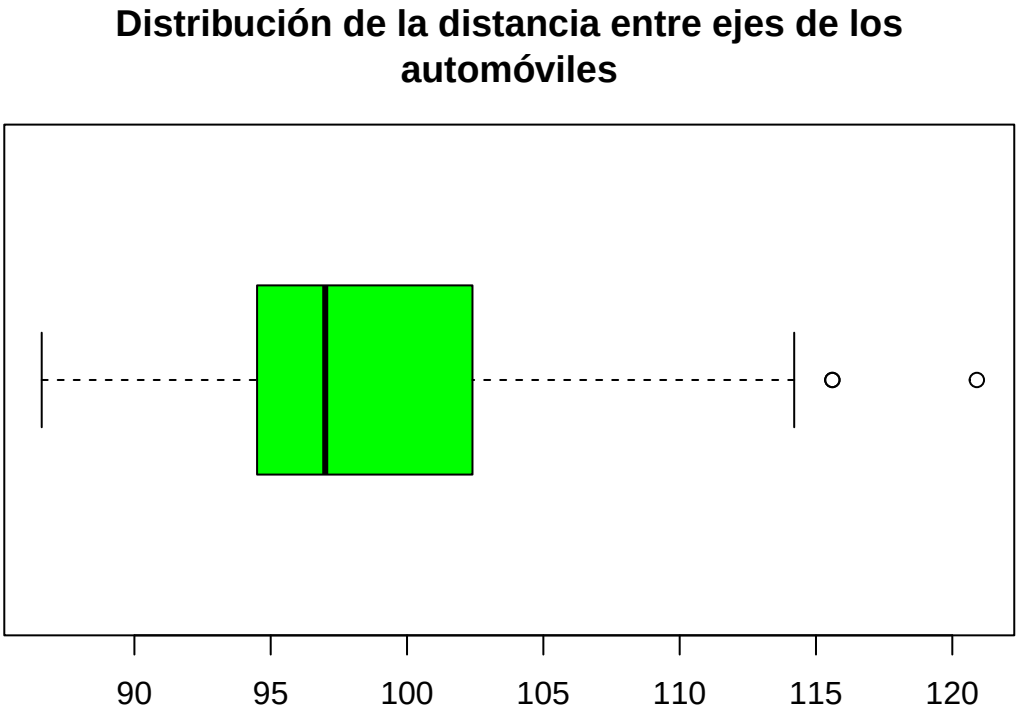
Se puede observar que hay una frecuencia alta de 111 y se puede asumir que los autos con una rueda motriz

fwd suele utilizar cuatro cilindros.

2. Exploración de los datos usando herramientas de visualización

a) Variables cuantitativas

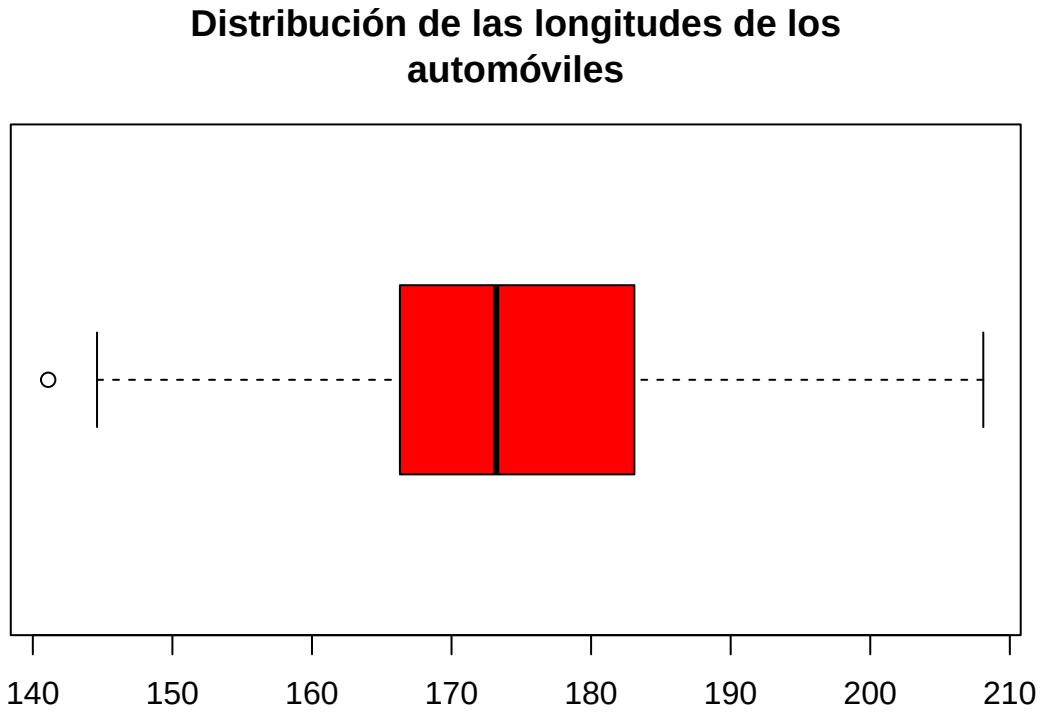
```
#Boxplot para wheelbase
boxplot(numerical_df$wheelbase, horizontal = TRUE, col = "green", main = "Distribución de la distancia e
automóviles")
```



Boxplots

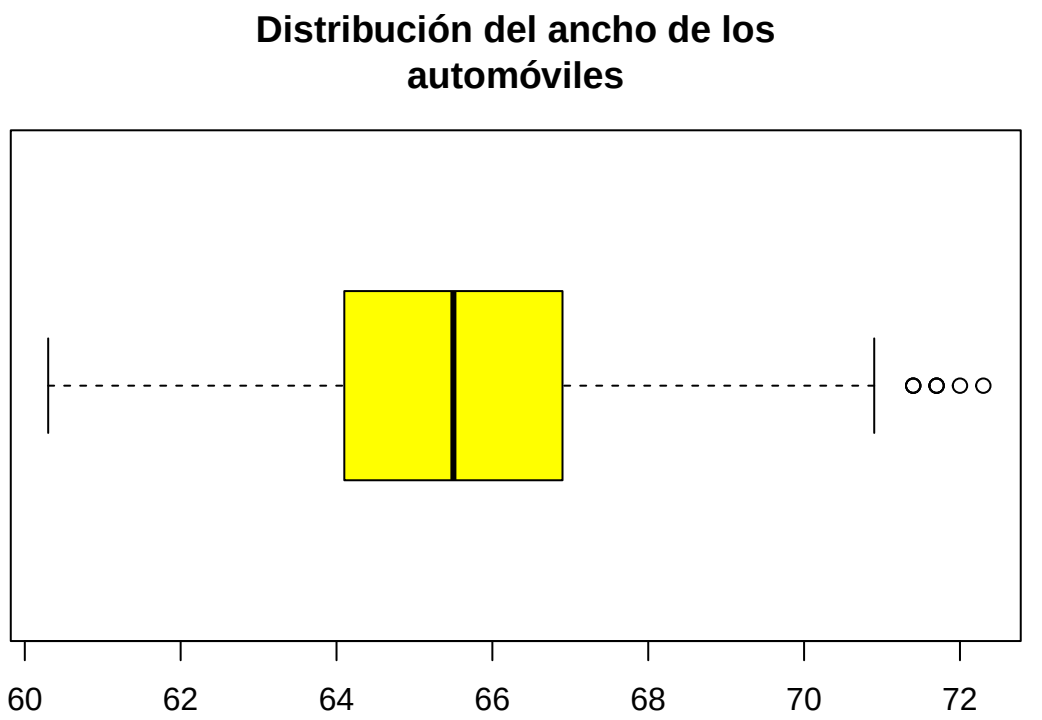
Se observa una distribución sesgada a la derecha con dos datos atípicos.

```
#Boxplot para car length
boxplot(numerical_df$carlength, horizontal = TRUE, col = "red", main = "Distribución de las longitudes d
automóviles")
```



Se observa un solo dato atípico antes del mínimo. Además, una distribución sesgada a la derecha.

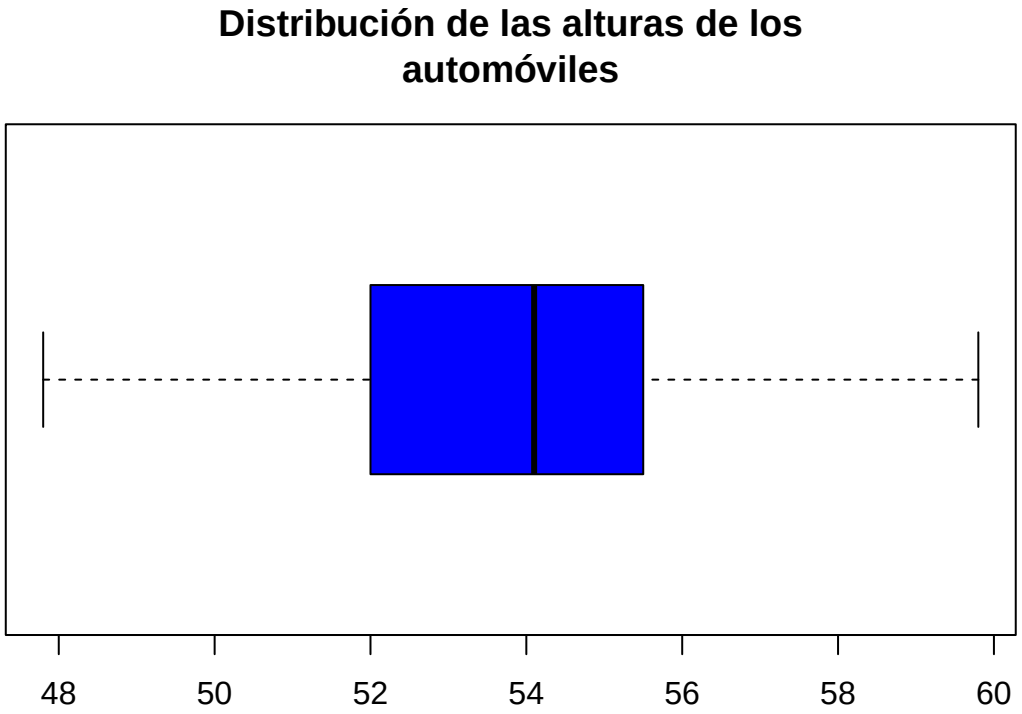
```
#Boxplot para car width
boxplot(numerical_df$carwidth, horizontal = TRUE, col = "yellow", main = "Distribución del ancho de los
automóviles")
```



Se observa una distribución normalizada, pero teniendo en cuenta que hay datos atípicos entre 71 y 73.

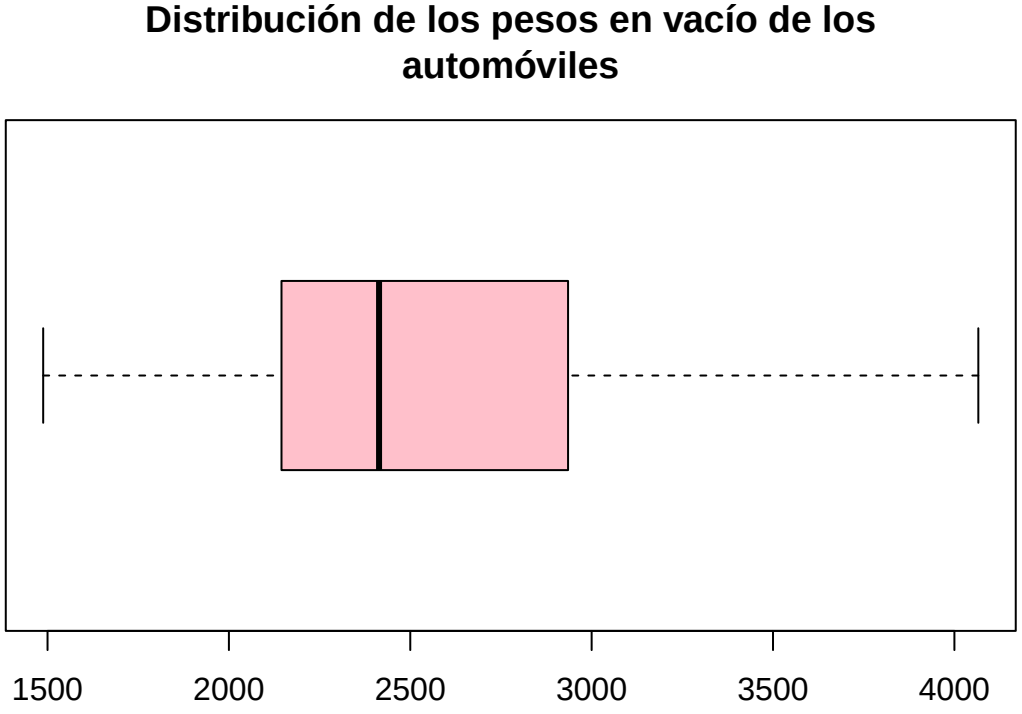
```
#Boxplot para car height
boxplot(numerical_df$carheight, horizontal = TRUE, col = "blue", main = "Distribución de las alturas de
```

```
automóviles")
```



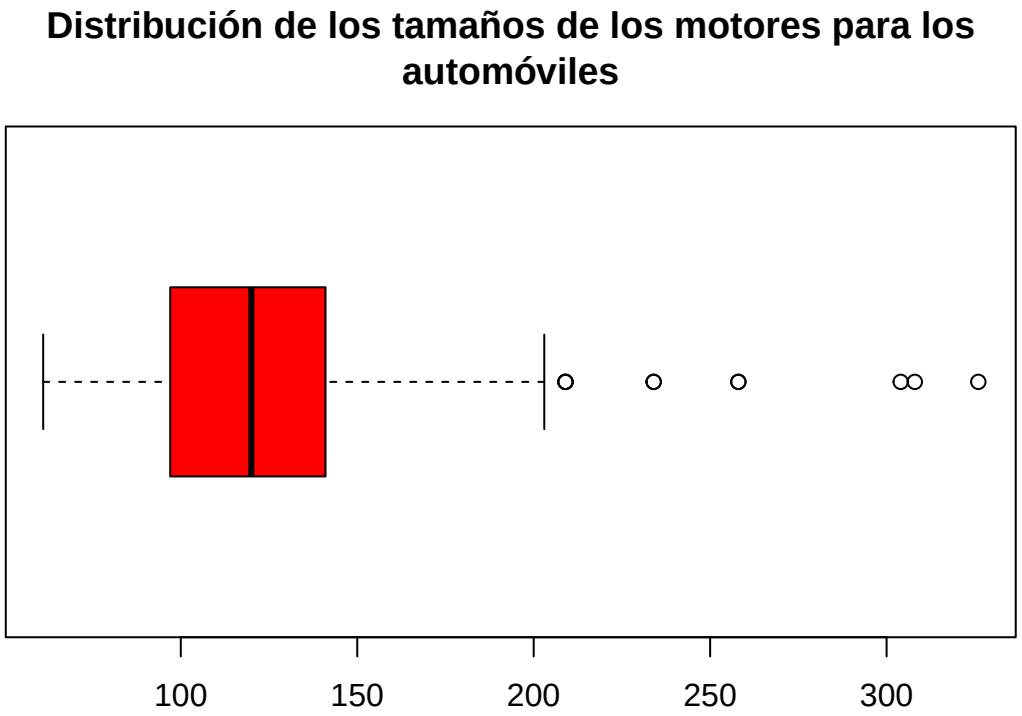
Se observa una distribución sesgada a la izquierda pero sin la presencia de datos atípicos.

```
#Boxplot para curb weight
boxplot(numerical_df$curbweight, horizontal = TRUE, col = "pink", main = "Distribución de los pesos en v
automóviles")
```



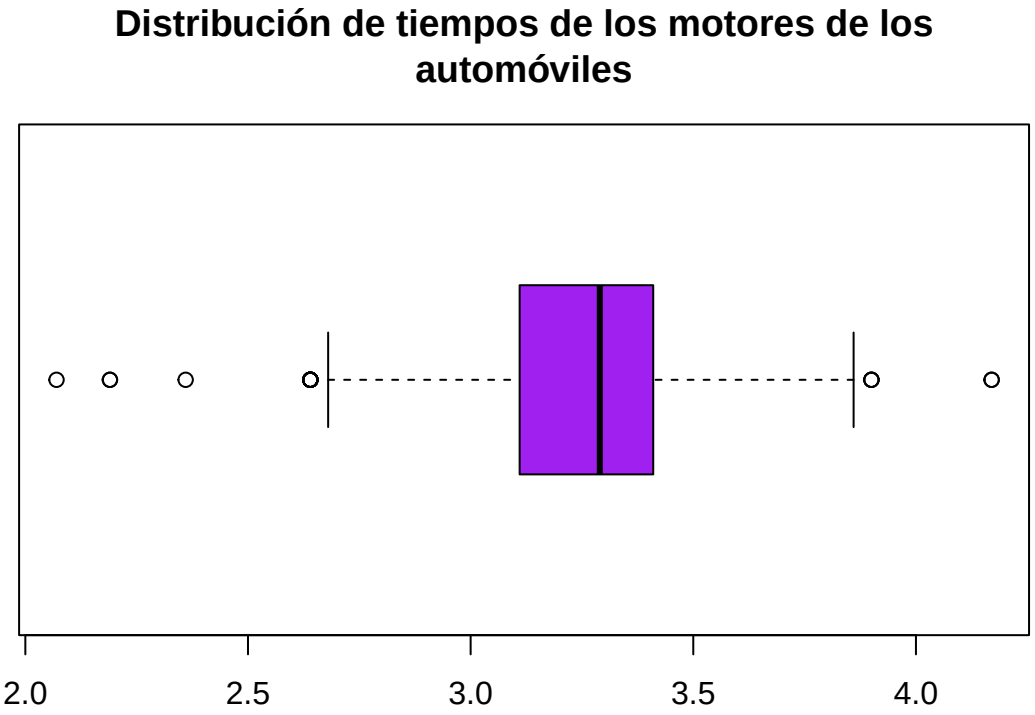
Se observa una distribución sesgada a la derecha pero sin la presencia de datos atípicos.

```r
#Boxplot para engine size
boxplot(numerical_df$enginesize, horizontal = TRUE, col = "red", main = "Distribución de los tamaños de
automóviles")
```



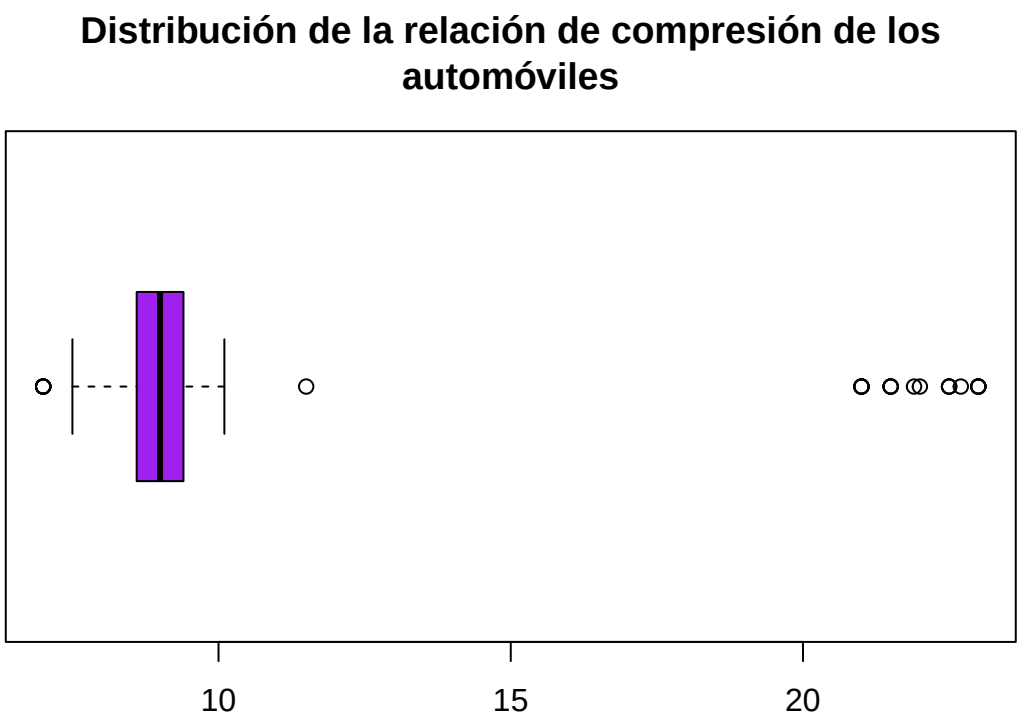
Se observa una gran cantidad de datos atípicos desde 225 aproximadamente, hasta más de 300. Sin embargo se ve una distribución normal.

```r
#Boxplot para stroke
boxplot(numerical_df$stroke, horizontal = TRUE, col = "purple", main = "Distribución de tiempos de los
automóviles")
```



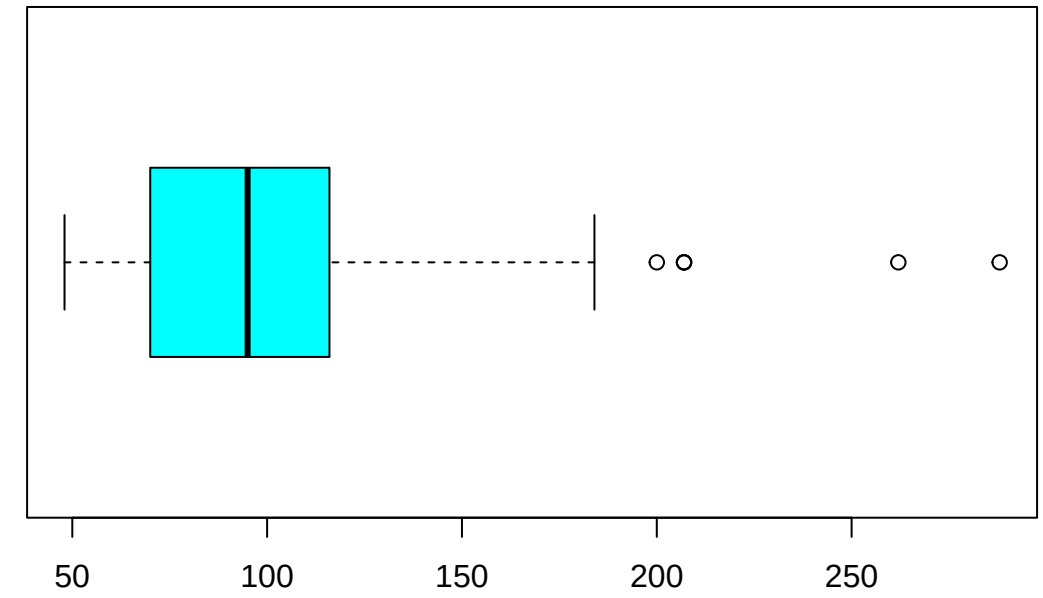
Se observa que hay valores atípicos entre 2 y 2.5, además de más de 4. Se observa una distribución asimétrica a la izquierda.

```
#Boxplot para compression ratio
boxplot(numerical_df$compressionratio, horizontal = TRUE, col = "purple", main = "Distribución de la rel
automóviles")
```

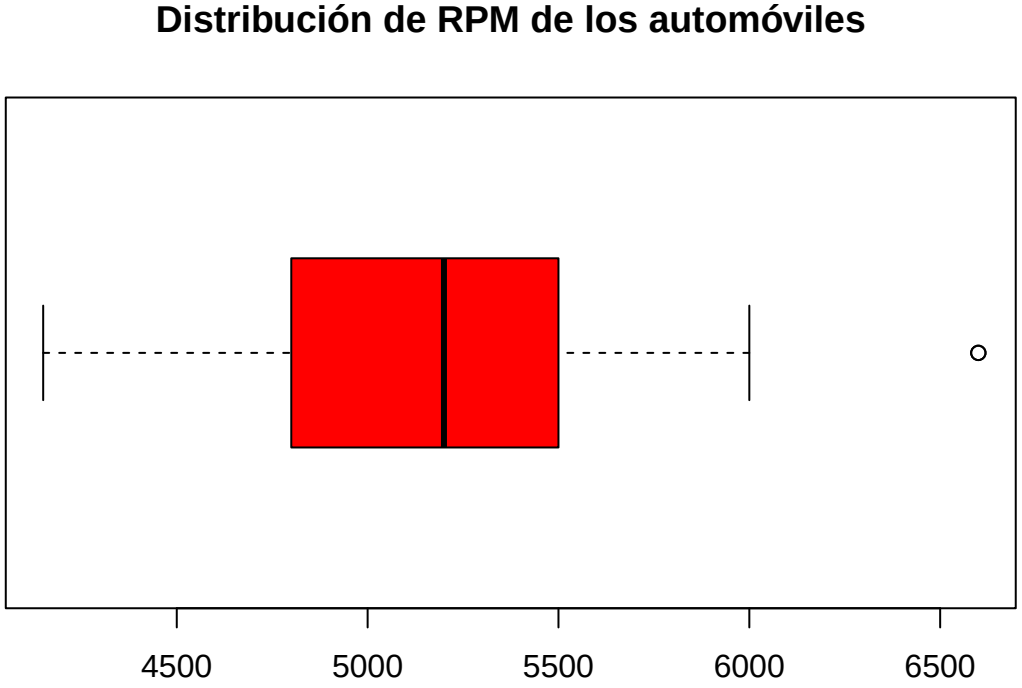


En este boxplot se observan valores atípicos mayores a 20. Por otra parte, hay una distribución normal ya que la media está centrada.

```r
#Boxplot para horse power
boxplot(numerical_df$horsepower, horizontal = TRUE, col = "cyan", main = "Distribución de los caballos d
```

