

Análisis - Compra-venta de coches usados en UK

Marta Gómez / Juan Fco Nieto

12/17/2021

Contents

1 Descripción del dataset	1
1.1 Objetivos y descripción del dataset original	1
2 Integración y selección de los datos de interés a analizar	2
2.1 Descripción de las variables	3
3 Limpieza de los datos	4
3.1 Análisis de valores nulos o vacíos	4
3.2 Análisis de valores atípicos	4
4 Análisis de los datos.	8
4.1 Selección de los grupos de datos que se quieren analizar/comparar (planificación de los análisis a aplicar).	8
4.2 Comprobación de la normalidad y homogeneidad de la varianza.	11
4.3 Análisis Estadístico	14
5 Análisis 2. Caso práctico compra-venta	22
6 Conclusiones	24
7 Exportación del código en R y de los datos producidos.	24
8 Contribución	24
0.0.0.0.1 Cargar librerías	

1 Descripción del dataset

1.1 Objetivos y descripción del dataset original

El presente proyecto tiene como objetivo final el análisis del mercado de segunda mano de coches de Reino Unido que ayude en la toma de decisiones tanto de un comprador como de un vendedor.

Para poder llevar a cabo dicho objetivo se procede a la integración, limpieza, validación y análisis de un conjunto de datasets de coches usados en Reino Unido creados en Julio 2020 por el usuario ‘Aditya’ (<https://www.kaggle.com/adityadesai13>) a través de web scraping de portales de compraventa británicos. El objetivo inicial del usuario era la creación de un modelo de regresión lineal de coches usados para hacer predicciones sobre la variable target “price”, interpretándose como un análisis del precio de mercado.

Citando al usuario:

“I collected the data to make a tool to predict how much my friend should sell his old car for compared to other stuff on the market, and then just extended the data set. Then made a more general car value regression model.”

El resultado del web scraping son 13 ficheros individuales tipo csv, entre los que seleccionamos un total de 9 ficheros, identificados por el nombre del fabricante, con las características de distintos modelos tales como el año, tipo de combustible, tipo de motor, kilometraje, precio actual, etc... Dichos ficheros se encuentran en el siguiente enlace: <https://www.kaggle.com/adityadesai13/used-car-dataset-ford-and-mercedes>. El resto de ficheros (“cclass.csv”, “focus.csv”, “unclean cclass.csv” y “unclean focus.csv”) no se tienen en cuenta para el presente análisis.

Para facilitar el estudio procedemos a la integración de los nueve ficheros de interés en un único fichero tipo csv al que llamaremos aprox100KUsedCars.csv. En dicho archivo incluiremos los campos de cada fichero más el campo “manufacturer” con el nombre del fabricante que extraeremos del nombre de cada fichero individual.

2 Integración y selección de los datos de interés a analizar

Para la integración de los ficheros hemos utilizado el script Ruby (ruby integration.rb) localizado en la carpeta “integration” en el enlace GitHub cuya ejecución crea en nuestra raíz del proyecto el fichero aprox100KUsedCars.csv. De este modo nos encontramos con un dataset con un total de 99187 filas y 10 columnas que representan 99187 ofertas en portales de compraventa de coches usados en Reino Unido. En la siguiente tabla se muestra un ejemplo del tipo de datos.

```
# Carga de los datos
used_cars <- read.csv("aprox100KUsedCars.csv", stringsAsFactors = TRUE)

sample_n(used_cars,10) %>% knitr::kable()
```

manufacturer	model	year	price	transmission	mileage	fuelType	tax	mpg	engineSize
toyota	Yaris	2016	7298	Manual	34586	Petrol	30	58.0	1.3
bmw	X2	2018	24495	Semi-Auto	13147	Diesel	150	58.9	2.0
bmw	4 Series	2014	18062	Semi-Auto	49745	Diesel	165	50.4	3.0
merc	A Class	2017	17500	Manual	24000	Petrol	145	49.6	1.6
vw	Touran	2020	24480	Semi-Auto	341	Diesel	145	47.9	2.0
ford	Kuga	2017	14292	Manual	19674	Petrol	160	45.6	1.5
merc	A Class	2016	13499	Automatic	15722	Diesel	20	74.3	1.5
vauxhall	Insignia	2015	6969	Manual	51276	Diesel	0	76.3	2.0
vauxhall	Zafira Tourer	2016	8495	Manual	58825	Petrol	205	42.2	1.4
hyundi	I10	2018	7560	Manual	8623	Petrol	145	60.1	1.0

- Tabla 1. Ejemplo del tipo de datos almacenada en el dataset.