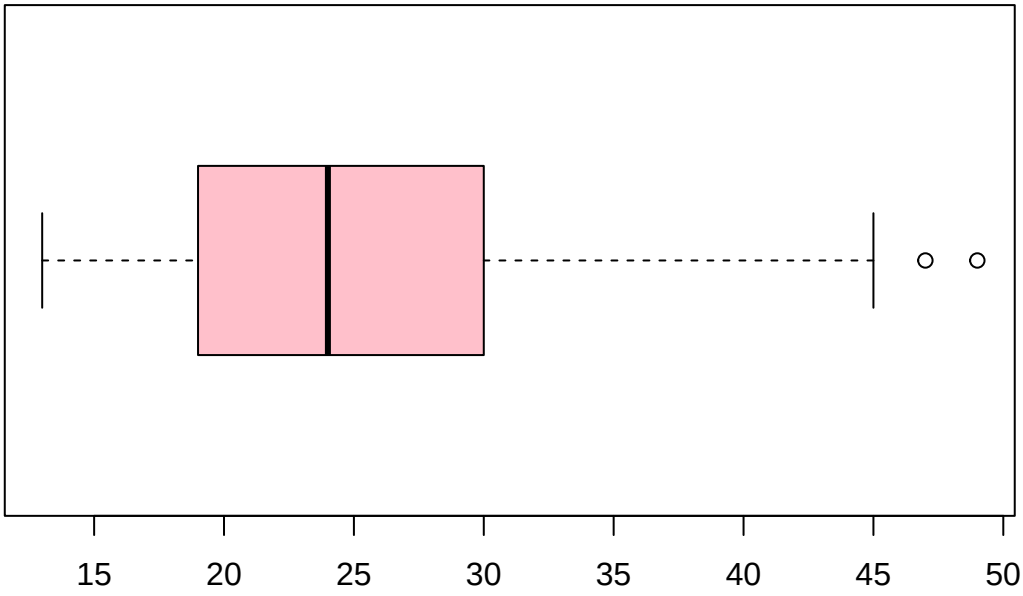



```r
#Boxplot para peak rpm
boxplot(numerical_df$peakrpm, horizontal = TRUE, col = "red", main = "Distribución de RPM de los automó
```



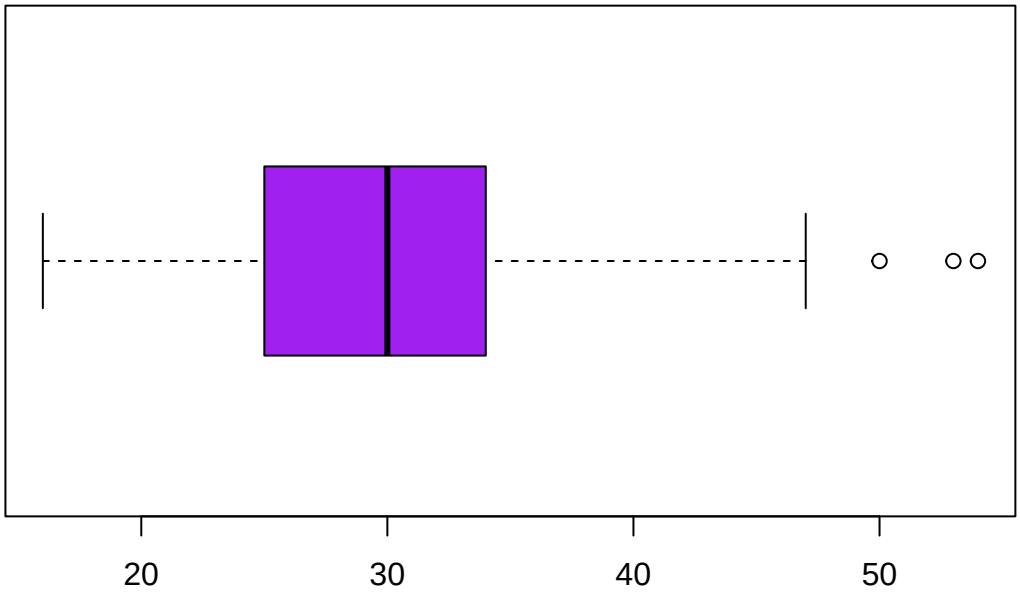

```r
#Boxplot para city mpg
boxplot(numerical_df$citympg, horizontal = TRUE, col = "pink", main = "Distribución de kilometraje en ci
```

Distribución de kilometraje en ciudad de los automóviles

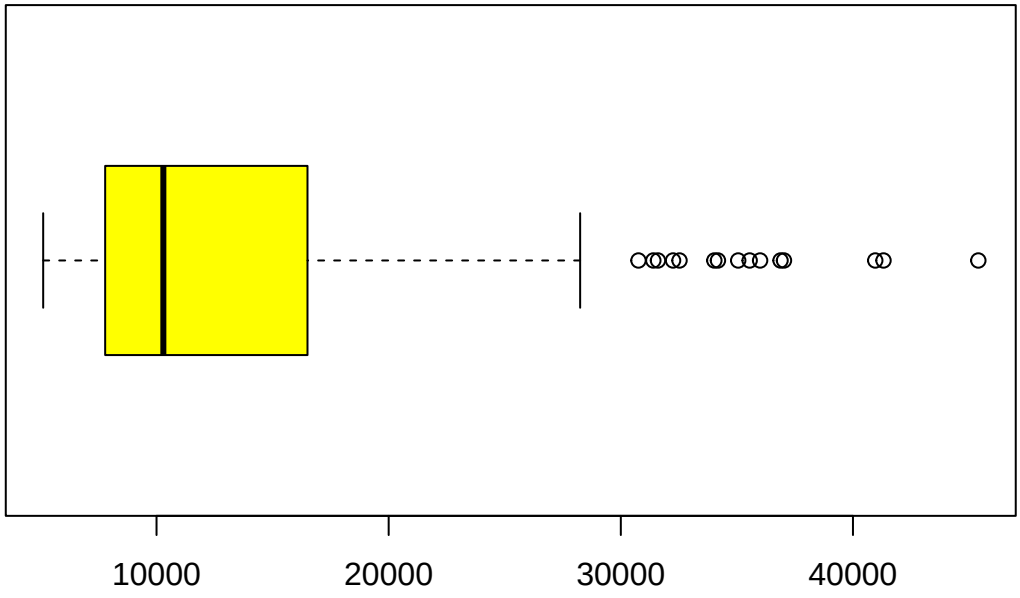



```
#Boxplot para highway mpg
boxplot(numerical_df$highwaympg, horizontal = TRUE, col = "purple", main = "Distribución de kilometraje
```

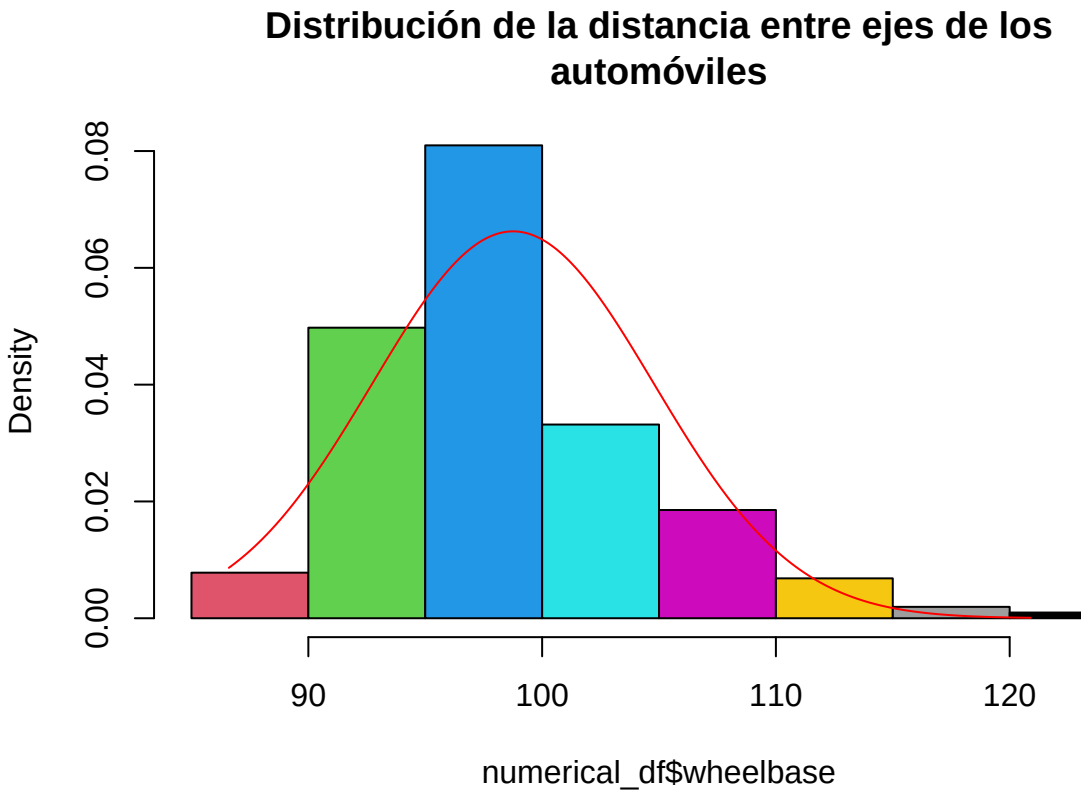
Distribución de kilometraje en autopista de los automóviles




```
#Boxplot para price
boxplot(numerical_df$price, horizontal = TRUE, col = "yellow", main = "Distribución de precios de los au
```



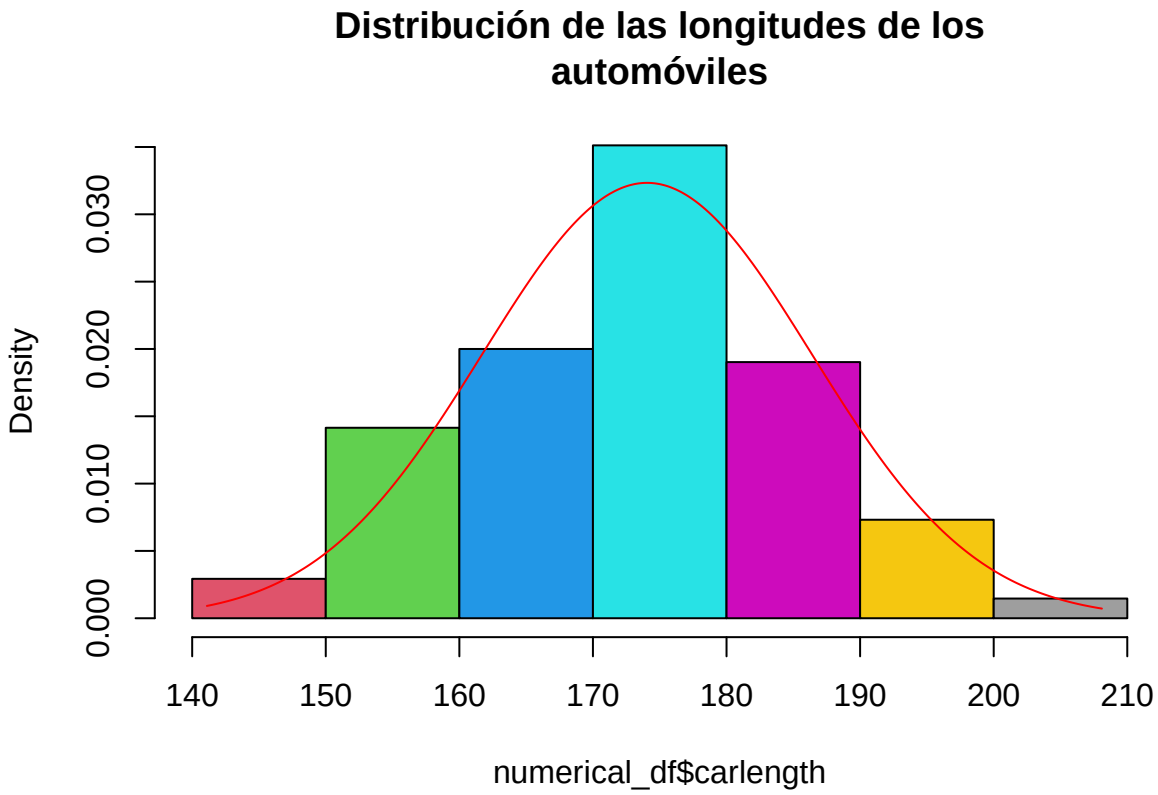

```
#Histograma para wheelbase
hist(numerical_df$wheelbase, prob = TRUE, col = 2:10, main = "Distribución de la distancia entre ejes de
automóviles")
x=seq(min(numerical_df$wheelbase),max(numerical_df$wheelbase),0.1)
y=dnorm(x,mean(numerical_df$wheelbase),sd(numerical_df$wheelbase))
lines(x,y,col="red")
```



Histogramas y densidades

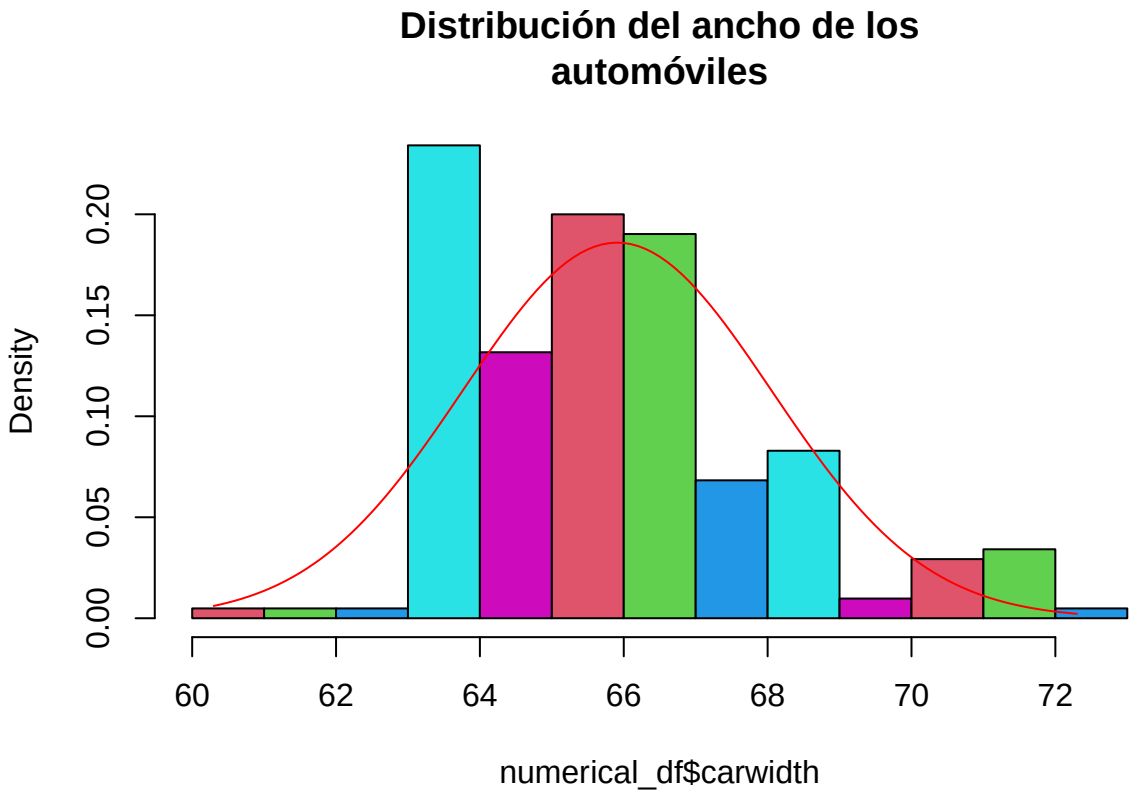
Se observa que hay una distribución relativamente normalizada ya que no hay una inclinación notoria en la curva.

```
#Histograma para car length
hist(numerical_df$carlength,  prob = TRUE, col = 2:10, main = "Distribución de las longitudes de los
automóviles")
x1=seq(min(numerical_df$carlength),max(numerical_df$carlength),0.1)
y1=dnorm(x1,mean(numerical_df$carlength),sd(numerical_df$carlength))
lines(x1,y1,col="red")
```



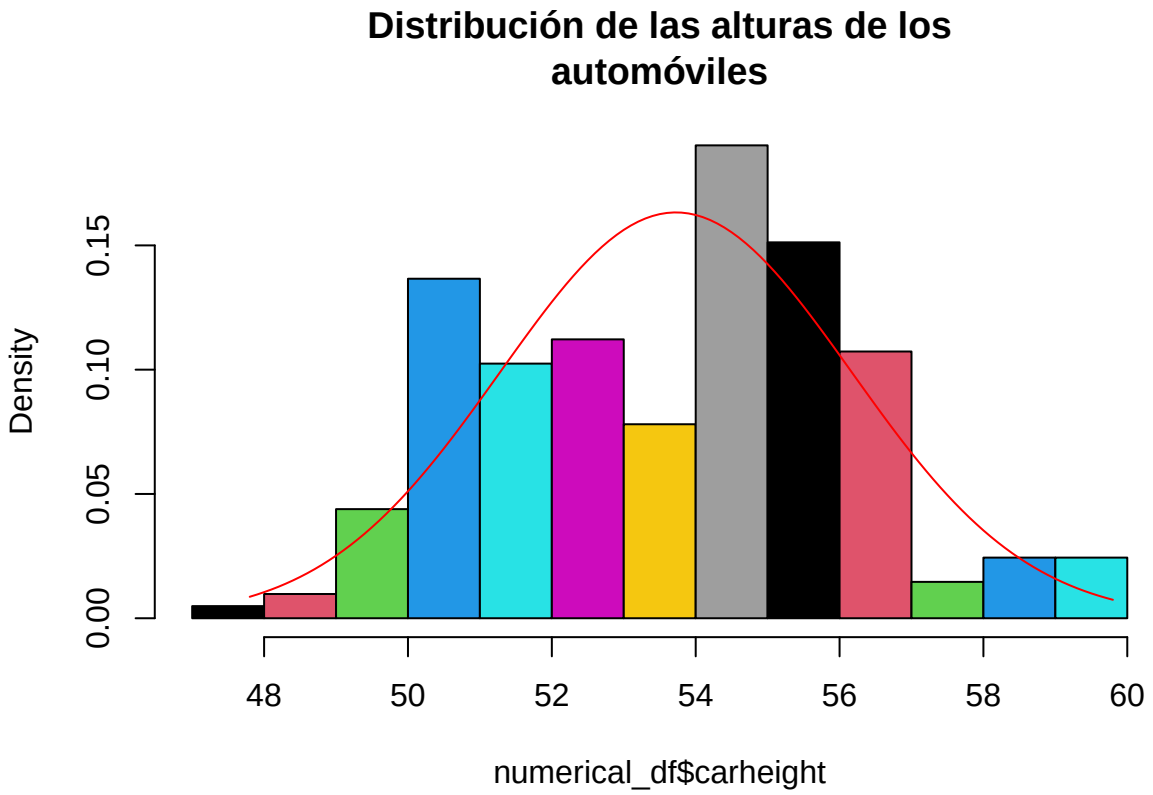
Se observa que hay una distribución relativamente normalizada ya que no hay una inclinación notoria en la curva.

```
#Histograma para car width
hist(numerical_df$carwidth, prob = TRUE, col = 2:6, main = "Distribución del ancho de los
automóviles")
x2=seq(min(numerical_df$carwidth),max(numerical_df$carwidth),0.1)
y2=dnorm(x2,mean(numerical_df$carwidth),sd(numerical_df$carwidth))
lines(x2,y2,col="red")
```



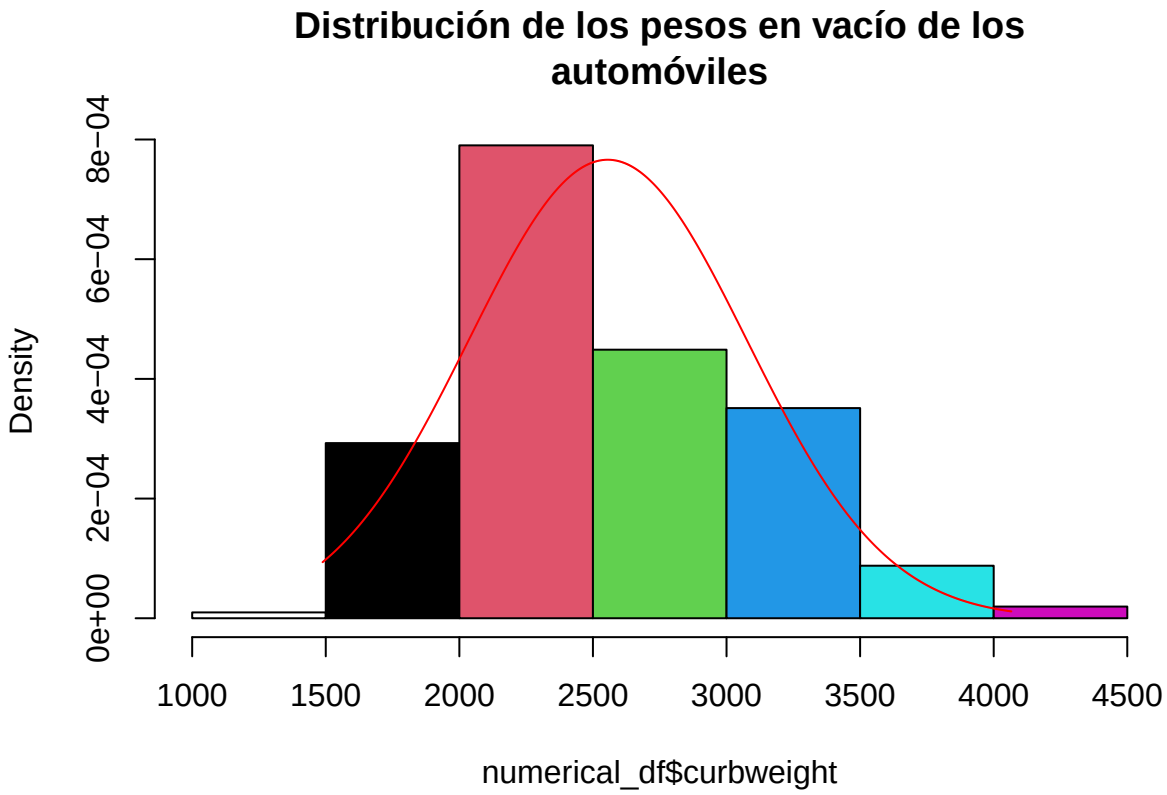
Se observa que hay una distribución relativamente normalizada ya que no hay una inclinación notoria en la curva.

```
#Histograma para car height
hist(numerical_df$carheight, prob = TRUE, col = 1:8, main = "Distribución de las alturas de los
automóviles")
x3=seq(min(numerical_df$carheight),max(numerical_df$carheight),0.1)
y3=dnorm(x3,mean(numerical_df$carheight),sd(numerical_df$carheight))
lines(x3,y3,col="red")
```



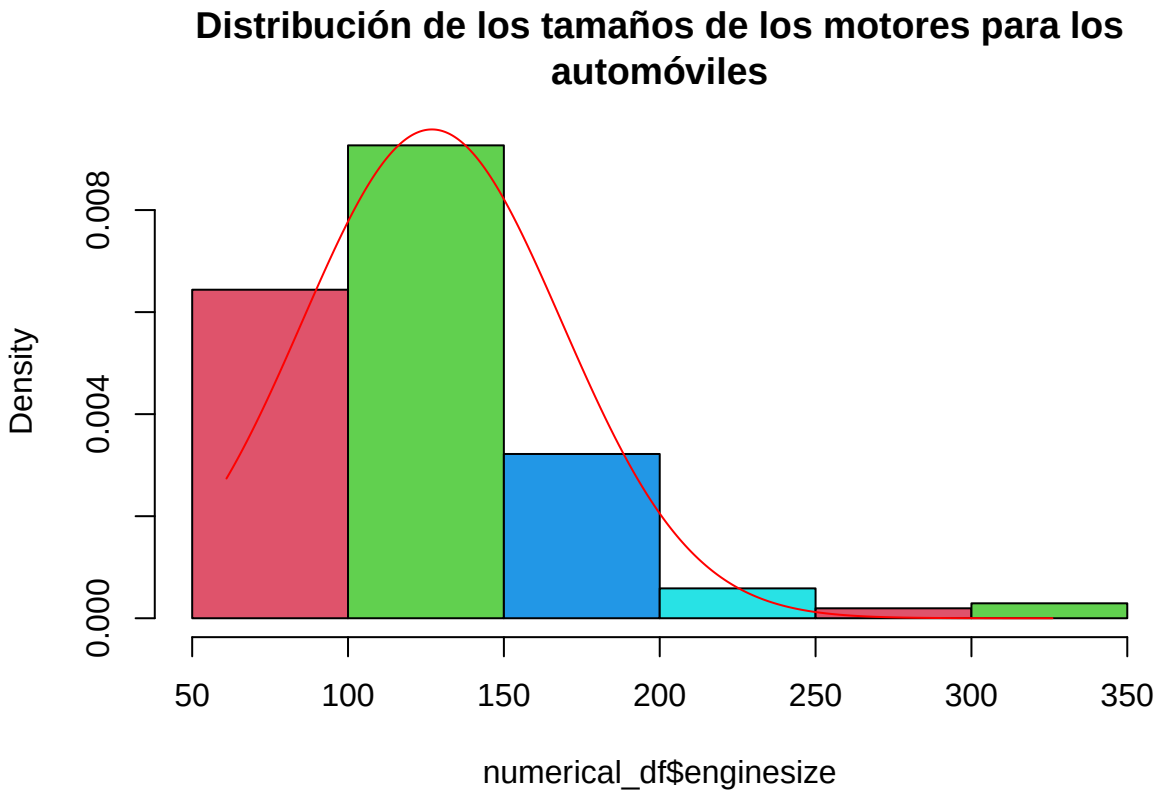
Se observa que hay una distribución relativamente normalizada ya que no hay una inclinación notoria en la curva.

```
#Histograma para curb weight
hist(numerical_df$curbweight, prob = TRUE, col = 0:8, main = "Distribución de los pesos en vacío de los
automóviles")
x4=seq(min(numerical_df$curbweight),max(numerical_df$curbweight),0.1)
y4=dnorm(x4,mean(numerical_df$curbweight),sd(numerical_df$curbweight))
lines(x4,y4,col="red")
```



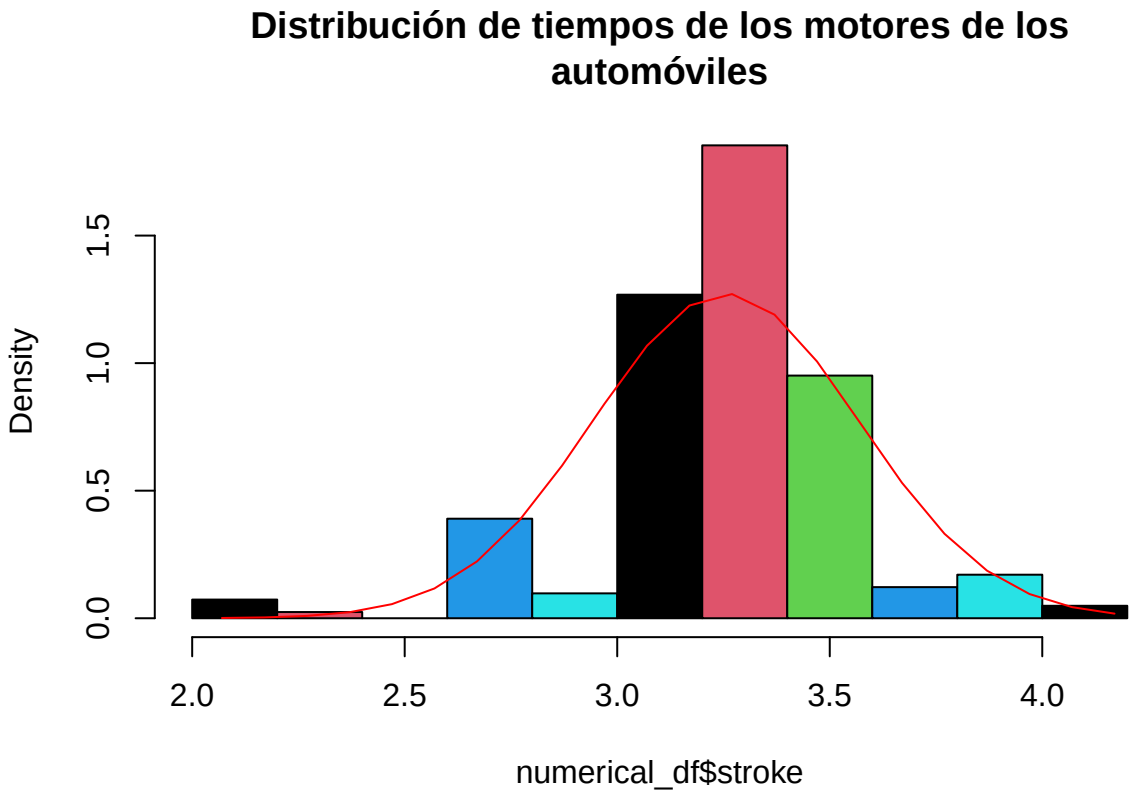
Se observa que hay una distribución relativamente normalizada ya que no hay una inclinación notoria en la curva.

```
#Histograma para engine size
hist(numerical_df$enginesize, prob = TRUE, col = 2:5, main = "Distribución de los tamaños de los motores
automóviles")
x5=seq(min(numerical_df$enginesize),max(numerical_df$enginesize),0.1)
y5=dnorm(x5,mean(numerical_df$enginesize),sd(numerical_df$enginesize))
lines(x5,y5,col="red")
```



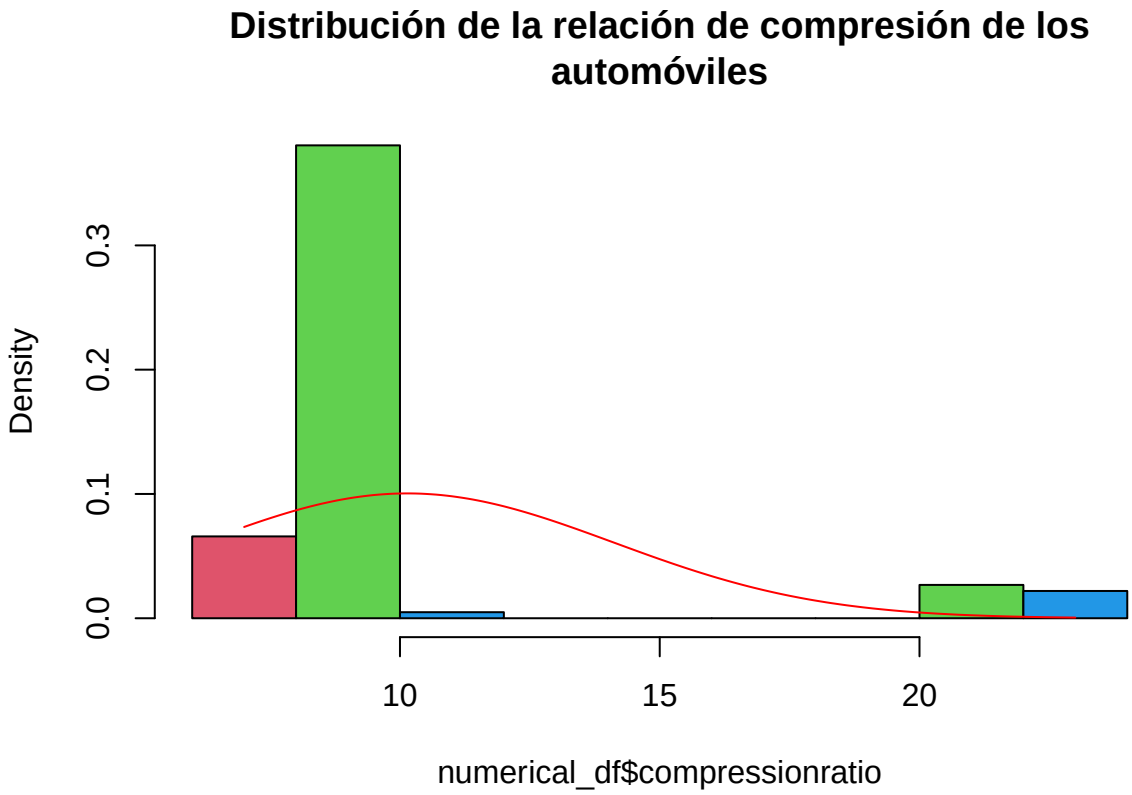
Aquí hay una distribución sesgada a la derecha ya que el mayor volumen se concentra a la izquierda de la gráfica.

```
#Histograma para stroke
hist(numerical_df$stroke, prob = TRUE, col = 1:5, main = "Distribución de tiempos de los motores de los
automóviles")
x6=seq(min(numerical_df$stroke),max(numerical_df$stroke),0.1)
y6=dnorm(x6,mean(numerical_df$stroke),sd(numerical_df$stroke))
lines(x6,y6,col="red")
```



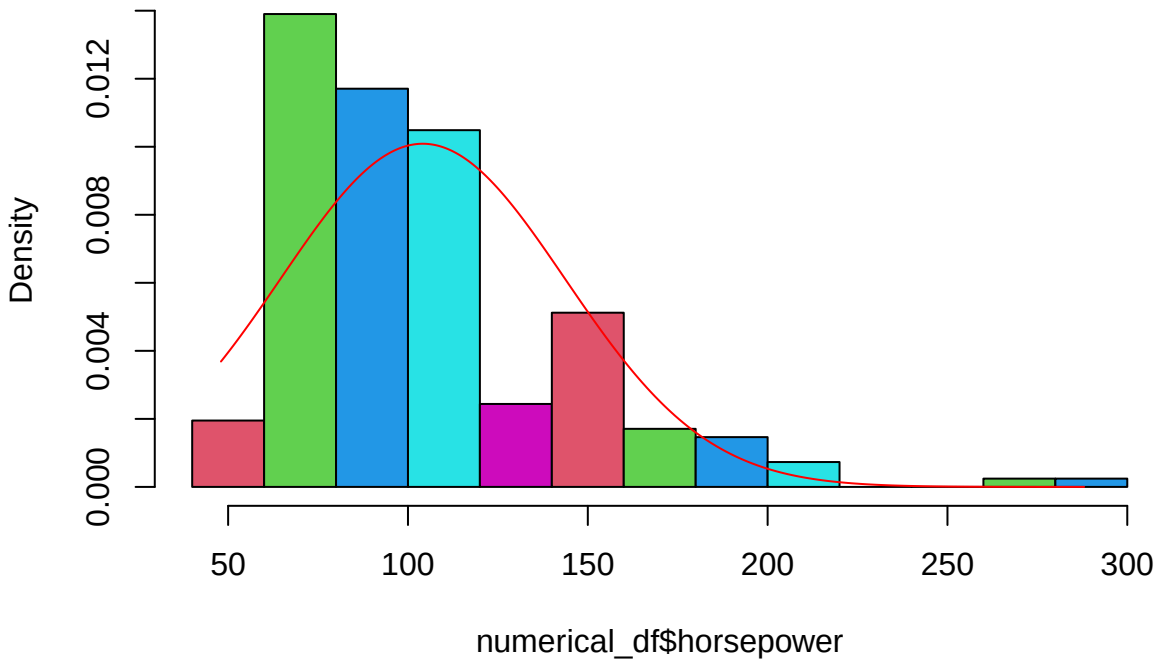
Por el contrario, en esta se observa una distribución sesgada a la izquierda ya que el volumen se concentra a la derecha de la gráfica.

```
#Histograma para compression ratio
hist(numerical_df$compressionratio, prob = TRUE, col = 10:15, main = "Distribución de la relación de co
automóviles")

x7=seq(min(numerical_df$compressionratio),max(numerical_df$compressionratio),0.1)
y7=dnorm(x7,mean(numerical_df$compressionratio),sd(numerical_df$compressionratio))
lines(x7,y7,col="red")
```



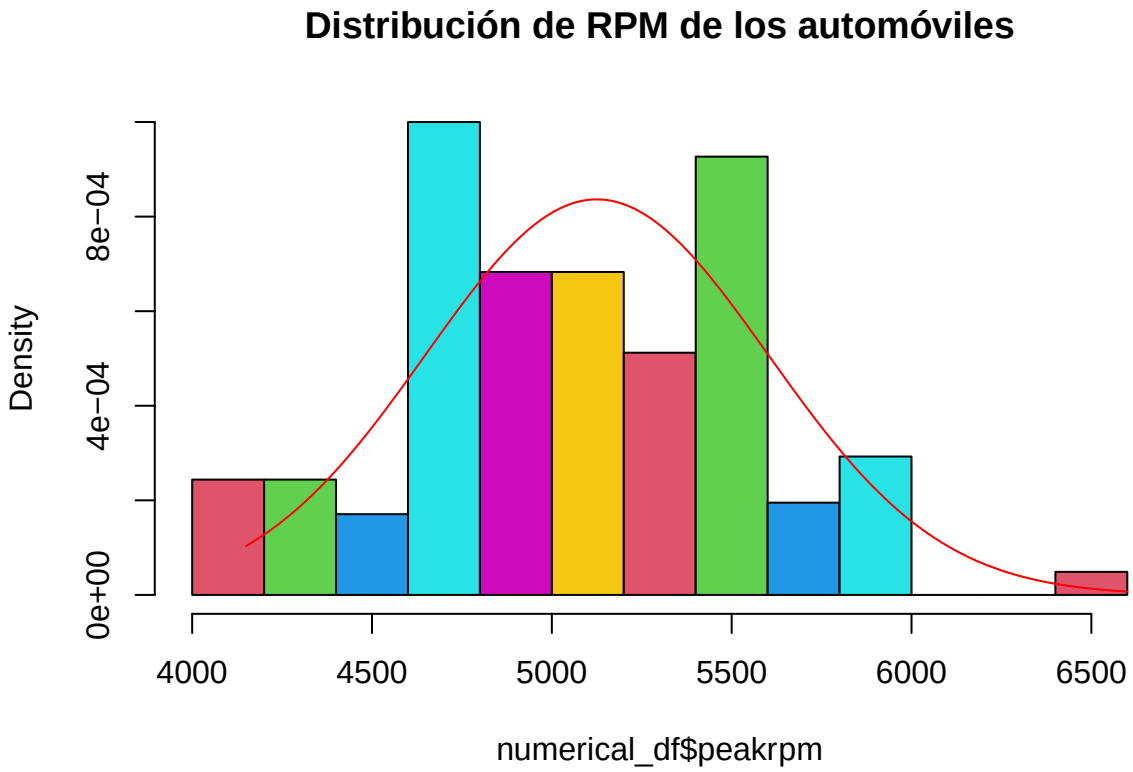
En esta gráfica de relación de compresión de los automóviles está sesgada a la derecha pero se observa una gran cantidad de datos atípicos que pueden afectar el análisis.

```
#Histograma para horse power
hist(numerical_df$horsepower, prob = TRUE, col = 10:14, main = "Distribución de los caballos de potencia

x8=seq(min(numerical_df$horsepower),max(numerical_df$horsepower),0.1)
y8=dnorm(x8,mean(numerical_df$horsepower),sd(numerical_df$horsepower))
lines(x8,y8,col="red")
```



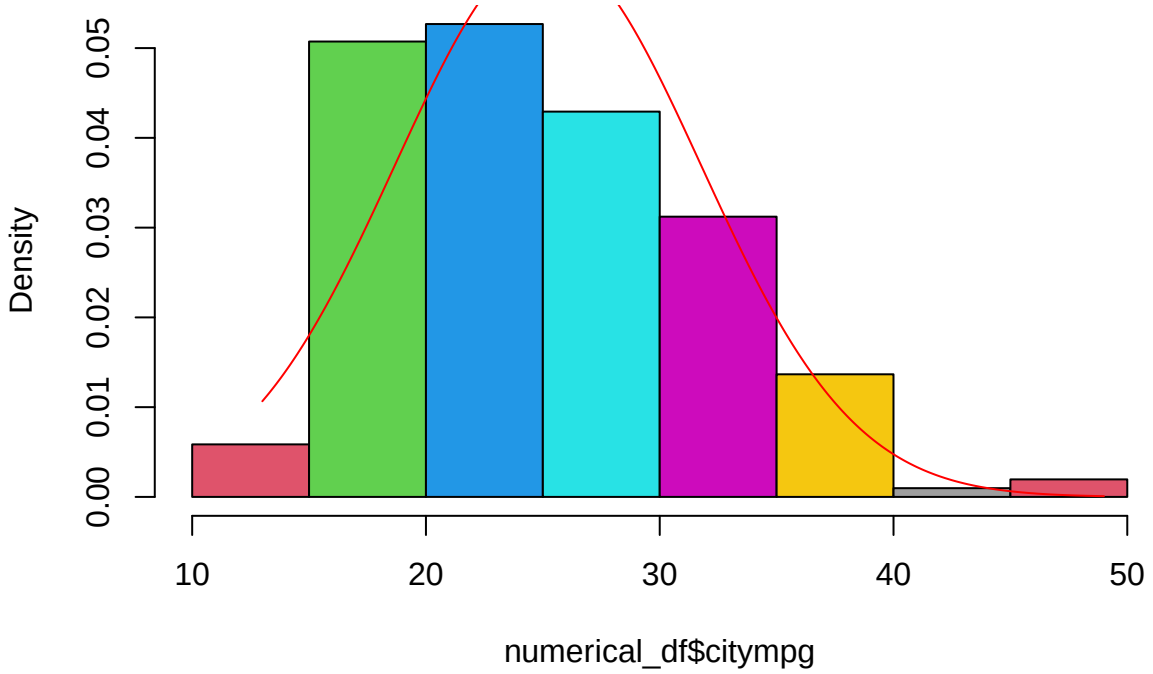
En esta gráfica de los caballos de potencia se observan datos atípicos entre 250 y 300 que generan ruido al análisis, pero se sabe que está sesgado a la derecha.

```
#Histograma para peak rpm
hist(numerical_df$peakrpm, prob = TRUE, col = 10:15, main = "Distribución de RPM de los automóviles")
x9=seq(min(numerical_df$peakrpm),max(numerical_df$peakrpm),0.1)
y9=dnorm(x9,mean(numerical_df$peakrpm),sd(numerical_df$peakrpm))
lines(x9,y9,col="red")
```



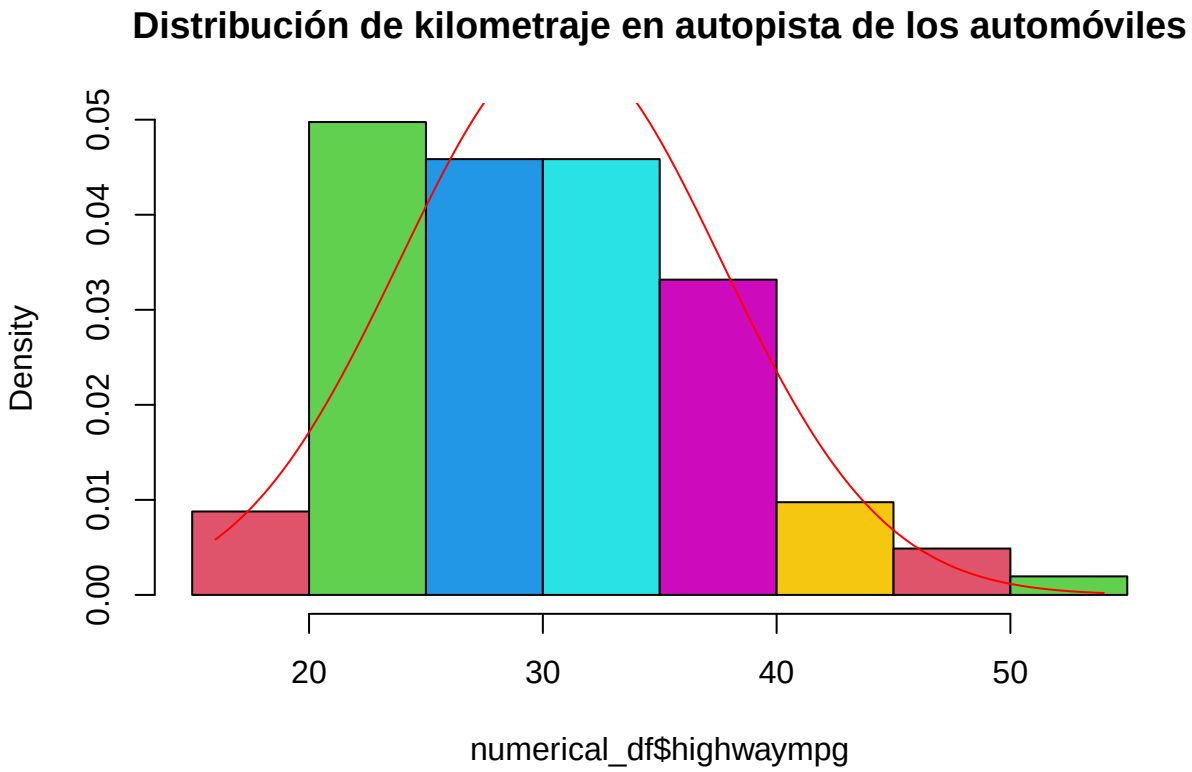
Hay pocos datos atípicos, pero se observa que la gráfica para los RPM está normalizada.

```
#Histograma para city mpg

hist(numerical_df$citympg, prob = TRUE, col = 2:8, main = "Distribución de kilometraje en ciudad de los
x10=seq(min(numerical_df$citympg),max(numerical_df$citympg),0.1)
y10=dnorm(x10,mean(numerical_df$citympg),sd(numerical_df$citympg))
lines(x10,y10,col="red")
```



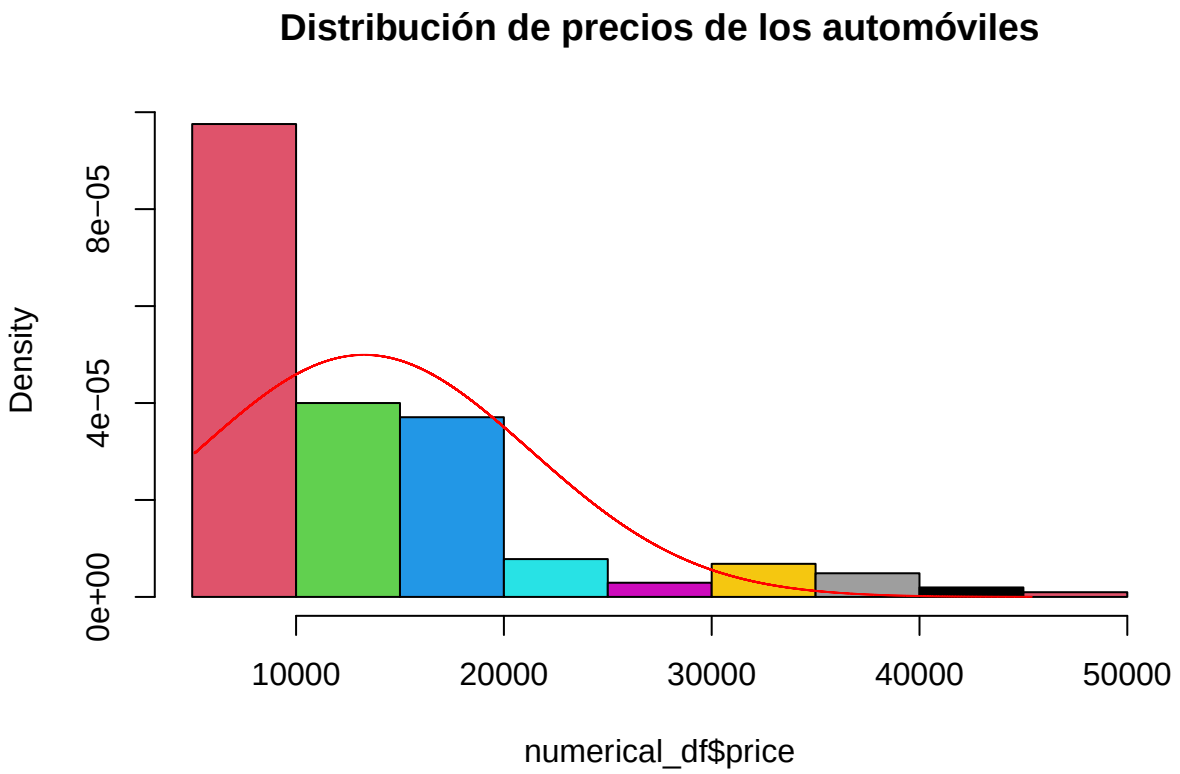
Se observa que está normalizada pero un poco influenciada por los datos atípicos.

```
#Histograma para highway mpg
hist(numerical_df$highwaympg, prob = TRUE, col = 10:15, main = "Distribución de kilometraje en autopista
x11=seq(min(numerical_df$highwaympg),max(numerical_df$highwaympg),0.1)
y11=dnorm(x11,mean(numerical_df$highwaympg),sd(numerical_df$highwaympg))
lines(x11,y11,col="red")
```



Para los MPG del highway se observa normalizada.

```
#Histograma para price
hist(numerical_df$price, prob = TRUE, col = 2:15, main = "Distribución de precios de los automóviles")
x12=seq(min(numerical_df$price),max(numerical_df$price),0.1)
y12=dnorm(x12,mean(numerical_df$price),sd(numerical_df$price))
lines(x12,y12,col="red")
```



El precio se observa que su gráfica está sesgada a la derecha con varios valores atípicos.

Coeficiente de correlación El coeficiente de correlación puede ayudar a seleccionar las variables importantes para el análisis de las características de los automóviles que determinan su precio. Por lo tanto se evaluará la correlación de cada variable con respecto al precio en este inciso.

```
price <- numerical_df$price
wheelbase <- numerical_df$wheelbase
r= cor(price,wheelbase)
cat("El coeficiente de correlación entre el precio de los automóviles y la distancia entre ejes de los a
    r =",r)
```

```
## El coeficiente de correlación entre el precio de los automóviles y la distancia entre ejes de los au
##     r = 0.5778156
```

```
price <- numerical_df$price
carlength <- numerical_df$carlength
r1= cor(price,carlength)
cat("El coeficiente de correlación entre el precio de los automóviles y la longitud de los automóviles e
    r =",r1)
```

```
## El coeficiente de correlación entre el precio de los automóviles y la longitud de los automóviles es
##     r = 0.68292
```

```
price <- numerical_df$price
carwidth <- numerical_df$carwidth
r2= cor(price,carwidth)
cat("El coeficiente de correlación entre el precio de los automóviles y el ancho de los automóviles esta
    r =",r2)
```

```
## El coeficiente de correlación entre el precio de los automóviles y el ancho de los automóviles está e
##     r = 0.7593253
```

```r
price <- numerical_df$price
carheight <- numerical_df$carheight
r3= cor(price,carheight)
cat("El coeficiente de correlación entre el precio de los automóviles y la altura de los automóviles es
    r =",r3)
```

```
## El coeficiente de correlación entre el precio de los automóviles y la altura de los automóviles está
##     r = 0.1193362
```

```r
price <- numerical_df$price
curbweight <- numerical_df$curbweight
r4= cor(price,curbweight)
cat("El coeficiente de correlación entre el precio de los automóviles y los pesos en vacío de los autom
    r =",r4)
```

```
## El coeficiente de correlación entre el precio de los automóviles y los pesos en vacío de los automóv
##     r = 0.8353049
```

```r
price <- numerical_df$price
enginesize <- numerical_df$enginesize
r5= cor(price,enginesize)
cat("El coeficiente de correlación entre el precio de los automóviles y el tamaño de los motores de los
    r =",r5)
```

```
## El coeficiente de correlación entre el precio de los automóviles y el tamaño de los motores de los au
##     r = 0.8741448
```

```r
price <- numerical_df$price
stroke <- numerical_df$stroke
r6= cor(price,stroke)
cat("El coeficiente de correlación entre el precio de los automóviles y tiempos de los motores de los au
    r =",r6)
```

```
## El coeficiente de correlación entre el precio de los automóviles y tiempos de los motores de los auto
##     r = 0.07944308
```

```r
price <- numerical_df$price
compressionratio <- numerical_df$compressionratio
r7= cor(price,compressionratio)
cat("El coeficiente de correlación entre el precio de los automóviles y la relación de compresión de los
    r =",r7)
```

```
## El coeficiente de correlación entre el precio de los automóviles y la relación de compresión de los a
##     r = 0.06798351
```

```r
price <- numerical_df$price
horsepower <- numerical_df$horsepower
r8= cor(price,horsepower)
cat("El coeficiente de correlación entre el precio de los automóviles y los caballos de potencia de los
    r =",r8)
```

```
## El coeficiente de correlación entre el precio de los automóviles y los caballos de potencia de los au
##     r = 0.8081388
```

```r
price <- numerical_df$price
peakrpm <- numerical_df$peakrpm
r9= cor(price,peakrpm)
cat("El coeficiente de correlación entre el precio de los automóviles y las RPM de los automóviles está
```

```
    r =",r9)
```

```
## El coeficiente de correlación entre el precio de los automóviles y las RPM de los automóviles está da
##     r = -0.08526715
```

```
price <- numerical_df$price
citympg <- numerical_df$citympg
r10= cor(price,citympg)
cat("El coeficiente de correlación entre el precio de los automóviles y los kilometrajes en ciudad de lo
    r =",r10)
```

```
## El coeficiente de correlación entre el precio de los automóviles y los kilometrajes en ciudad de los
##     r = -0.6857513
```

```
price <- numerical_df$price
highwaympg <- numerical_df$highwaympg
r11= cor(price,highwaympg)
cat("El coeficiente de correlación entre el precio de los automóviles y los kilometrajes en carretera de
    r =",r11)
```

```
## El coeficiente de correlación entre el precio de los automóviles y los kilometrajes en carretera de l
##     r = -0.6975991
```

```
#Coeficientes de correlación entre todas las combinaciones posibles de variables numericas

data <- numerical_df

# Calcular la matriz de correlación
cor_matrix <- cor(data[, c("wheelbase", "carlength", "carwidth", "carheight", "curbweight", "enginesize

print(cor_matrix)
```

```
##                   wheelbase  carlength   carwidth   carheight curbweight
## wheelbase         1.0000000  0.8745875  0.7951436  0.58943476  0.7763863
## carlength         0.8745875  1.0000000  0.8411183  0.49102946  0.8777285
## carwidth          0.7951436  0.8411183  1.0000000  0.27921032  0.8670325
## carheight         0.5894348  0.4910295  0.2792103  1.00000000  0.2955717
## curbweight        0.7763863  0.8777285  0.8670325  0.29557173  1.0000000
## enginesize        0.5693287  0.6833599  0.7354334  0.06714874  0.8505941
## stroke            0.1609590  0.1295326  0.1829417 -0.05530667  0.1687900
## compressionratio  0.2497858  0.1584137  0.1811286  0.26121423  0.1513617
## horsepower        0.3532945  0.5526230  0.6407321 -0.10880206  0.7507393
## peakrpm          -0.3604687 -0.2872422 -0.2200123 -0.32041072 -0.2662432
## citympg          -0.4704136 -0.6709087 -0.6427043 -0.04863963 -0.7574138
## highwaympg       -0.5440819 -0.7046616 -0.6772179 -0.10735763 -0.7974648
## price             0.5778156  0.6829200  0.7593253  0.11933623  0.8353049
##                   enginesize      stroke compressionratio  horsepower
## wheelbase         0.56932868  0.16095905       0.24978585  0.35329448
## carlength         0.68335987  0.12953261       0.15841371  0.55262297
## carwidth          0.73543340  0.18294169       0.18112863  0.64073208
## carheight         0.06714874 -0.05530667       0.26121423 -0.10880206
## curbweight        0.85059407  0.16879004       0.15136174  0.75073925
## enginesize        1.00000000  0.20312859       0.02897136  0.80976865
## stroke            0.20312859  1.00000000       0.18611011  0.08093954
## compressionratio  0.02897136  0.18611011       1.00000000 -0.20432623
```

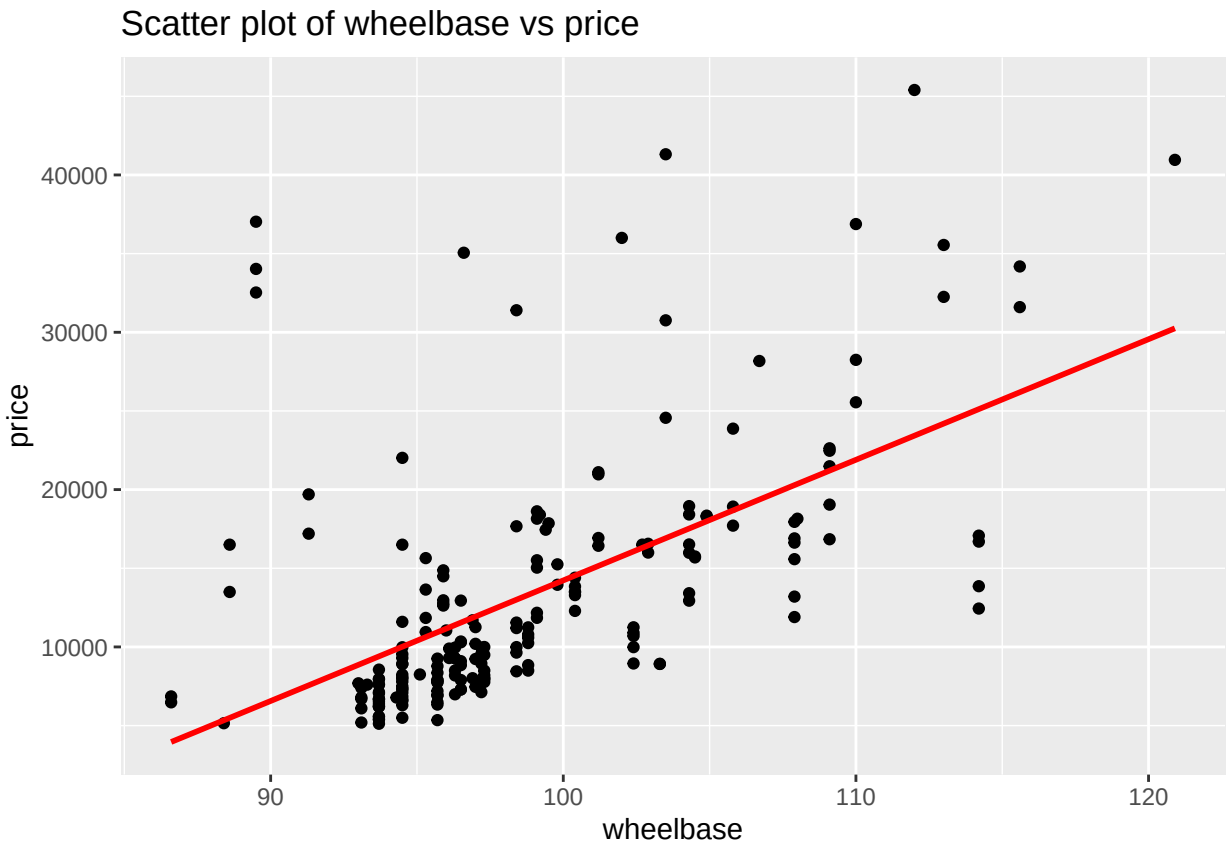
```
## horsepower        0.80976865  0.08093954      -0.20432623  1.00000000
## peakrpm          -0.24465983 -0.06796375      -0.43574051  0.13107251
## citympg          -0.65365792 -0.04214475       0.32470142 -0.80145618
## highwaympg       -0.67746991 -0.04393093       0.26520139 -0.77054389
## price             0.87414480  0.07944308       0.06798351  0.80813882
##                      peakrpm     citympg  highwaympg       price
## wheelbase        -0.36046875 -0.47041361 -0.54408192  0.57781560
## carlength        -0.28724220 -0.67090866 -0.70466160  0.68292002
## carwidth         -0.22001230 -0.64270434 -0.67721792  0.75932530
## carheight        -0.32041072 -0.04863963 -0.10735763  0.11933623
## curbweight       -0.26624318 -0.75741378 -0.79746479  0.83530488
## enginesize       -0.24465983 -0.65365792 -0.67746991  0.87414480
## stroke           -0.06796375 -0.04214475 -0.04393093  0.07944308
## compressionratio -0.43574051  0.32470142  0.26520139  0.06798351
## horsepower        0.13107251 -0.80145618 -0.77054389  0.80813882
## peakrpm           1.00000000 -0.11354438 -0.05427481 -0.08526715
## citympg          -0.11354438  1.00000000  0.97133704 -0.68575134
## highwaympg       -0.05427481  0.97133704  1.00000000 -0.69759909
## price            -0.08526715 -0.68575134 -0.69759909  1.00000000
```

Se puede observar que price tiene una correlación positiva y mayor que cero con: curbweight, horsepower, carwidth, enginesize, lo que quiere decir que cuando estas variables aumentan, entonces muy probablemente el precio aumentará. Sin embargo, tiene una correlación negativa con: city mpg, y highwaympg, lo que quiere decir que cuando estas aumenten (en kilometraje), el precio del vehículo tenderá a disminuir.

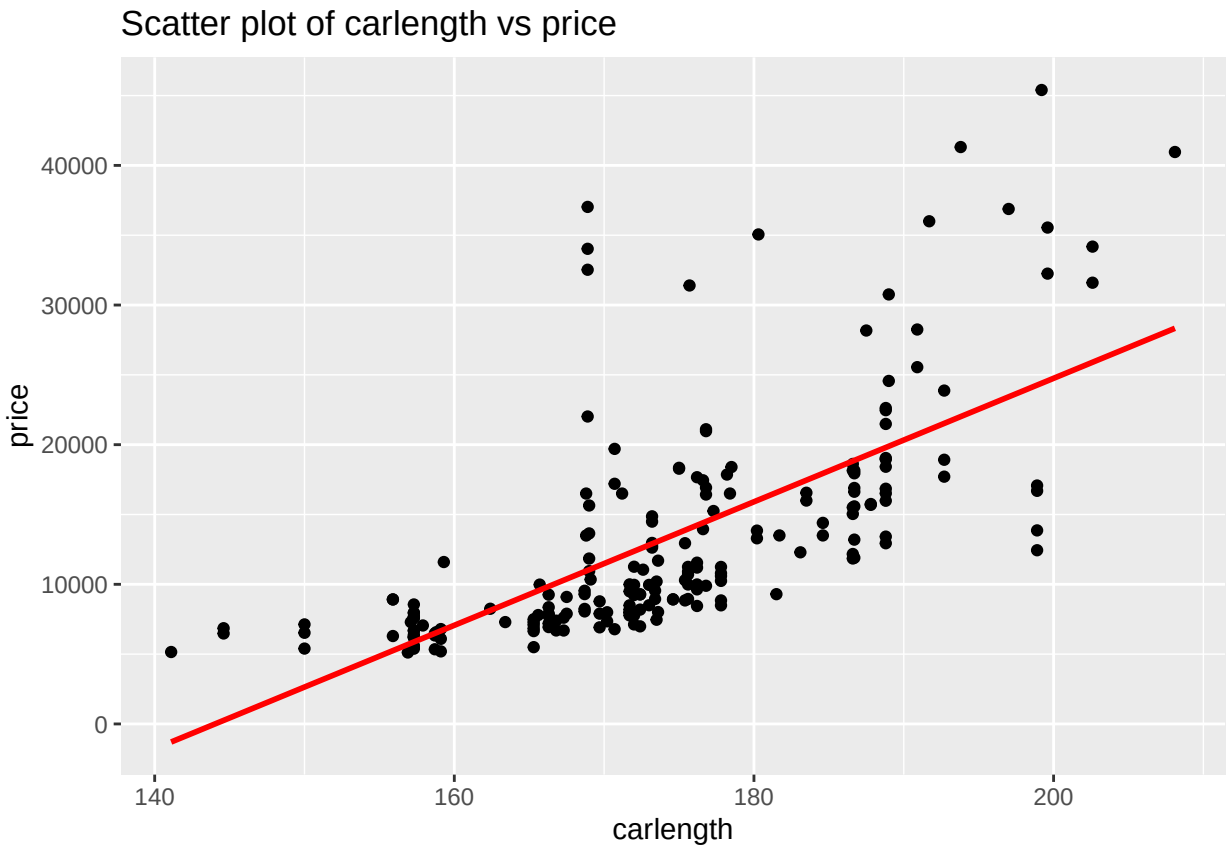
```
#nuevo data frame
data <- numerical_df
#Variables
y_variable <- "price"
x_variables <- colnames(data)[colnames(data) != y_variable]

# Diagramas de dispersion
for (x_var in x_variables) {
  p <- ggplot(data, aes(x = !!sym(x_var), y = !!sym(y_variable))) +
    geom_point() +
    geom_smooth(method = "lm", col = "red", se = FALSE) +  # Agregar línea de regresión
    labs(title = paste("Scatter plot of", x_var, "vs", y_variable),
         x = x_var,
         y = y_variable)

  print(p)
}
```

Diagrama de dispersión

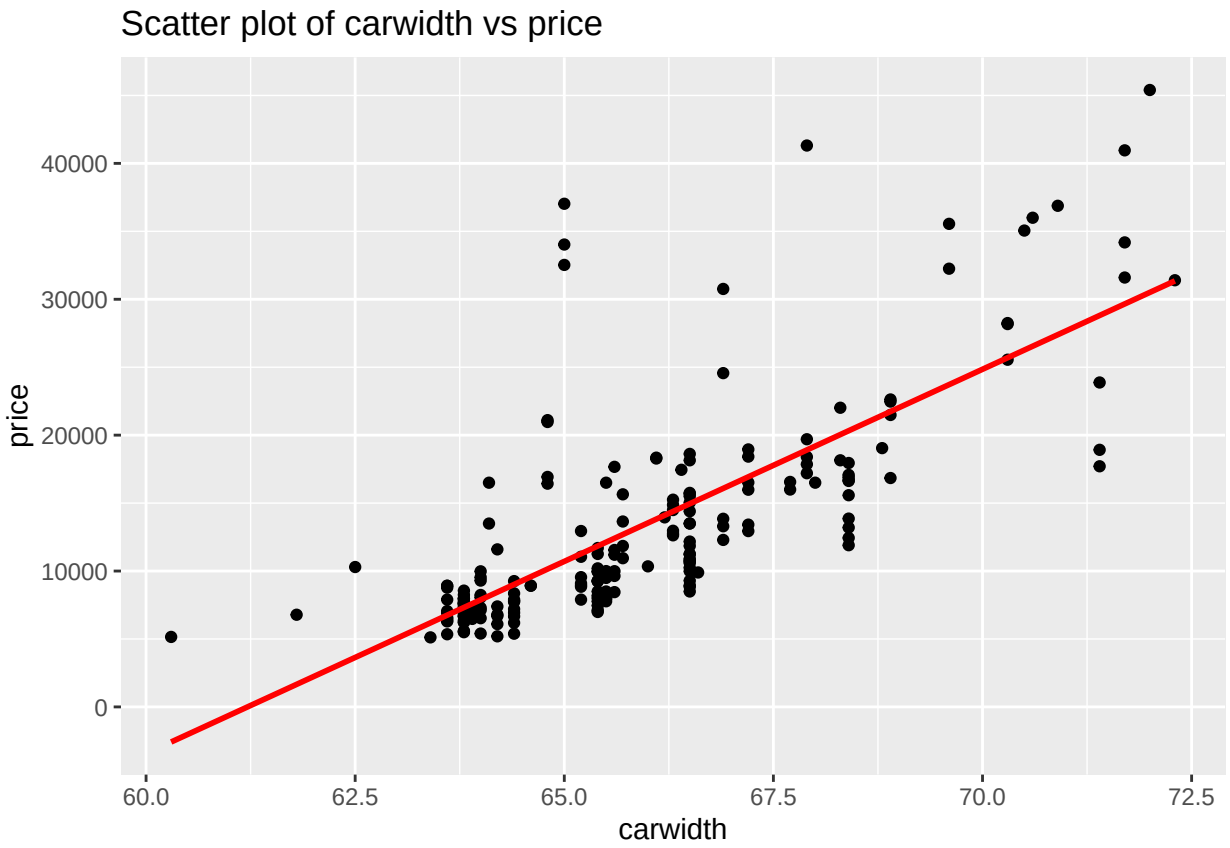
```
## `geom_smooth()` using formula = 'y ~ x'
```



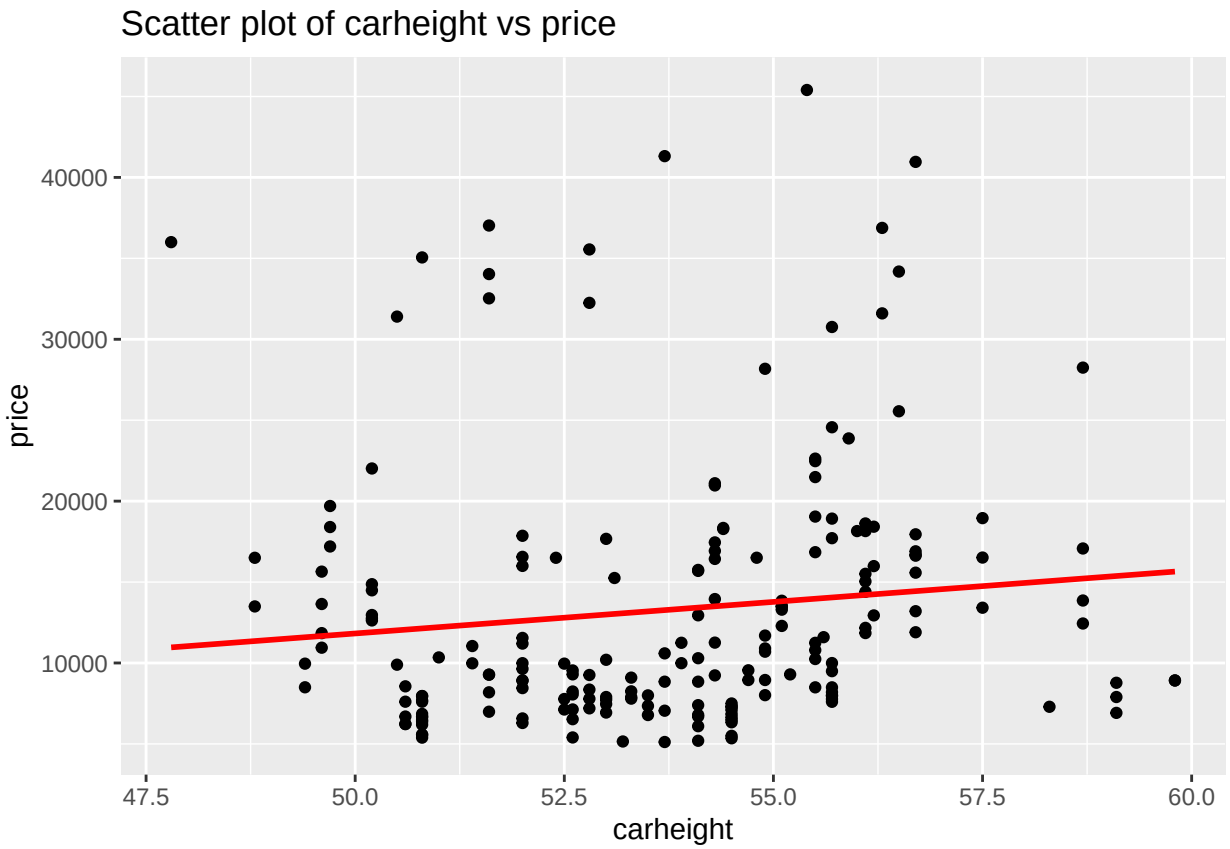

```
## `geom_smooth()` using formula = 'y ~ x'
```



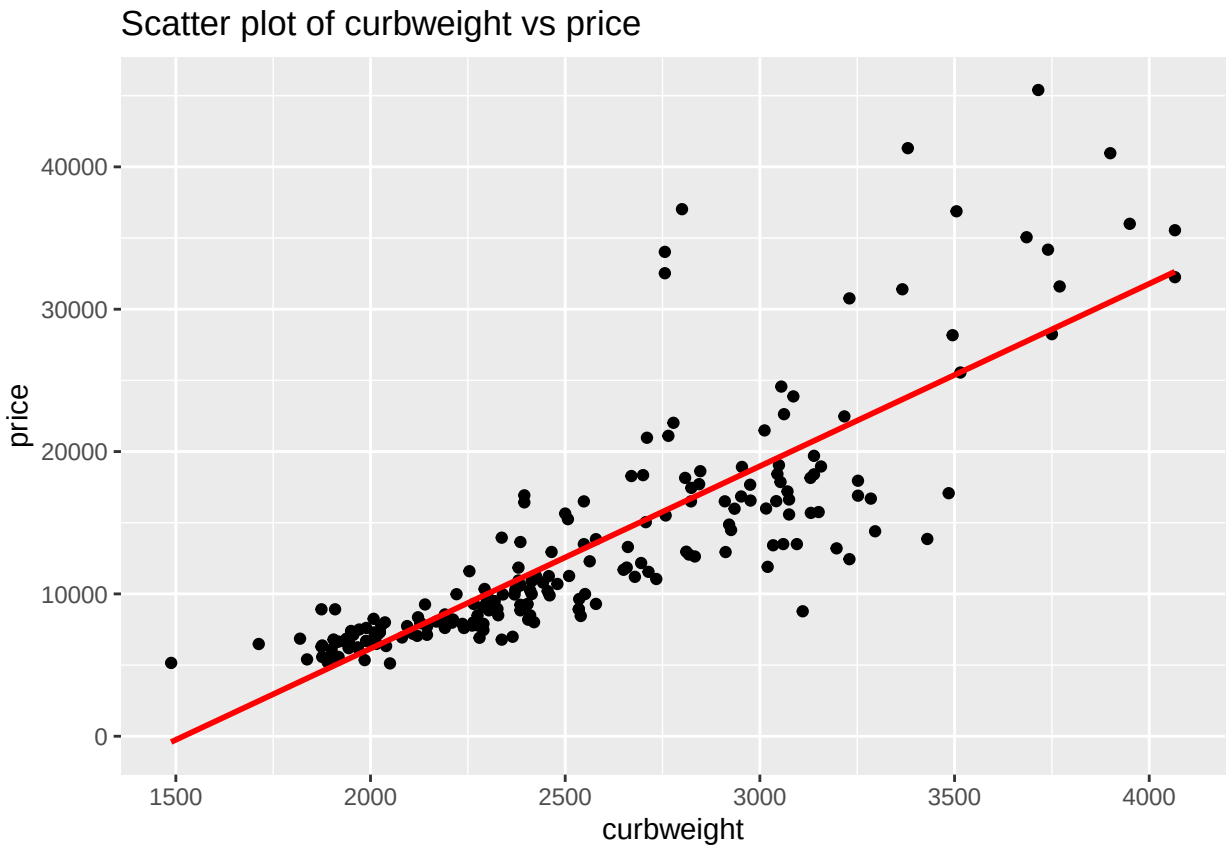

```
## `geom_smooth()` using formula = 'y ~ x'
```




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

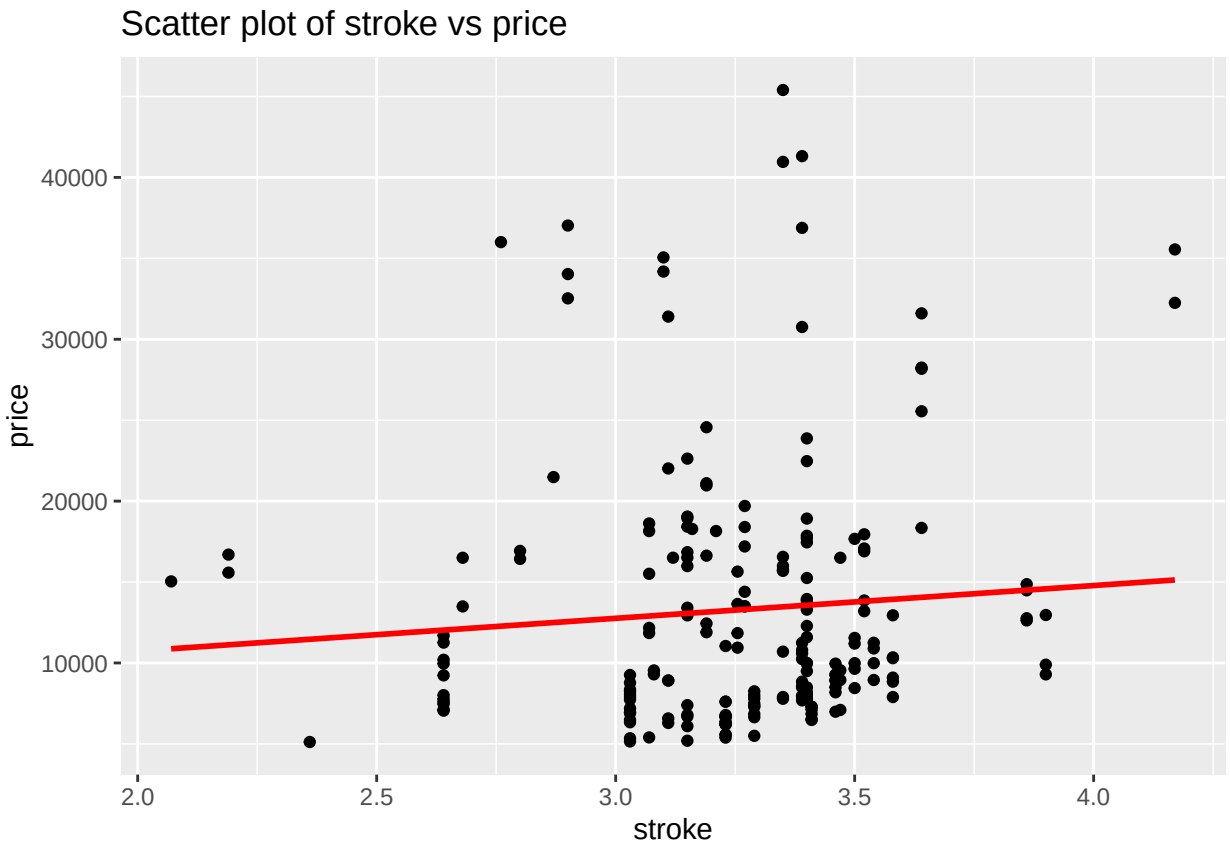



```
## `geom_smooth()` using formula = 'y ~ x'
```

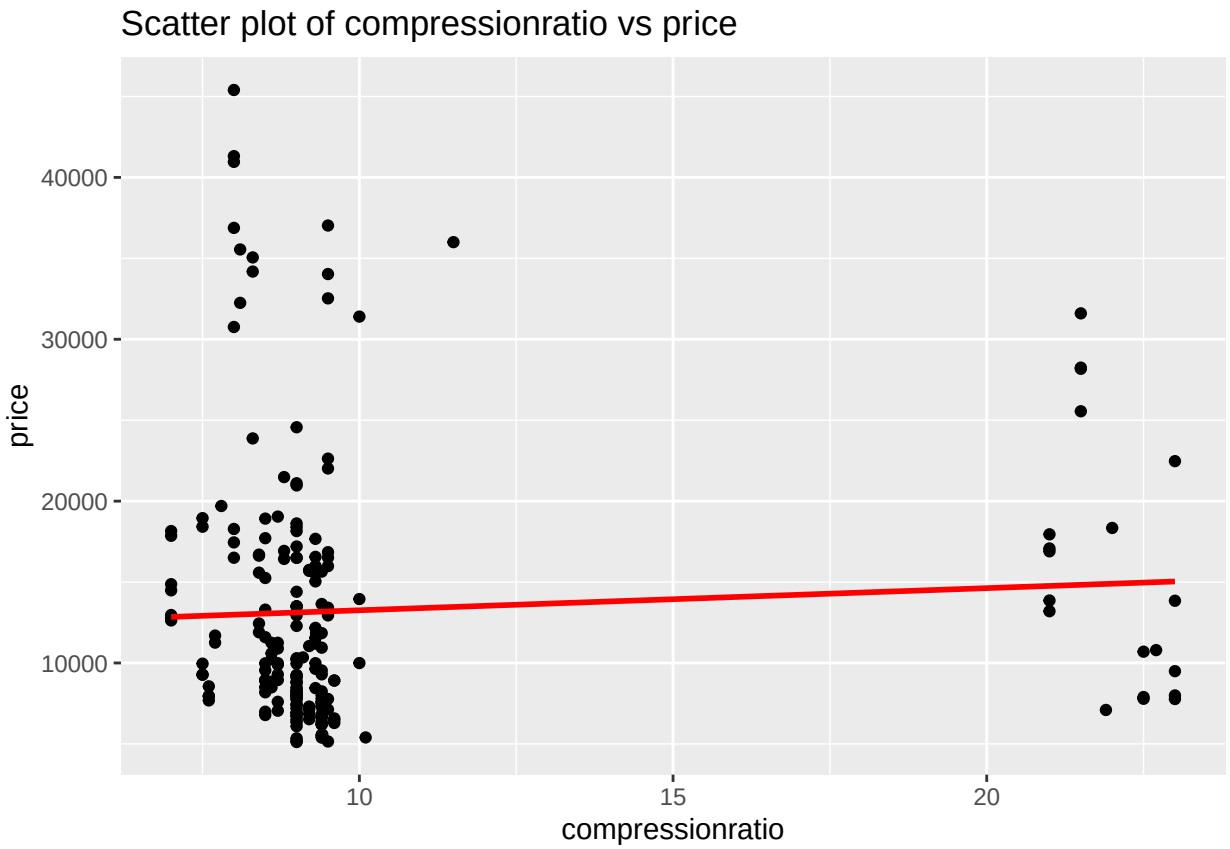



```
## `geom_smooth()` using formula = 'y ~ x'
```

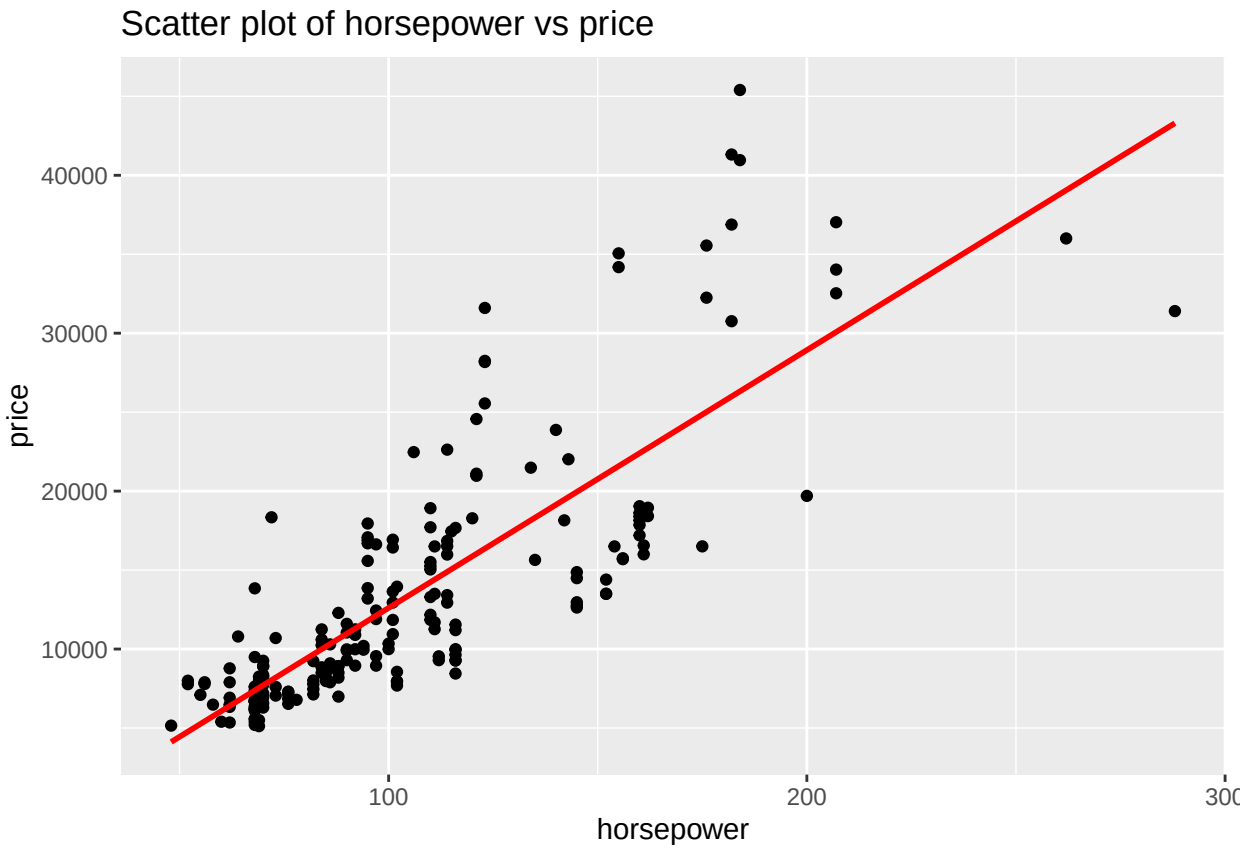

```
## `geom_smooth()` using formula = 'y ~ x'
```



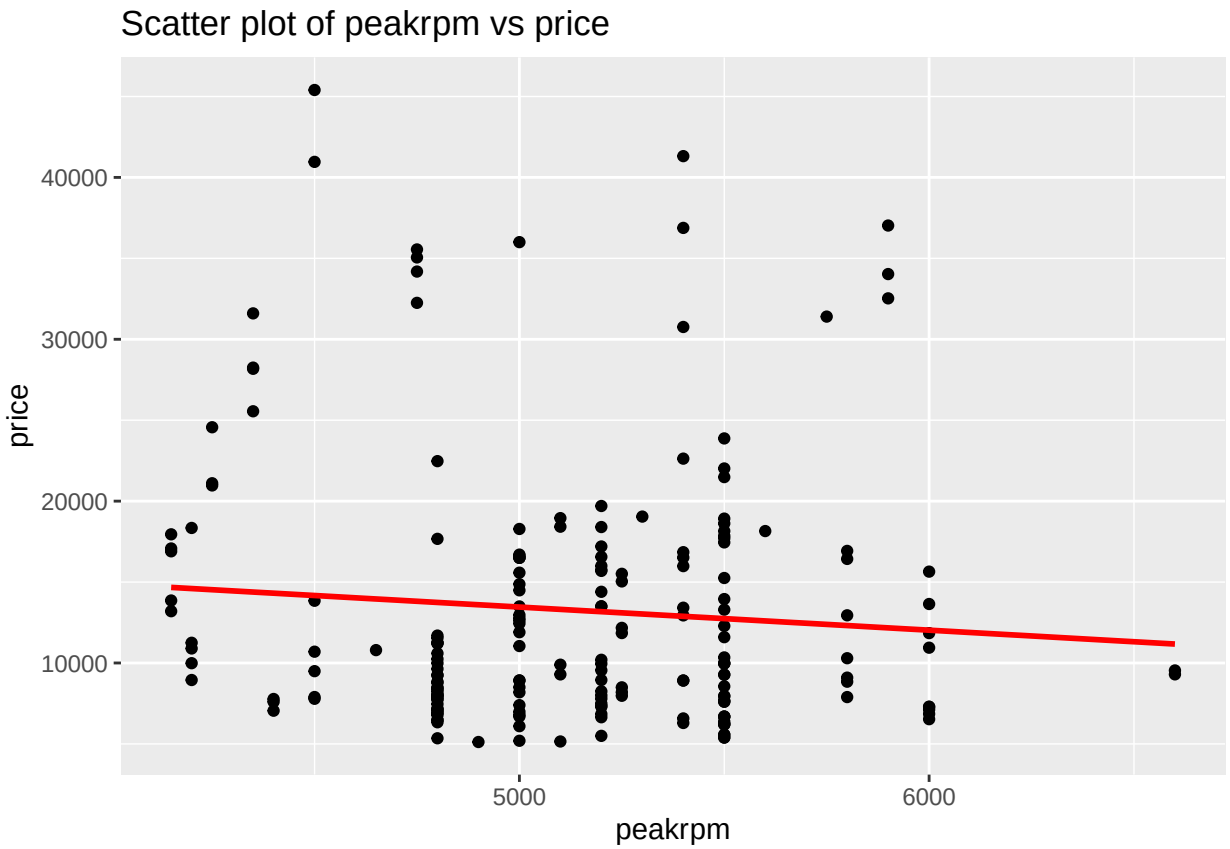

```
## `geom_smooth()` using formula = 'y ~ x'
```



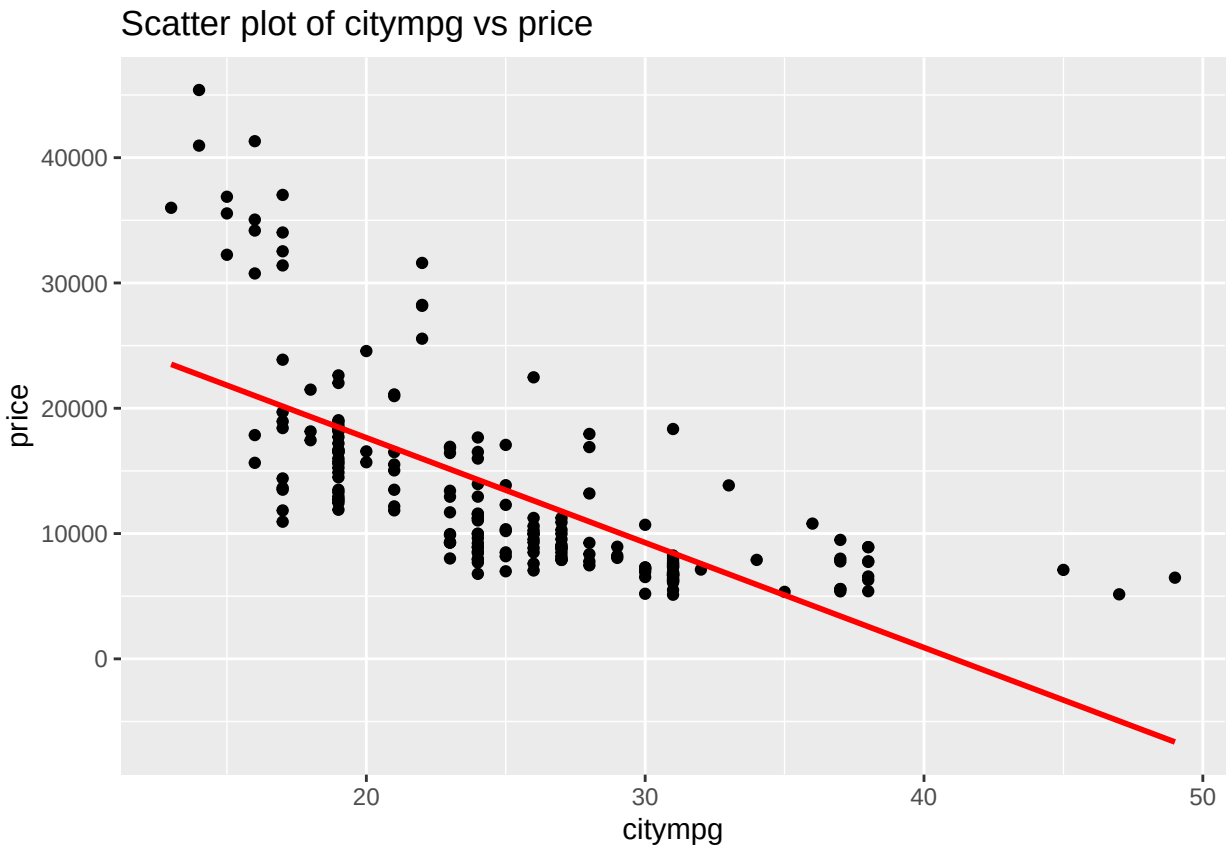

```
## `geom_smooth()` using formula = 'y ~ x'
```




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




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




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

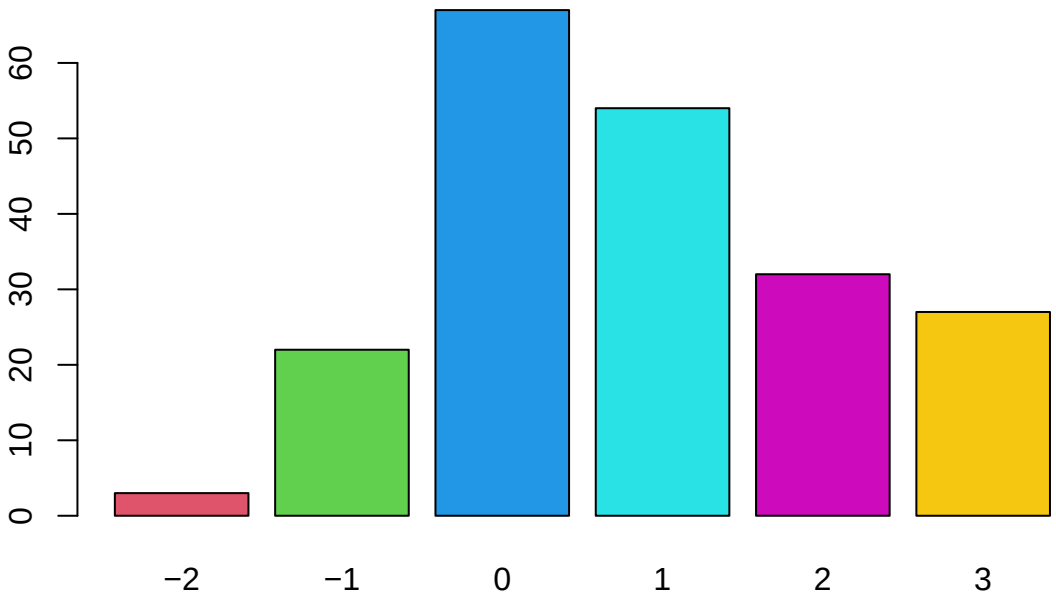
b) Variables categóricas

```
T = table(categorical_df$symboling)
T
```

Diagrama de barra

-2	-1	0	1	2	3
3	22	67	54	32	27

```
barplot(T, col = 2:15, main = "Frecuencia en categoría de riesgos por vehiculo")
```

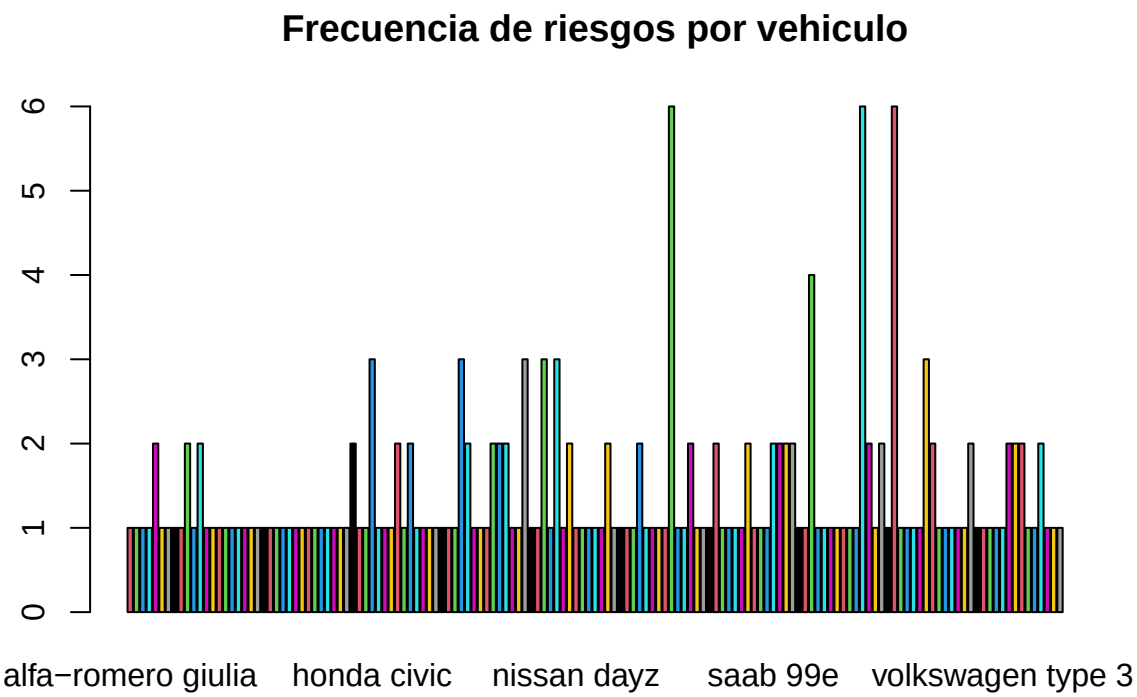


Se observa que es de utilidad la clase “category”, ya que se acumulan 60 presencias en la categoría “0”, y 50 en la de “1” y se puede interpretar que la mayoría de los valores se acumulan entre esas dos categorías. Por otra parte hay 30 valores para categoría de riesgo “2” y muy pocas para la categoría más baja de riesgo.

```
T = table(categorical_df$CarName)
T
```

alfa-romero giulia	alfa-romero stelvio	alfa-romero Quadrifoglio	audi 100 ls	audi 100ls	audi 5000	audi 5000s (diesel)	audi fox	bmw 320i	bmw x1	bmw x3	bmw x4	bmw x5	bmw z4	buick century	buick century luxus (sw)	buick electra 225 custom	buick opel isuzu deluxe	buick regal sport coupe (turbo)	buick skylark	chevrolet impala	chevrolet monte carlo	dodge challenger se	dodge colt (sw)	dodge colt hardtop	dodge coronet custom	dodge coronet custom (sw)	dodge d200	dodge dart custom	dodge monaco (sw)	honda accord	honda accord cvcc	honda accord lx	honda civic	honda civic 1300	honda civic 1500 gl	honda civic cvcc	honda prelude	i
1	1	1	1	2	1	1	1	1	2	1	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	3	1	1	1	2	1	2	1

```
barplot(T, col = 2:15, main = "Frecuencia de riesgos por vehiculo")
```

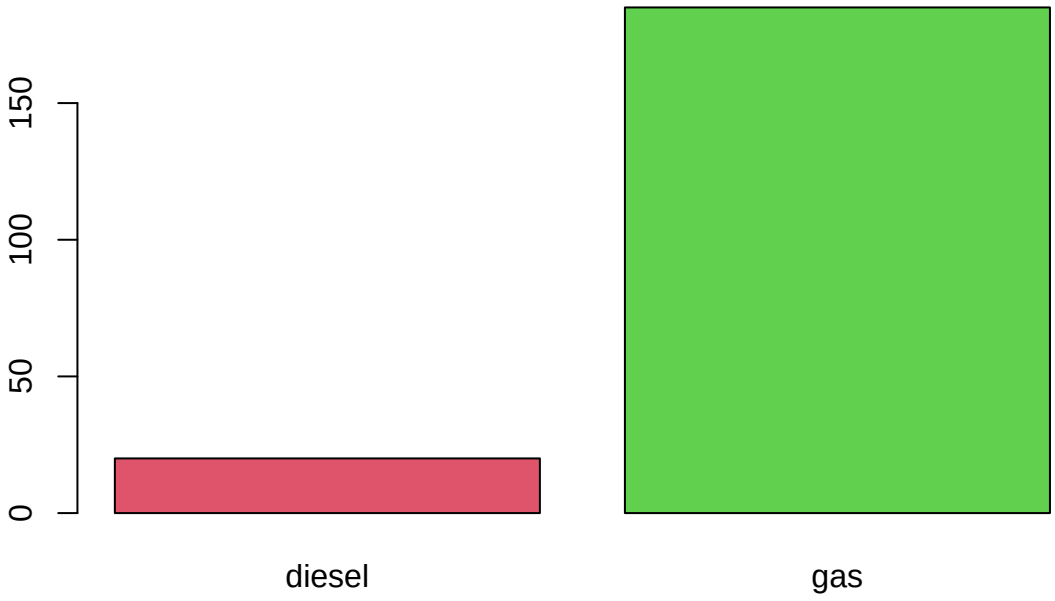


Se observa que no es de utilidad la clase "CarName", ya que son pocos valores por categoría y no tienen ningún patrón. La mayoría de las marcas tienen frecuencia de 1, y la máxima de 6.

```
T = table(categorical_df$fueltype)
T
```

diesel	gas
20	185

```
barplot(T, col = 2:15, main = "Frecuencia de combustible por vehiculo")
```

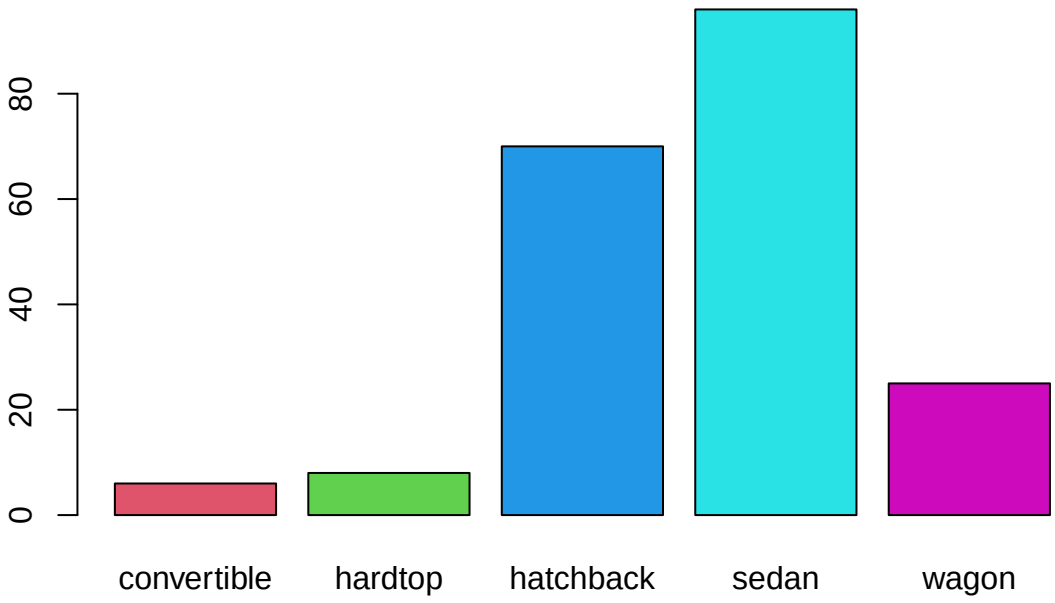


Aquí es muy notorio que la mayoría de los vehículos utilizan gas, teniendo una frecuencia de más de 150 y de 25 para diésel.

```
T = table(categorical_df$carbody)
T
```

convertible	hardtop	hatchback	sedan	wagon
6	8	70	96	25

```
barplot(T, col = 2:15, main = "Frecuencia de  tipo de vehiculo")
```

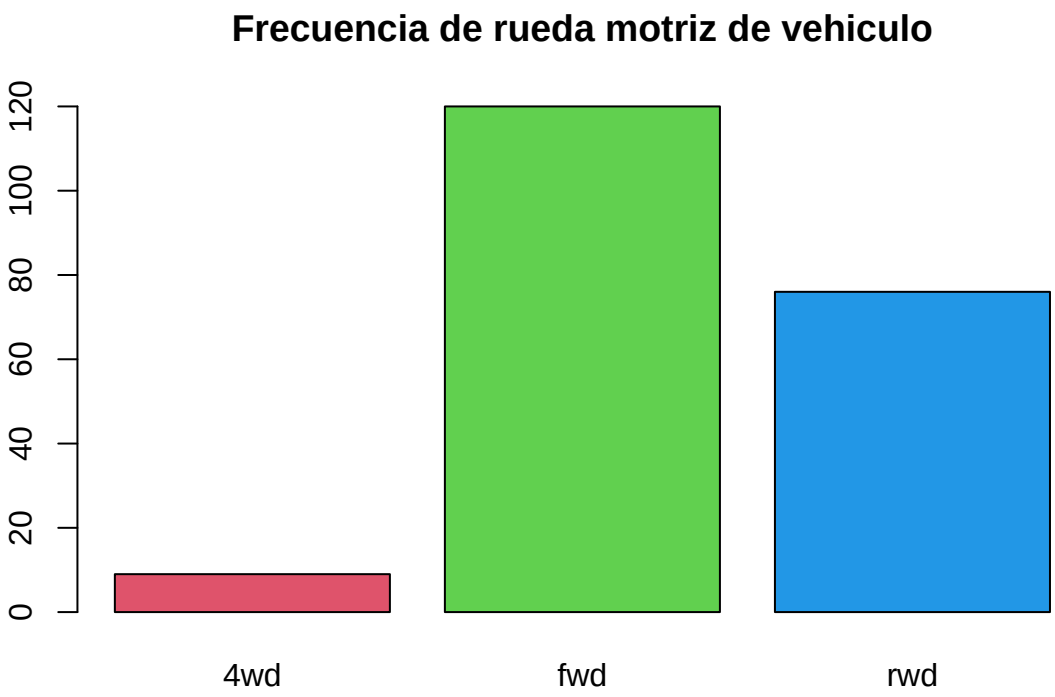


Se observa que hay más de 80 vehículos sedán, aproximadamente 70 de hatchback, y pocos convertibles, hardtop, y wagon.

```
T = table(categorical_df$drivewheel)
T
```

4wd	fwd	rwd
9	120	76

```
barplot(T, col = 2:15, main = "Frecuencia de rueda motriz de vehiculo")
```

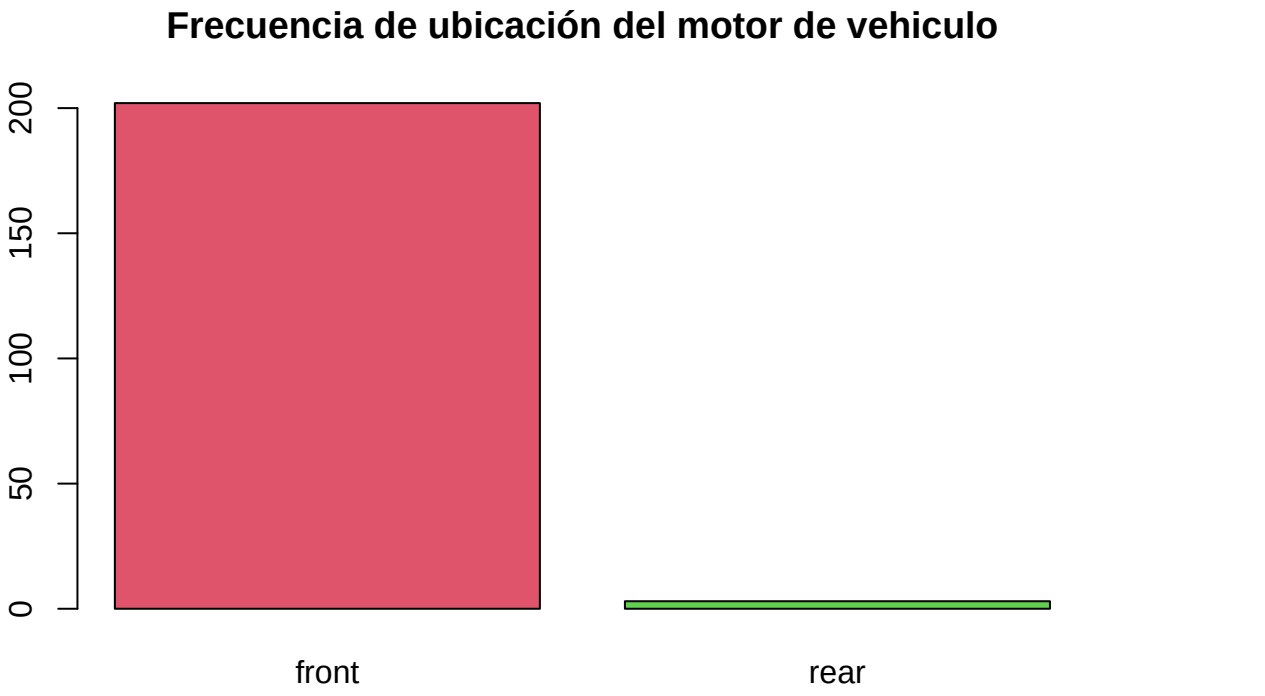


Se observa que más de 120 vehículos utilizan una rueda motriz “fwd”, la mitad utiliza un rwd y por último casi 10 utilizan “4wd”. Por lo que el “4wd” no es tan significativo en el análisis.

```
T = table(categorical_df$enginelocation)
T
```

front	rear
202	3

```
barplot(T, col = 2:15, main = "Frecuencia de ubicación del motor de vehiculo")
```

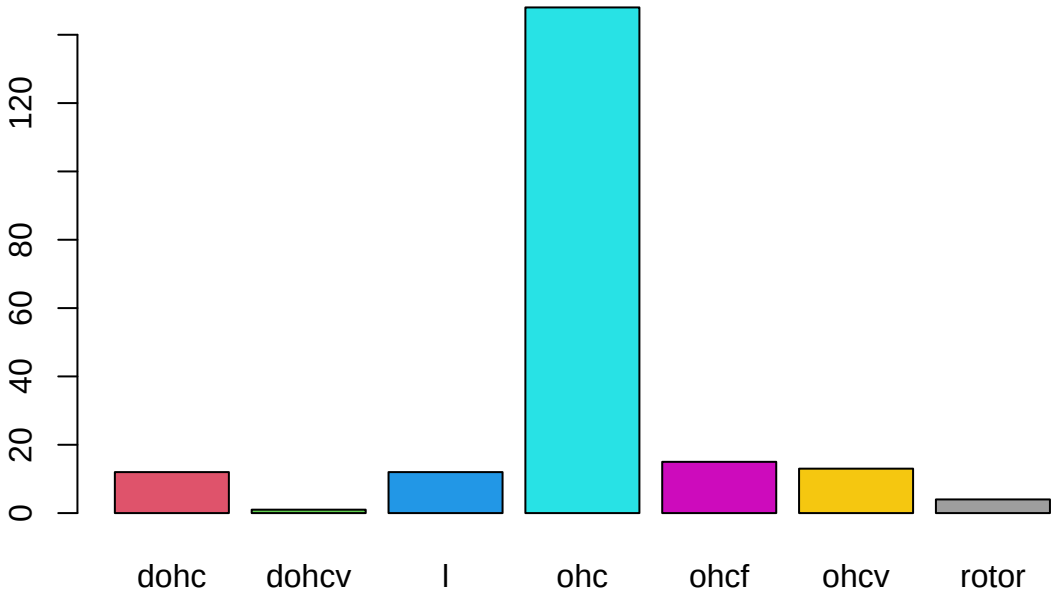


Es muy evidente que 200 vehículos ubican su motor al frente, y menos de 50 en la parte trasera por lo que se tomará como más significativo el los motores al frente del vehículo.

```
T = table(categorical_df$enginetype)
T
```

dohc	dohcv	l	ohc	ohcf	ohcv	rotor
12	1	12	148	15	13	4

```
barplot(T, col = 2:15, main = "Frecuencia de tipo de motor del vehiculo")
```

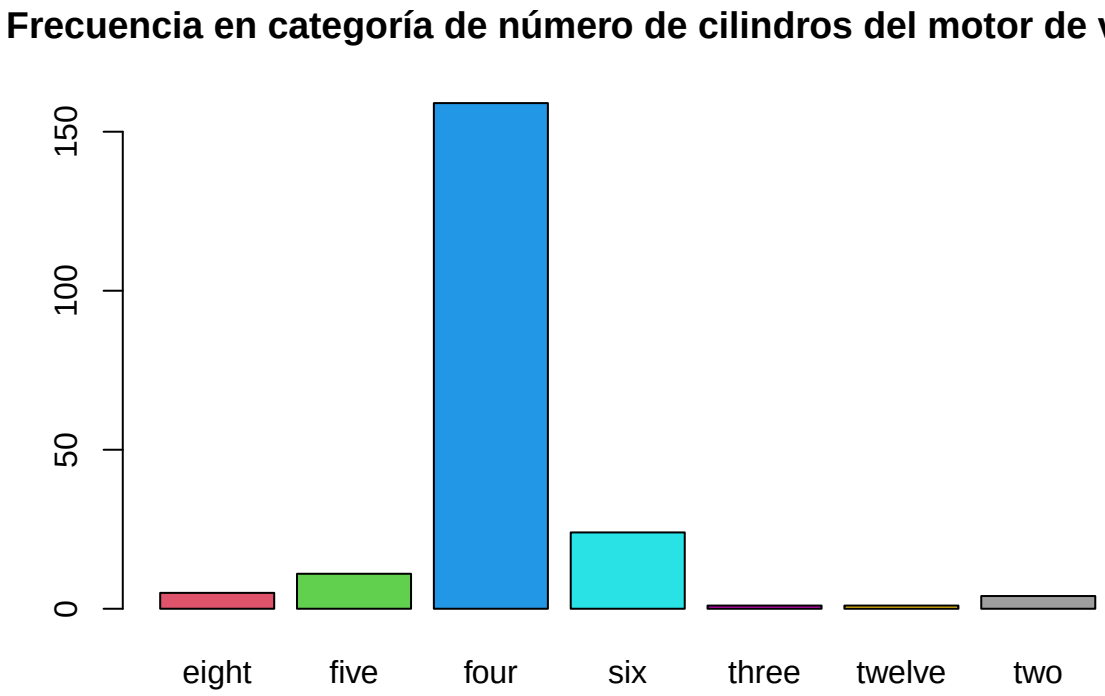


La mayoría de los vehículos utilizan el motor “ohc”, los valores “dohcv” y “rotor” son casi nulos por lo que serán no significativos.

```
T = table(categorical_df$cylindernumber)
T
```

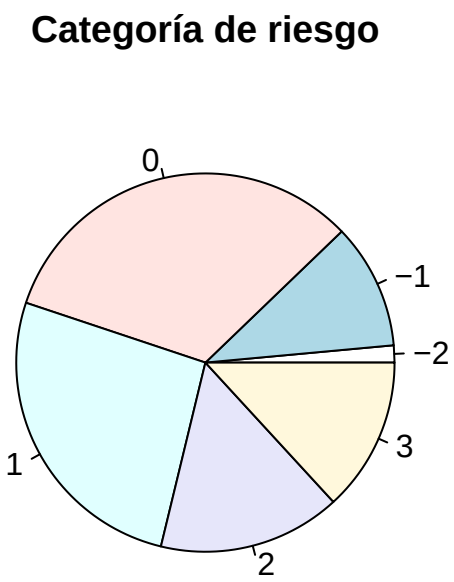
eight	five	four	six	three	twelve	two
5	11	159	24	1	1	4

```
barplot(T, col = 2:15, main = "Frecuencia en categoría de número de cilindros del motor de vehiculo")
```



Se observa que la mayoría de los motores utilizan 4 cilindros, y los demás son tan pequeños que no se les consideran significativos para el análisis.

```
Tabla = table(categorical_df$symboling)
Tabla=prop.table(Tabla)
names(Tabla)=c("-2","-1","0","1","2","3")
pie(Tabla, main = "Categoría de riesgo", labels = c("-2","-1","0","1","2","3"))
```

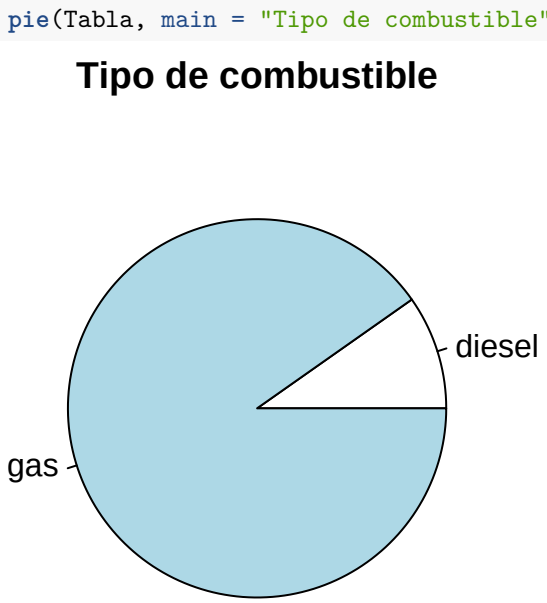


Diagramas de pastel

Se observa que hay una mayor cantidad de vehículos en la categoría de riesgo 0 y una muy baja en la -2.

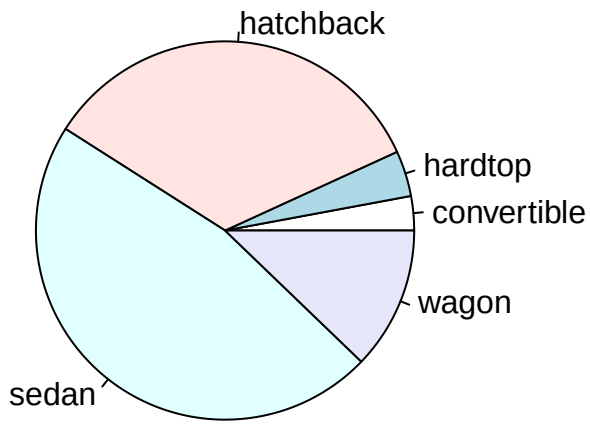
```
Tabla = table(categorical_df$fueltype)
Tabla=prop.table(Tabla)
names(Tabla)=c("diesel","gas")
```

```
pie(Tabla, main = "Tipo de combustible", labels = c("diesel","gas"))
```



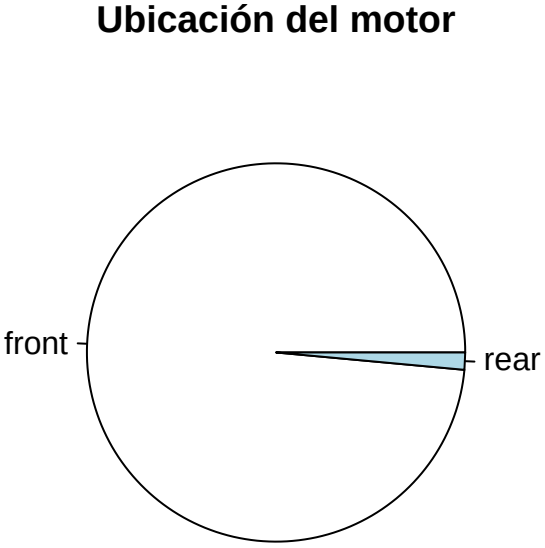
Así como se observó en la gráfica de barras, el uso o consumo de gas por vehículos predomina sobre el uso del diésel.

```
Tabla = table(categorical_df$carbody)
Tabla=prop.table(Tabla)
names(Tabla)=c("convertible",  "hardtop", "hatchback",   "sedan",  "wagon")
pie(Tabla, main = "Tipo de vehículo", labels = c("convertible",  "hardtop", "hatchback",   "sedan",  "wa
```



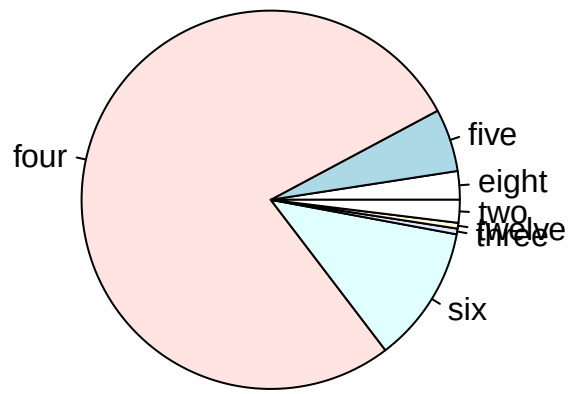
Así como en el gráfico de barras, se observa que predomina el sedán.

```
Tabla = table(categorical_df$enginelocation)
Tabla=prop.table(Tabla)
names(Tabla)=c("front", "rear")
pie(Tabla, main = "Ubicación del motor", labels = c("front", "rear"))
```



Así como en el gráfico de barras, se observa que la mayoría de los autos tienen su motor en el frente.

```
Tabla = table(categorical_df$cylindernumber)
Tabla=prop.table(Tabla)
names(Tabla)=c("eight",  "five", "four", "six",  "three", "twelve",  "two")
pie(Tabla, main = "Número de cilindros del motor", labels = c("eight",  "five", "four", "six",  "three"
```



Así como en el gráfico de barras, se observa que la mayoría de los autos tienen un motor de 4 cilindros.

Diagramas de caja y bigote de precio por categoría y barras por categoría

3. Identificación de problemas de calidad de datos

a) Valores faltantes

```
# Se verifica si hay NaN en el dataframe numerico

any(apply(numerical_df, 2, function(x) any(is.nan(x))))
```

Se evaluará si existen valores faltantes en el dataframe:

```
## [1] FALSE
```

No existen NaN en las variables numéricas

```
#Se verifica si hay NA en el dataframe categorico
any(is.na(categorical_df))
```

```
## [1] FALSE
```

No existen NA en las variables numéricas

Se puede observar que no existen valores faltantes en el data frame por lo que el análisis puede ser significativo.

b) Outliers

Se observan los datos atípicos para las variables numéricas Teniendo en cuenta los boxplots que se realizaron y analizaron previamente se puede observar que existen muchos datos atípicos en la mayoría de las clases, pero, principalmente en la relación de la compresión de los automóviles y la distribución del tamaño de los motores para los automóviles. Por otra parte, no existe ningún dato atípico en los pesos en vacío de los automóviles. Sin embargo se hizo un mayor énfasis en el análisis de cada gráfica para explicar esto de una mejor manera.

Se observan los datos atípicos para las variables categóricas Como se observó en el gráfico de barras realizado y analizado previamente, para la la clase de número de cilindros por vehículo se tienen como valores atípicos tres cilindros y veinte cilindros ya que su presencia es muy poca y es insignificante. Además, para el tipo de motor de vehículo se consideran como outliers el motor "dohcv", y "rotor". Sin embargo se hizo un mayor énfasis en el análisis de cada gráfica para explicar esto de una mejor manera.

4. Variables importantes para el análisis de las características de los automóviles que determinan su precio.

A partir de todo lo mencionado anteriormente, como las gráficas, las frecuencias de cada variable, el resumen de sus medidas estadísticas y correlaciones con la variable dependiente "precio", las variables que se consideran importantes para determinar el precio de un automóvil son:

1. Curbweight
2. Horsepower
3. Carwidth
4. enginesize
5. Citympg
6. Highwaympg
7. Symboling
8. Cylinder number
9. Engine type
10. Carbody

Estas variables fueron seleccionadas debido a que tienen una correlación o positiva o negativa con el precio del vehículo, y valores que son significativos para el análisis.

- Preparación de la base de datos

1. Selecciona el conjunto de datos a utilizar.

a) Maneja datos categóricos: transforma a variables dummy si es necesario. Primeramente eliminaré las variables que no serán de utilidad para mi análisis.

```r
columns <- c("curbweight", "horsepower", "carwidth", "enginesize", "citympg", "highwaympg", "symboling"

# Crear un nuevo dataframe solo con las columnas seleccionadas
df_selected <- M[, columns]
```

b) Maneja apropiadamente datos atípicos. Posteriormente, eliminaré los datos atípicos de las variables numéricas ya que eliminar los outliear de las variables categóricas realmente no influiría mucho en el análisis de los valores.

Para esto, definí las variables numéricas a las que se les quieren extraer los outliers.

```r
# Definir las variables numericas
num_vars <- c("curbweight", "horsepower", "carwidth", "enginesize", "citympg", "highwaympg", "price")

#remover outliers
remove_outliers <- function(df, num_vars) {
  # Se registran los valores atipicos
  keep_rows <- rep(TRUE, nrow(df))

  # Ciclo for para cada variable numerica
  for (var in num_vars) {
    # Se calculan los valores atipicos de cada variable numerica
    outliers <- boxplot.stats(df[[var]])$out

    # Se remueven las filas con outliers
    keep_rows <- keep_rows & !df[[var]] %in% outliers
  }

  # Se regresa un dataframe sin los valores atipicos
  df[keep_rows, ]
}

# Se remueven los outliers y se crea un nuevo dataframe con los valores sin outliers
df_selected_no_outliers <- remove_outliers(df_selected, num_vars)
```

Por último se crean variables dummy para el mejor análisis de los datos, volviendo las variables categóricas a numéricas en un rango entre 0 o 1. Si están presentes, estas tendrán un valor de 1, en caso contrario, su valor será de 0.

```r
data <- df_selected_no_outliers
# Convertir variables categoricas a variables dummy
dummy_columns <- character()

for (col in names(data)) {

  if (is.factor(data[[col]]) || is.character(data[[col]])) {

    dummy_mat <- model.matrix(~ data[[col]] - 1, data = data)

    colnames(dummy_mat) <- sub("^data\\[[\"']|['\"]\\]$", "", colnames(dummy_mat), perl = TRUE)

    # Agregar columnas dummy a dataframe sin outliers
    data <- cbind(data, dummy_mat)

    # Almacenar nombres de las columnas dummy para cada variable categorica
```

```
    dummy_columns <- c(dummy_columns, colnames(dummy_mat))
  }
}

# Crear un nuevo dataframe solo con las columnas numéricas y las variables dummy, sin categoricas
numeric_columns <- c("curbweight", "horsepower", "carwidth", "enginesize", "citympg", "highwaympg", "pri
selected_columns <- c(dummy_columns, numeric_columns)
df_cleaned <- data[selected_columns]
```

2. Transforma los datos en caso necesario.

a) Revisa si es necesario discretizar los datos Considero que no es necesario discretizar los datos ya que al predecir el precio de un automóvil es preferible mantener las variables continuas en lugar de discretizarlas para conservar la información detallada en los datos sin perder información importante sobrbe la relación entrelas variables de los automóviles y su precio. Al tener una gran cantidad de variables continuas creo que perdería valores reales al discretizarlos y no reflejaría la realidad del contexto, lo cual no es conveniente para el cliente.

b) Revisa si es necesario escalar y normalizar los datos Teniendo en cuenta que las distintas variables tienen diferentes sesgos, y están distribuidos hacia la derecha o hacia la izquierda, considero prudente para el análisis el normalizar los datos. Además, si en el futuro decido usar algún algoritmo de optimización, la normalización puede mejorar la eficiencia del entrenamiento y tener una mejor predicción.

```
# Lista de variables numéricas
numeric_columns <- c("curbweight", "horsepower", "carwidth", "enginesize", "citympg", "highwaympg", "pri

# Normalizar variables numericas
normalize_zscore <- function(column) {
  (column - mean(column, na.rm = TRUE)) / sd(column, na.rm = TRUE)
}

# Crear un nuevo dataframe para los datos normalizados
df_normalized <- df_cleaned

# Normalizar las columnas numéricas
df_normalized[numeric_columns] <- lapply(df_normalized[numeric_columns], normalize_zscore)
```

4. Identificación de los datos influyentes

Para esta entrega se requiere un análisis de los datos con outliers, y sin outliers, por lo que crearé variables dummy para los datos con outliers.

```
data1 <- df_selected
# Convertir variables categoricas a variables dummy
dummy_columns <- character()

for (col in names(data1)) {

  if (is.factor(data1[[col]]) || is.character(data1[[col]])) {

    dummy_mat <- model.matrix(~ data1[[col]] - 1, data1 = data1)

    colnames(dummy_mat) <- sub("^data1\\[[\"']|['\"]\\]$", "", colnames(dummy_mat), perl = TRUE)

    # Agregar columnas dummy a dataframe sin outliers
```

```
    data1 <- cbind(data1, dummy_mat)

    # Almacenar nombres de las columnas dummy para cada variable categorica
    dummy_columns <- c(dummy_columns, colnames(dummy_mat))
  }
}

# Crear un nuevo dataframe solo con las columnas numéricas y las variables dummy, sin categoricas
numeric_columns <- c("curbweight", "horsepower", "carwidth", "enginesize", "citympg", "highwaympg", "pri
selected_columns <- c(dummy_columns, numeric_columns)
df_outliers <- data1[selected_columns]
```

Entonces, teniendo las variables dummy con outliers en "df_outliers" y sin outliers y normalizada en "df_normalized", se procederá a la obtención de datos influyentes.

Teniendo las variables predictoras alamacenadas en "X" y la variable dependiente en "Y", se crea un modelo de regresión lineal con las variables predictoras. Posteriormente se calcularán las medidas de influencia utilizando "influence.measures()" y se calculará el umbral de la distancia de Cook, si existen valores fuera de ese umbral serán valores influyentes.

```
X <- df_outliers[, c("horsepower", "curbweight", "carwidth", "enginesize", "highwaympg", "citympg")]
Y <- df_outliers$price

# Construir la fórmula
formula <- as.formula(paste("Y ~", paste(names(X), collapse = "+")))

# Crear el modelo
modelo <- lm(formula, data = df_outliers)

# Obtener la influencia de los datos
influencia <- influence.measures(modelo)

# Cook's distance
umbral_cooks <- 4/nrow(df_outliers)

# Obtener valores influyentes
valores_influyentes <- which(influencia$cooks > umbral_cooks)

# Mostrar índices de valores influyentes
print(valores_influyentes)
```

Para datos con outliers

```
## integer(0)
```

Después de realizar los cálculos, se determinó que no existen valores influyentes en el conjunto de datos según el umbral de la distancia de Cook definido. Por lo que se concluye con que ninguna observación tuvo un impacto significativo en el modelo de regresión lineal en términos de influencia sobre los resultados.

```
X1 <- df_normalized[, c("horsepower", "curbweight", "carwidth", "enginesize", "highwaympg", "citympg")]
Y1 <- df_normalized$price

# Construir la fórmula
```

```r
formula <- as.formula(paste("Y1 ~", paste(names(X1), collapse = "+")))

# Crear el modelo
modelo <- lm(formula, data = df_normalized)

# Obtener la influencia de los datos
influencia <- influence.measures(modelo)

# Cook's distance
umbral_cooks <- 4/nrow(df_outliers)

# Obtener valores influyentes
valores_influyentes <- which(influencia$cooks > umbral_cooks)

# Mostrar índices de valores influyentes
print(valores_influyentes)
```

Para datos sin outliers

```
## integer(0)
```

Después de realizar los cálculos, se determinó que no existen valores influyentes en el conjunto de datos según el umbral de la distancia de Cook definido. Por lo que se concluye con que ninguna observación tuvo un impacto significativo en el modelo de regresión lineal en términos de influencia sobre los resultados.

- Análisis de datos y pregunta base

1. Selecciona al menos dos de las herramientas estadísticas que hemos analizado en el curso: regresión lineal simple y múltiple, anova o pruebas de hipótesis (medias o proporción). Justifica la elección de la herramienta estadística.

Para este caso se están trabajando con variables categóricas y numéricas, por lo que una regresión lineal simple no será de una gran ayuda. Por lo tanto, se seleccionaron las siguientes herramientas estadísticas:

1. Regresión lineal múltiple: Esta herramienta estadística fue seleccionada ya que puede manejar tanto variables numéricas como categóricas, y proporciona coeficientes para cada variable predictora, lo que permite cuantificar el efecto estimado de cada variable.
2. Pruebas de hipótesis para medias: Esta herramienta estadística se seleccionó porque dado que la empresa automovilística china está interesada en competir en el mercado estadounidense, se pueden tomar decisiones al conocer las diferencias de precios entre las categorías planteadas. Además, pueden evaluar si la varianza de los precios es similar en diferentes grupos.

Pruebas de hipótesis

1. Se comprueban los supuestos requeridos por el modelo:

```r
# Se aplica la prueba de normalidad Shapiro-Wilk a cada categoría
for (col in names(df_outliers)) {
  p_value <- shapiro.test(df_outliers[[col]])$p.value
  if (p_value < 0.05) {
    cat(col, "(p-value:", p_value, ")\n")
  } else {
    cat(col, p_value, ")\n")
```

```
  }
}
```

1. Prueba de Normalidad

```
## data1[[col]]eight (p-value: 1.346487e-29 )
## data1[[col]]five (p-value: 2.829188e-28 )
## data1[[col]]four (p-value: 1.823143e-23 )
## data1[[col]]six (p-value: 4.212195e-26 )
## data1[[col]]three (p-value: 8.469581e-31 )
## data1[[col]]twelve (p-value: 8.469581e-31 )
## data1[[col]]two (p-value: 7.392494e-30 )
## data1[[col]]dohc (p-value: 4.416892e-28 )
## data1[[col]]dohcv (p-value: 8.469581e-31 )
## data1[[col]]l (p-value: 4.416892e-28 )
## data1[[col]]ohc (p-value: 1.719543e-22 )
## data1[[col]]ohcf (p-value: 1.561463e-27 )
## data1[[col]]ohcv (p-value: 6.806816e-28 )
## data1[[col]]rotor (p-value: 7.392494e-30 )
## data1[[col]]convertible (p-value: 2.36989e-29 )
## data1[[col]]hardtop (p-value: 6.77949e-29 )
## data1[[col]]hatchback (p-value: 1.31902e-21 )
## data1[[col]]sedan (p-value: 1.036975e-20 )
## data1[[col]]wagon (p-value: 5.860206e-26 )
## curbweight (p-value: 2.891342e-06 )
## horsepower (p-value: 1.736468e-11 )
## carwidth (p-value: 5.013066e-09 )
## enginesize (p-value: 3.056836e-14 )
## citympg (p-value: 7.824283e-06 )
## highwaympg (p-value: 0.0006515526 )
## price (p-value: 1.849093e-15 )
```

Como se observa en los resultados, no hay distribuciones normales ya que el valor de p es muy pequeño, por lo que se hará una normalización de los datos con valores atípicos:

```
# Lista de variables numéricas
numeric_columns1 <- c("curbweight", "horsepower", "carwidth", "enginesize", "citympg", "highwaympg", "pr

# Normalizar variables numericas
normalize_zscore1 <- function(column) {
  (column - mean(column, na.rm = TRUE)) / sd(column, na.rm = TRUE)
}

# Crear un nuevo dataframe para los datos normalizados
df_normalized1 <- df_outliers

# Normalizar las columnas numéricas
df_normalized1[numeric_columns1] <- lapply(df_normalized1[numeric_columns1], normalize_zscore1)
```

2. Homogeneidad de Varianzas

Para probar si existe homogeneidad entre las varianzas de las variables numéricas se utilizará la prueba de Levene, que evalúa si las varianzas de diferentes grupos son estadísticamente diferentes.

```
# Grupo al que pertenecen los valores
grupo <- rep(c("horsepower", "curbweight", "carwidth", "enginesize", "highwaympg", "citympg"), each = n
```

```r
valores <- c(df_normalized1$horsepower, df_normalized1$curbweight, df_normalized1$carwidth, df_normaliz

# Prueba de Levene
leveneTest(valores, grupo)
```

```
## Warning in leveneTest.default(valores, grupo): grupo coerced to factor.
```

	Df	F value	Pr(>F)
group	5	0.7950145	0.5532388
	1224	NA	NA

Tomando, por ejemplo, un alpha de 0.03 y teniendo un valor F de 0.79, entonces no se rechaza la hipótesis nula de que las varianzas de las variables numéricas son iguales entre los grupos. Por lo que se pasa la prueba de homogeneidad de varianzas.

2. Aplicación de la herramienta estadística

Regresión lineal múltiple

1. Se analizan los datos proporcionados Se realiza una matriz de correlación

```r
#Se quiere observar la correlación entre las variables numéricas
df_correlacion <- df_normalized1[c("horsepower", "curbweight", "carwidth", "enginesize", "highwaympg",
matriz_correlacion <- cor(df_correlacion)
matriz_correlacion
```

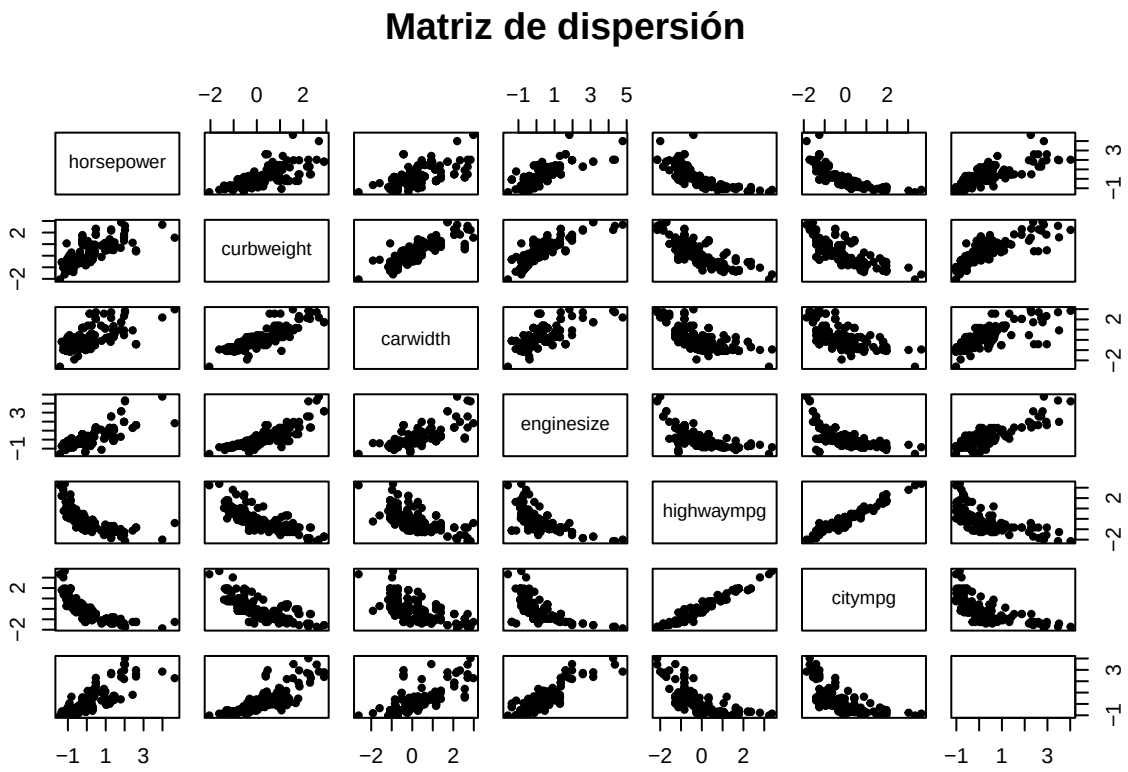
	horsepower	curbweight	carwidth	enginesize	highwaympg	citympg	price
horsepower	1.0000000	0.7507393	0.6407321	0.8097687	-0.7705439	-0.8014562	0.8081388
curbweight	0.7507393	1.0000000	0.8670325	0.8505941	-0.7974648	-0.7574138	0.8353049
carwidth	0.6407321	0.8670325	1.0000000	0.7354334	-0.6772179	-0.6427043	0.7593253
enginesize	0.8097687	0.8505941	0.7354334	1.0000000	-0.6774699	-0.6536579	0.8741448
highwaympg	-0.7705439	-0.7974648	-0.6772179	-0.6774699	1.0000000	0.9713370	-0.6975991
citympg	-0.8014562	-0.7574138	-0.6427043	-0.6536579	0.9713370	1.0000000	-0.6857513
price	0.8081388	0.8353049	0.7593253	0.8741448	-0.6975991	-0.6857513	1.0000000

```r
Rc = rcorr(as.matrix(df_correlacion))
Rc
```

```
##            horsepower curbweight carwidth enginesize highwaympg citympg price
## horsepower       1.00       0.75     0.64       0.81      -0.77   -0.80  0.81
## curbweight       0.75       1.00     0.87       0.85      -0.80   -0.76  0.84
## carwidth         0.64       0.87     1.00       0.74      -0.68   -0.64  0.76
## enginesize       0.81       0.85     0.74       1.00      -0.68   -0.65  0.87
## highwaympg      -0.77      -0.80    -0.68      -0.68       1.00    0.97 -0.70
## citympg         -0.80      -0.76    -0.64      -0.65       0.97    1.00 -0.69
## price            0.81       0.84     0.76       0.87      -0.70   -0.69  1.00
##
## n= 205
##
##
## P
```

```
##            horsepower curbweight carwidth enginesize highwaympg citympg price
## horsepower             0          0         0          0         0       0
## curbweight  0                     0         0          0         0       0
## carwidth    0          0                    0          0         0       0
## enginesize  0          0          0                    0         0       0
## highwaympg  0          0          0         0                    0       0
## citympg     0          0          0         0          0                 0
## price       0          0          0         0          0         0
```

```
pairs(df_correlacion,labels=c("horsepower", "curbweight", "carwidth", "enginesize", "highwaympg", "cityn
```



Ahora, teniendo en cuenta que unas variables son más significativas que otras, se debe generar un modelo que sea eficiente y hacer un análisis de correlación entre estas variables. Se deberán ir quitando las variables 1 a 1 para no afectar la significación. Se puede observar que carwidth y curbweight tienen una alta correlación. Sin embargo, curbweight tiene una mayor correlación con price, por lo tanto se utilizará esta. Además, highway mpg y citympg tienen una alta correlación, pero highway tiene una mayor correlación negativa con price, por lo que se usará este.

Se realizará el modelo con las variables seleccionadas:

1. Curbweight
2. Horsepower
3. enginesize
4. Highwaympg
5. Cylinder number
6. Engine type
7. Carbody

```
df_normalized2 <- df_normalized1
```

```r
df_normalized2 <- subset(df_normalized2, select = -c(citympg, carwidth))
```

```r
nuevos_nombres <- c(
  "eight", "five", "four", "six", "three", "twelve", "two", "dohc",
  "dohcv", "l", "ohc", "ohcf", "ohcv",
  "rotor", "convertible", "hardtop", "hatchback",
  "sedan", "wagon", "curbweight", "horsepower", "enginesize","highwaympg","price"
)

colnames(df_normalized2) <- nuevos_nombres
```

```r
R=lm(price~eight+five+four+ six+ three+twelve+two+dohc+dohcv+l+ ohc+ +ohcf+ohcv+rotor+convertible+hardt
summary(R)
```

2. Se crea un modelo de regresión lineal múltiple sin haber transformado las variables

```
##
## Call:
## lm(formula = price ~ eight + five + four + six + three + twelve +
##     two + dohc + dohcv + l + ohc + +ohcf + ohcv + rotor + convertible +
##     hardtop + hatchback + sedan + wagon + curbweight + horsepower +
##     enginesize + highwaympg, data = df_normalized2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.94005 -0.16119 -0.01172  0.15566  1.22424
##
## Coefficients: (4 not defined because of singularities)
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.35143    0.22561   1.558 0.121014
## eight        0.73441    0.36891   1.991 0.047981 *
## five        -1.22231    0.25873  -4.724 4.56e-06 ***
## four        -1.75580    0.23663  -7.420 4.12e-12 ***
## six         -0.97011    0.25799  -3.760 0.000228 ***
## three       -1.38817    0.45048  -3.082 0.002374 **
## twelve      -0.76823    0.53692  -1.431 0.154172
## two               NA         NA      NA       NA
## dohc         0.69467    0.15756   4.409 1.76e-05 ***
## dohcv       -1.50607    0.47106  -3.197 0.001632 **
## l            1.13245    0.19456   5.821 2.54e-08 ***
## ohc          1.19227    0.15037   7.929 2.03e-13 ***
## ohcf         1.25248    0.16532   7.576 1.65e-12 ***
## ohcv              NA         NA      NA       NA
## rotor             NA         NA      NA       NA
## convertible  0.62732    0.15836   3.961 0.000106 ***
## hardtop      0.21168    0.15566   1.360 0.175542
## hatchback    0.04217    0.08923   0.473 0.637039
## sedan        0.19031    0.07933   2.399 0.017440 *
## wagon             NA         NA      NA       NA
## curbweight   0.41037    0.07750   5.295 3.35e-07 ***
## horsepower   0.38444    0.06378   6.028 8.81e-09 ***
## enginesize   0.15260    0.09084   1.680 0.094655 .
## highwaympg   0.14654    0.05172   2.833 0.005117 **
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3215 on 185 degrees of freedom
## Multiple R-squared:  0.9062, Adjusted R-squared:  0.8966
## F-statistic: 94.12 on 19 and 185 DF,  p-value: < 2.2e-16
```

3. Elección del mejor modelo con el criterio de información de Akaike

```
step(R,direction="both",trace=1)
```

Modelo mixto

```
## Start:  AIC=-446.26
## price ~ eight + five + four + six + three + twelve + two + dohc +
##     dohcv + l + ohc + +ohcf + ohcv + rotor + convertible + hardtop +
##     hatchback + sedan + wagon + curbweight + horsepower + enginesize +
##     highwaympg
##
##
## Step:  AIC=-446.26
## price ~ eight + five + four + six + three + twelve + two + dohc +
##     dohcv + l + ohc + ohcf + ohcv + rotor + convertible + hardtop +
##     hatchback + sedan + curbweight + horsepower + enginesize +
##     highwaympg
##
##
## Step:  AIC=-446.26
## price ~ eight + five + four + six + three + twelve + two + dohc +
##     dohcv + l + ohc + ohcf + ohcv + convertible + hardtop + hatchback +
##     sedan + curbweight + horsepower + enginesize + highwaympg
##
##
## Step:  AIC=-446.26
## price ~ eight + five + four + six + three + twelve + two + dohc +
##     dohcv + l + ohc + ohcf + convertible + hardtop + hatchback +
##     sedan + curbweight + horsepower + enginesize + highwaympg
##
##
## Step:  AIC=-446.26
## price ~ eight + five + four + six + three + twelve + dohc + dohcv +
##     l + ohc + ohcf + convertible + hardtop + hatchback + sedan +
##     curbweight + horsepower + enginesize + highwaympg
##
##                Df Sum of Sq    RSS     AIC
## - hatchback     1    0.0231 19.148 -448.01
## <none>                      19.125 -446.26
## - hardtop       1    0.1912 19.316 -446.22
## - twelve        1    0.2116 19.337 -446.00
## - enginesize    1    0.2918 19.417 -445.16
## - eight         1    0.4097 19.535 -443.92
## - sedan         1    0.5949 19.720 -441.98
## - highwaympg    1    0.8299 19.955 -439.55
```

```
## - three        1    0.9817 20.107 -438.00
## - dohcv        1    1.0567 20.182 -437.24
## - six          1    1.4617 20.587 -433.16
## - convertible  1    1.6223 20.747 -431.57
## - dohc         1    2.0095 21.135 -427.78
## - five         1    2.3073 21.433 -424.91
## - curbweight   1    2.8986 22.024 -419.33
## - l            1    3.5025 22.628 -413.79
## - horsepower   1    3.7565 22.882 -411.50
## - four         1    5.6917 24.817 -394.85
## - ohcf         1    5.9337 25.059 -392.87
## - ohc          1    6.4989 25.624 -388.29
##
## Step:  AIC=-448.01
## price ~ eight + five + four + six + three + twelve + dohc + dohcv +
##     l + ohc + ohcf + convertible + hardtop + sedan + curbweight +
##     horsepower + enginesize + highwaympg
##
##               Df Sum of Sq    RSS     AIC
## - hardtop      1    0.1750 19.323 -448.15
## <none>                     19.148 -448.01
## - twelve       1    0.2586 19.407 -447.26
## - enginesize   1    0.3526 19.501 -446.27
## + hatchback    1    0.0231 19.125 -446.26
## + wagon        1    0.0231 19.125 -446.26
## - eight        1    0.3883 19.537 -445.90
## - highwaympg   1    0.8562 20.005 -441.05
## - three        1    1.0251 20.173 -439.32
## - dohcv        1    1.0713 20.220 -438.85
## - sedan        1    1.1177 20.266 -438.38
## - six          1    1.6147 20.763 -433.42
## - convertible  1    1.7942 20.942 -431.65
## - dohc         1    2.0062 21.154 -429.59
## - five         1    2.4022 21.550 -425.79
## - curbweight   1    3.2025 22.351 -418.31
## - l            1    3.5210 22.669 -415.41
## - horsepower   1    4.1332 23.282 -409.95
## - ohcf         1    5.9597 25.108 -394.46
## - four         1    5.9669 25.115 -394.41
## - ohc          1    6.4777 25.626 -390.28
##
## Step:  AIC=-448.15
## price ~ eight + five + four + six + three + twelve + dohc + dohcv +
##     l + ohc + ohcf + convertible + sedan + curbweight + horsepower +
##     enginesize + highwaympg
##
##               Df Sum of Sq    RSS     AIC
## <none>                     19.323 -448.15
## + hardtop      1    0.1750 19.148 -448.01
## - eight        1    0.3356 19.659 -446.62
## + wagon        1    0.0330 19.290 -446.50
## + hatchback    1    0.0069 19.316 -446.22
## - twelve       1    0.3860 19.709 -446.09
## - enginesize   1    0.5526 19.876 -444.37
```

```
## - highwaympg   1    0.9258 20.249 -440.56
## - sedan        1    0.9770 20.300 -440.04
## - three        1    1.1534 20.477 -438.26
## - dohcv        1    1.1681 20.491 -438.12
## - convertible  1    1.6799 21.003 -433.06
## - six          1    1.8895 21.213 -431.02
## - dohc         1    2.0647 21.388 -429.34
## - five         1    2.5526 21.876 -424.71
## - curbweight   1    3.0275 22.351 -420.31
## - l            1    3.8402 23.163 -412.99
## - horsepower   1    4.4225 23.746 -407.90
## - ohcf         1    6.4139 25.737 -391.39
## - four         1    6.4171 25.740 -391.37
## - ohc          1    6.7692 26.092 -388.58
```

```
##
## Call:
## lm(formula = price ~ eight + five + four + six + three + twelve +
##     dohc + dohcv + l + ohc + ohcf + convertible + sedan + curbweight +
##     horsepower + enginesize + highwaympg, data = df_normalized2)
##
## Coefficients:
## (Intercept)        eight         five         four          six        three
##      0.4456       0.6503      -1.2697      -1.8192      -1.0581      -1.4843
##      twelve         dohc        dohcv            l          ohc         ohcf
##     -0.9827       0.7034      -1.5746       1.1724       1.2093       1.2724
## convertible        sedan   curbweight   horsepower   enginesize   highwaympg
##      0.5708       0.1465       0.3746       0.4018       0.1955       0.1538
```

En este caso se busca obtener el menor AIC para medir el criterio,y saber qué tan bueno es el modelo utilizado. Mientras más pequeño es el AIC, mejor el modelo. El mejor modelo es el que utiliza eight, five, four, six, three, twelve, dohc, dohcv, l, ohc, ohcf, convertible, sedan, curbweight, horsepower, enginesize, highwaympg que da un AIC de -448.15.

4. Verificación del modelo

a) Economía de las variables: Ahora, se genera un nuevo modelo después de haber realizado el método de step:

```
R1 =lm(price ~ eight + five + four + six + three + twelve +
    dohc + dohcv + l + ohc + ohcf + convertible + sedan + curbweight +
    horsepower + enginesize + highwaympg, data = df_normalized2)
R1
```

```
##
## Call:
## lm(formula = price ~ eight + five + four + six + three + twelve +
##     dohc + dohcv + l + ohc + ohcf + convertible + sedan + curbweight +
##     horsepower + enginesize + highwaympg, data = df_normalized2)
##
## Coefficients:
## (Intercept)        eight         five         four          six        three
##      0.4456       0.6503      -1.2697      -1.8192      -1.0581      -1.4843
##      twelve         dohc        dohcv            l          ohc         ohcf
##     -0.9827       0.7034      -1.5746       1.1724       1.2093       1.2724
## convertible        sedan   curbweight   horsepower   enginesize   highwaympg
```

```
##      0.5708       0.1465       0.3746       0.4018       0.1955       0.1538
```

Se observa que se obtienen los coeficientes para cada variable. Antes eran 23 variables con 23 coeficientes, pero al momento de hacer la economía de las variables se obtienen 17 variables con 17 coeficientes.

Con esto se puede obtener la ecuación para la regresión lineal múltiple como:

```
b0 = R1$coefficients[1]
b1 = R1$coefficients[2]
b2 = R1$coefficients[3]
b3 = R1$coefficients[4]
b4 = R1$coefficients[5]
b5 = R1$coefficients[6]
b6 = R1$coefficients[7]
b7 = R1$coefficients[8]
b8 = R1$coefficients[9]
b9 = R1$coefficients[10]
b10 = R1$coefficients[11]
b11 = R1$coefficients[12]
b12 = R1$coefficients[13]
b13 = R1$coefficients[14]
b14 = R1$coefficients[15]
b15 = R1$coefficients[16]
b16 = R1$coefficients[17]
b17 = R1$coefficients[18]

cat("Precio = ", b0, "+", b1, "eight", b2, "five", b3, "four",b4,"six",b5, "three",b6, "twelve", "+", b
```

```
## Precio =  0.4455554 + 0.6503434 eight -1.269718 five -1.819154 four -1.058123 six -1.484294 three -0
```

b) Significación global (Prueba de F para el modelo): $\alpha = 0.03$

```
summary(R1)
```

```
##
## Call:
## lm(formula = price ~ eight + five + four + six + three + twelve +
##     dohc + dohcv + l + ohc + ohcf + convertible + sedan + curbweight +
##     horsepower + enginesize + highwaympg, data = df_normalized2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.93262 -0.16534 -0.01999  0.16153  1.19200
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.44556    0.19350   2.303 0.022400 *
## eight        0.65034    0.36088   1.802 0.073137 .
## five        -1.26972    0.25547  -4.970 1.51e-06 ***
## four        -1.81915    0.23085  -7.880 2.61e-13 ***
## six         -1.05812    0.24745  -4.276 3.03e-05 ***
## three       -1.48429    0.44427  -3.341 0.001008 **
## twelve      -0.98274    0.50848  -1.933 0.054781 .
## dohc         0.70339    0.15736   4.470 1.35e-05 ***
## dohcv       -1.57465    0.46834  -3.362 0.000938 ***
## l            1.17235    0.19231   6.096 6.08e-09 ***
```

```
## ohc          1.20930    0.14941   8.094 7.19e-14 ***
## ohcf         1.27244    0.16151   7.878 2.64e-13 ***
## convertible  0.57083    0.14157   4.032 8.05e-05 ***
## sedan        0.14647    0.04763   3.075 0.002421 **
## curbweight   0.37459    0.06920   5.413 1.89e-07 ***
## horsepower   0.40183    0.06142   6.542 5.65e-10 ***
## enginesize   0.19550    0.08454   2.313 0.021834 *
## highwaympg   0.15382    0.05139   2.993 0.003134 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3215 on 187 degrees of freedom
## Multiple R-squared:  0.9053, Adjusted R-squared:  0.8967
## F-statistic: 105.1 on 17 and 187 DF,  p-value: < 2.2e-16
```

Hipótesis

- Sobre el modelo (significación global):

$H_0 : \beta 1 + \beta 2 = ... = \beta_k = 0$

$H_1 : al\ menos\ un\ \beta 1 \neq 0$

Interpretación:

- Significación global

Teniendo: F = 105.1 Valor p = 2.2e-16

El valor p es muy pequeño, mucho menor a *alpha*. Por ende, se rechaza H_0.

c) Significación individual (Prueba de t de student para $beta_i$) : *Hipótesis* - Sobre las β_i (significación individual)

$H_0 : \beta_i = 0$

$H_1 : \beta_i \neq 0$

Interpretación

- Significación individual

Ignorando el intercepto, y teniendo en cuenta las variables predictoras con t* con valores muy grandes y diferentes de cero tanto positiva como negativamente, se puede concluir que se está muy lejos de la media, por lo tanto todas las variables seleccionadas son significativas. Por lo tanto se rechaza H_0.

d) Variación explicada por el modelo (coeficiente de determinación Tomando el Adjusted R-squared de 0.8967, se infiere que el modelo explica el 89.67% del precio, a partir de las variables predictoras.

5. Validez del modelo (Utilizando pruebas de hipótesis)

a) Normalidad de los residuos (prueba de Anderson Darling): H_0 : Los datos provienen de una población normal. H_1 : Los datos no provienen de una población normal.

Regla de decisión: * Se rechazará H_0 si p < *alpha*

```
library(nortest)
ad.test(R1$residuals)
```

```
##
##  Anderson-Darling normality test
```

```
##
## data:  R1$residuals
## A = 1.6251, p-value = 0.0003461
```

Se observa que el valor p es muy pequeño, por lo que se infiere que los datos no provienen de una distribución normal.

Tranformación de Yeo-Johnson Debido a que no se pasó la prueba de normalidad de los residuos, se realizará una transformación para hacer un modelo que contenga datos con una distribución normal.

Como se cuentan con valores negativos dentro de las variables predictoras, y se desea normalizar el conjunto de datos, se utilizará la transformación de Yeo-Johnson. Únicamente se aplicará esta transformación a las variables numéricas ya que las categóricas son variables dummy, por lo que no requieren tener una transformación.

```
# Variables numéricas
num_vars <- c("curbweight", "horsepower", "enginesize", "highwaympg")

# Se aplica la transformación de Yeo-Johnson a las variables numéricas y se guardan los resultados en d
df_final <- df_normalized2 %>%
  mutate(across(all_of(num_vars), ~ bestNormalize::yeojohnson(.x)$x.t, .names = "{.col}_yeojohnson"))
```

```
# Lista de columnas a eliminar
columnas_a_eliminar <- c("curbweight", "horsepower", "enginesize", "highwaympg")

# Se eliminan las columnas
df_final <- df_final[, !(names(df_final) %in% columnas_a_eliminar)]
```

```
nuevos_nombres <- c("curbweight", "horsepower", "enginesize", "highwaympg")
df_final <- df_final %>%
  rename(
    curbweight = curbweight_yeojohnson,
    horsepower = horsepower_yeojohnson,
    enginesize = enginesize_yeojohnson,
    highwaympg = highwaympg_yeojohnson
  )
```

Se hace el nuevo modelo de regresión lineal múltiple pero ahora con las variables normalizadas y transformadas

```
R2 =lm(price ~ eight + five + four + six + three + twelve +
    dohc + dohcv + l + ohc + ohcf + convertible + sedan + curbweight +
    horsepower + enginesize + highwaympg, data = df_final)
R2
```

Nuevo modelo de regresión lineal Múltiple

```
##
## Call:
## lm(formula = price ~ eight + five + four + six + three + twelve +
##     dohc + dohcv + l + ohc + ohcf + convertible + sedan + curbweight +
##     horsepower + enginesize + highwaympg, data = df_final)
##
## Coefficients:
## (Intercept)        eight         five         four          six        three
##    -0.06215      2.21763     -0.84228     -1.55438     -0.20358     -0.86513
##      twelve         dohc        dohcv            l          ohc         ohcf
```

```
##     1.57306      0.87625     -1.21635      1.33359      1.36681      1.40800
## convertible        sedan   curbweight   horsepower   enginesize   highwaympg
##     0.52346      0.10825      0.46169      0.32195     -0.06212      0.14884
```

```
step(R2,direction="both",trace=1)
```

Se realiza el el criterio de información de Akaike

```
## Start:  AIC=-380.89
## price ~ eight + five + four + six + three + twelve + dohc + dohcv +
##     l + ohc + ohcf + convertible + sedan + curbweight + horsepower +
##     enginesize + highwaympg
##
##               Df Sum of Sq    RSS     AIC
## - six          1    0.0507 26.878 -382.50
## - enginesize   1    0.0592 26.886 -382.44
## <none>                     26.827 -380.89
## - three        1    0.3872 27.214 -379.95
## - highwaympg   1    0.5075 27.335 -379.05
## - sedan        1    0.5343 27.361 -378.84
## - five         1    0.8297 27.657 -376.64
## - dohcv        1    0.8939 27.721 -376.17
## - twelve       1    1.3179 28.145 -373.06
## - convertible  1    1.3906 28.218 -372.53
## - horsepower   1    2.3845 29.212 -365.43
## - dohc         1    3.2248 30.052 -359.62
## - four         1    3.3029 30.130 -359.08
## - curbweight   1    4.0865 30.914 -353.82
## - eight        1    4.3549 31.182 -352.05
## - l            1    5.1267 31.954 -347.04
## - ohcf         1    7.9604 34.788 -329.62
## - ohc          1    8.7782 35.605 -324.86
##
## Step:  AIC=-382.5
## price ~ eight + five + four + three + twelve + dohc + dohcv +
##     l + ohc + ohcf + convertible + sedan + curbweight + horsepower +
##     enginesize + highwaympg
##
##               Df Sum of Sq    RSS     AIC
## <none>                     26.878 -382.50
## - enginesize   1    0.2978 27.176 -382.24
## - three        1    0.3391 27.217 -381.93
## + six          1    0.0507 26.827 -380.89
## - highwaympg   1    0.5024 27.380 -380.70
## - sedan        1    0.5149 27.393 -380.61
## - dohcv        1    0.9264 27.804 -377.55
## - convertible  1    1.4455 28.323 -373.76
## - five         1    2.3181 29.196 -367.54
## - horsepower   1    2.6246 29.502 -365.40
## - twelve       1    2.6711 29.549 -365.08
## - dohc         1    3.3062 30.184 -360.72
## - l            1    5.1316 32.009 -348.68
## - curbweight   1    5.1704 32.048 -348.43
## - ohcf         1    8.0905 34.968 -330.56
```

```
## - ohc          1    8.9262 35.804 -325.71
## - eight        1   12.7835 39.661 -304.74
## - four         1   16.8062 43.684 -284.94
```

```
##
## Call:
## lm(formula = price ~ eight + five + four + three + twelve + dohc +
##     dohcv + l + ohc + ohcf + convertible + sedan + curbweight +
##     horsepower + enginesize + highwaympg, data = df_final)
##
## Coefficients:
## (Intercept)        eight         five         four        three       twelve
##     -0.2162       2.4029      -0.6583      -1.3773      -0.7291       1.7632
##        dohc        dohcv            l          ohc         ohcf  convertible
##      0.8414      -1.2355       1.3069       1.3437       1.3825       0.5318
##       sedan   curbweight   horsepower   enginesize   highwaympg
##      0.1060       0.4811       0.3312      -0.1015       0.1481
```

Se obtiene que el modelo con el AIC más pequeño de -382.02 es el que cuenta con las variables predictoras eight, five, four, twelve, dohc, dohcv, l, ohc, ohcf, convertible, sedan, curbweight, horsepower, y highwaympg.

Modelo normalizado y transformado

a) Economía de las variables: Ahora, se genera un nuevo modelo después de haber realizado el método de step:

```
R1 =lm(price ~ eight + five + four + twelve + dohc + dohcv +
    l + ohc + ohcf + convertible + sedan + curbweight + horsepower +
    highwaympg, data = df_final)
R1
```

```
##
## Call:
## lm(formula = price ~ eight + five + four + twelve + dohc + dohcv +
##     l + ohc + ohcf + convertible + sedan + curbweight + horsepower +
##     highwaympg, data = df_final)
##
## Coefficients:
## (Intercept)        eight         five         four       twelve         dohc
##    -0.19145      2.22406     -0.60656     -1.26982      1.60873      0.75365
##       dohcv            l          ohc         ohcf  convertible        sedan
##    -1.02028      1.12514      1.22155      1.25898      0.51239      0.11269
##  curbweight   horsepower   highwaympg
##     0.42332      0.26754      0.08799
```

Se observa que se obtienen los coeficientes para cada variable. Antes eran 17 variables con 17 coeficientes (con datos no normalizados), pero al momento de hacer la economía de las variables se obtienen 14 variables con 14 coeficientes (utilizando valores normalizados).

Con esto se puede obtener la ecuación para la regresión lineal múltiple como:

```
b0 = R1$coefficients[1]
b1 = R1$coefficients[2]
b2 = R1$coefficients[3]
b3 = R1$coefficients[4]
b4 = R1$coefficients[5]
b5 = R1$coefficients[6]
```

```
b6 = R1$coefficients[7]
b7 = R1$coefficients[8]
b8 = R1$coefficients[9]
b9 = R1$coefficients[10]
b10 = R1$coefficients[11]
b11 = R1$coefficients[12]
b12 = R1$coefficients[13]
b13 = R1$coefficients[14]
b14 = R1$coefficients[15]
b15 = R1$coefficients[16]
b16 = R1$coefficients[17]
b17 = R1$coefficients[18]

cat("Precio = ", b0, "+", b1, "eight", b2, "five", b3, "four","+",b4,"twelve","+",b5, "dohc",b6, "dohcv
```

```
## Precio =  -0.1914492 + 2.22406 eight -0.60656 five -1.26982 four + 1.608731 twelve + 0.7536543 dohc +
```

b) Significación global (Prueba de F para el modelo normalizado): *Hipótesis*

- Sobre el modelo (significación global):

$H_0 : \beta 1 + \beta 2 = ... = \beta_k = 0$

$H_1 : al\ menos\ un\ \beta 1 \neq 0$

$\alpha = 0.03$

```
summary(R2)
```

```
##
## Call:
## lm(formula = price ~ eight + five + four + six + three + twelve +
##     dohc + dohcv + l + ohc + ohcf + convertible + sedan + curbweight +
##     horsepower + enginesize + highwaympg, data = df_final)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.12795 -0.23310  0.00979  0.16627  1.41414
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.06215    0.28710  -0.216  0.82884
## eight        2.21763    0.40250   5.510 1.18e-07 ***
## five        -0.84228    0.35024  -2.405  0.01716 *
## four        -1.55438    0.32395  -4.798 3.26e-06 ***
## six         -0.20358    0.34256  -0.594  0.55303
## three       -0.86513    0.52658  -1.643  0.10208
## twelve       1.57306    0.51901   3.031  0.00278 **
## dohc         0.87625    0.18482   4.741 4.20e-06 ***
## dohcv       -1.21635    0.48728  -2.496  0.01342 *
## l            1.33359    0.22309   5.978 1.12e-08 ***
## ohc          1.36681    0.17473   7.822 3.70e-13 ***
## ohcf         1.40800    0.18902   7.449 3.36e-12 ***
## convertible  0.52346    0.16813   3.113  0.00214 **
## sedan        0.10825    0.05609   1.930  0.05514 .
## curbweight   0.46169    0.08650   5.337 2.71e-07 ***
```

```
## horsepower   0.32195    0.07897   4.077 6.74e-05 ***
## enginesize  -0.06212    0.09668  -0.643  0.52130
## highwaympg   0.14884    0.07913   1.881  0.06154 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3788 on 187 degrees of freedom
## Multiple R-squared:  0.8685, Adjusted R-squared:  0.8565
## F-statistic: 72.65 on 17 and 187 DF,  p-value: < 2.2e-16
```

Interpretación:

- Significación global

Teniendo: F = 71.76 Valor p = 2.2e-16

El valor p es muy pequeño, mucho menor a *alpha*. Por ende, se rechaza H_0.

c) Significación individual (Prueba de t de student para $beta_i$) : *Hipótesis*

- Sobre las β_i (significación individual)

$H_0 : \beta_i = 0$

$H_1 : \beta_i \neq 0$

Interpretación

- Significación individual

Ignorando el intercepto, y teniendo en cuenta las variables predictoras con t* con valores muy grandes y diferentes de cero tanto positiva como negativamente, se puede concluir que se está muy lejos de la media, por lo tanto todas las variables seleccionadas son significativas. Por lo tanto se rechaza H_0.

d) Variación explicada por el modelo (coeficiente de determinación Tomando el Adjusted R-squared de 0.855, se infiere que el modelo explica el 85.5% del precio, a partir de las variables predictoras.

5. Validez del modelo (Utilizando pruebas de hipótesis de medias)

a) Normalidad de los residuos (prueba de Shapiro-Wilk): H_0 : Los datos provienen de una población normal.

H_1 : Los datos no provienen de una población normal.

Regla de decisión:

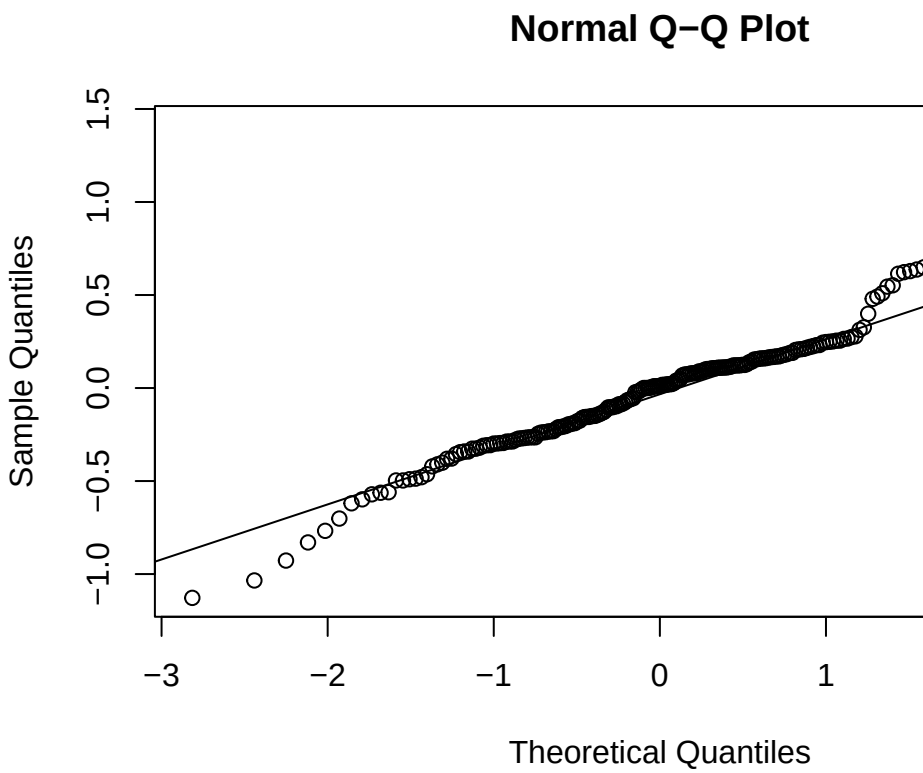
- Se rechazará H_0 si p < *alpha*

```
resultado_shapiro <- shapiro.test(df_final$curbweight)
resultado_shapiro
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df_final$curbweight
## W = 0.9817, p-value = 0.009049
```

- Conclusión: Se observa que el valor p es mayor que *alpha* = 0.03, por lo tanto no se rechaza H_0 y se asume que los datos provienen de una población normal.

```r
qqnorm(R2$residuals)
qqline(R2$residuals)
```



Gráfica de normalidad de los residuos

Se confirma en la gráfica que hay normalidad debido a que la línea en su mayoría va recta sin desplazarse hacia abajo o hacia arriba.

b) Verificación de media cero (prueba t de student): *Hipótesis*

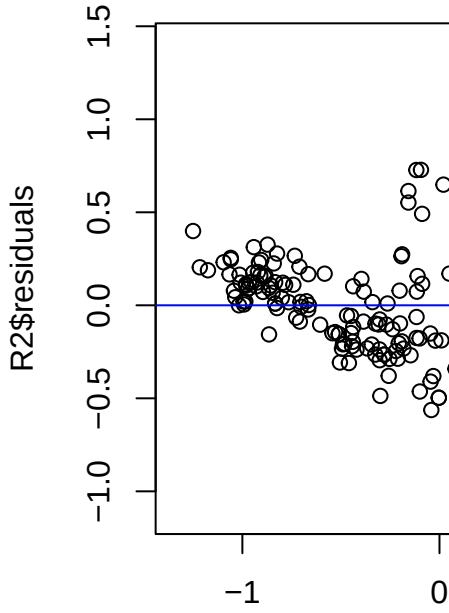
- $H_0 : \mu = 0$
- $H_1 : \mu \neq 0$
- Regla de decisión: Se rechazará H_0 si p $<$ *alpha*

```r
t.test(R2$residuals)
```

```
##
##  One Sample t-test
##
## data:  R2$residuals
## t = 6.7983e-16, df = 204, p-value = 1
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -0.04993757  0.04993757
## sample estimates:
##    mean of x
## 1.721845e-17
```

- Conclusión: No se rechaza la hipótesis nula H_0 ya que el valor de p es mayor que *alpha*, por lo tanto se asume que la media es de 0

```r
plot(R2$fitted.values,R2$residuals)
abline(h=0, col="blue")
```



c) Homocedasticidad o independencia (análisis de gráficos de los residuos):

Se observa que hay homocedasticidad y no hay un sesgo o comportamiento, por lo que es simétrico, y se asume independencia.

Conclusión A lo largo de este reporte se hicieron análisis tanto gráficos como numéricos. Iniciando por generar el resumen de las medidas estadísticas de las variables numéricas, notando hacia donde se agrupan la mayoría de los valores, y observando en las boxplots que hay muchos valores atípicos en la mayoría de las variables numéricas excepto en la variable de curbweight. Por lo que al momento de seleccionar las variables útiles para el análisis: "curbweight", "horsepower", "carwidth", "enginesize", "citympg", "highwaympg", "symboling", "cylindernumber", "price", "enginetype", "carbody", se hizo una normalización de las mismas usando el método de Z-score. Por otra parte, como se consideran relevantes algunas variables categóricas, se transformaron en variables dummy que están en un rango de cero y uno. Estas variables a tratar fueron seleccionadas ya que tienen un impacto en la correlación positiva o negativa con respecto al precio del vehículo. Se dejaron de lado otras variables que o se repetían mucho o eran insignificantes para el análisis deseado. Sobre el análisis realizado, se observa que el kilometraje en ciudad y carretera tiene un impacto negativo en el precio, es decir, que cuanto mayor es este kilometraje, el precio disminuye. Por otra parte, tomando en cuenta las variables como el peso en vacío del vehículo, los caballos de potencia, el tamaño del vehículo y del motor, el precio aumenta conforme estas variables aumentan. Esto puede significar mucho para ayudar al cliente a obtener los precios de sus vehículos al establecer su compañía en el mercado estadounidense.

Más adelante se realizaron modelos de regresión lineal múltiple para generar un nuevo modelo que pudiera predecir el precio de un vehículo a partir de variables predictoras, para encontrar este modelo con las variables útiles para la predicción se utilizó el método "mixto" y se encontró un modelo con un coeficiente de determinación de 0.855, lo que indica que es un buen modelo para ayudar al cliente a obtener los precios de sus vehículos al establecer su compañía en Estados Unidos. Sin embargo, para validar este modelo, se realizaron las pruebas de hipótesis de medias, y la verificación del modelo. Todas las pruebas fueron aprobadas, por lo que se valida el modelo para su aplicación en la situación planteada.