2.1 Descripción de las variables

A continuación mostramos un resumen del tipo de variables presentes en el dataset que constituyen los datos de interés a analizar en el que podemos observar la presencia de dos tipos principales de variables: categóricas (factor) y numéricas (integer o numeric).

```
# Tabla resumen del tipo de variables que conforman el dataset
tb_var <- sapply(used_cars, class)
knitr::kable(data.frame(variables = names(tb_var), clase = as.vector(tb_var)))
```

variables	clase
manufacturer	factor
model	factor
year	integer
price	integer
transmission	factor
mileage	integer
fuelType	factor
tax	integer
mpg	numeric
engineSize	numeric

- Tabla 2. Tipo de variables

2.1.1 Variables categóricas (factor)

- manufacturer: Fabricante del automóvil. Variable categórica nominal con 9 categorías (Niveles) diferentes.
- model: Modelo del automóvil. Variable categórica nominal con 195 categorías diferentes.
- transmission: Tipo de transmisión. Variable categórica nominal con 4 categorías: manual, automática, semiautomática y otras.
- fuelType: Tipo de combustible. Variable categórica nominal con 5 categorías: diesel, eléctrico, híbrido, gasolina y otros.

```
sk <- skim(used_cars)
sk %>% yank('factor') %>% select(c(skim_variable, n_unique)) %>% rename(Variable=skim_variable, Niveles=n_unique)
```

Variable	Niveles
manufacturer	9
model	195
transmission	4
fuelType	5

- Tabla 3. Resumen de las variables categóricas y sus posibles valores

2.1.2 Variables numéricas

- year: Año de matriculación del coche. Variable numérica discreta.

- price: Precio en Libras que se colocó en el portal de compraventa a fecha Julio 2020. Variable numérica continua.
- mileage: kilométraje. Millas que el coche ha recorrido desde su puesta en funcionamiento. (En España utilizamos Kilometraje, porque medimos esta distancia en kilómetros). Variable numérica continua.
- tax: Impuesto de circulación (en Libras). Dependiendo de los años del vehículo, emisión de gases (sobre todo) y otros factores este impuesto varía. Variable numérica continua
- mpg: Consumo de combustible del vehículo en millas por galón. Variable numérica continua
- engineSize: Tamaño del motor en litros. Variable numérica continua.

```
sk %>% yank('numeric') %>% select(c(skim_variable, mean, sd, p25, p50, p75)) %>% rename(Variable=skim_v
```

Variable	Media	Desviación Típica	Q1	Q2/Mediana	Q3
year	2017.088	2.124	2016.0	2017.0	2019.0
price	16805.348	9866.773	9999.0	14495.0	20870.0
mileage	23058.914	21148.524	7425.0	17460.0	32339.0
tax	120.300	63.151	125.0	145.0	145.0
mpg	55.167	16.139	47.1	54.3	62.8
engineSize	1.663	0.558	1.2	1.6	2.0

- Tabla 4. Análisis descriptivo de las variables numéricas

3 Limpieza de los datos

3.1 Análisis de valores nulos o vacíos

Se comprueba la no existencia de ceros o elementos vacíos en el dataset mediante la ejecución de la función `is_blank` creada para tal propósito.

```
# Se crea la función is_blank que devuelve la presencia (TRUE) o no (FALSE) de valores vacíos ("" o null)

is_blank <- function(x){
  return (any(x=="") || anyNA(x))
}
used_cars %>% summarise_all(.funs=c('is_blank'), )
```

```
## manufacturer model year price transmission mileage fuelType tax mpg
## 1 FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## engineSize
## 1 FALSE
```

3.2 Análisis de valores atípicos

La representación gráfica en forma de diagrama de barras (boxplot) identifica la presencia de valores extremos superiores en todas las variables numéricas y sólo en algunas se observan también valores extremos inferiores.

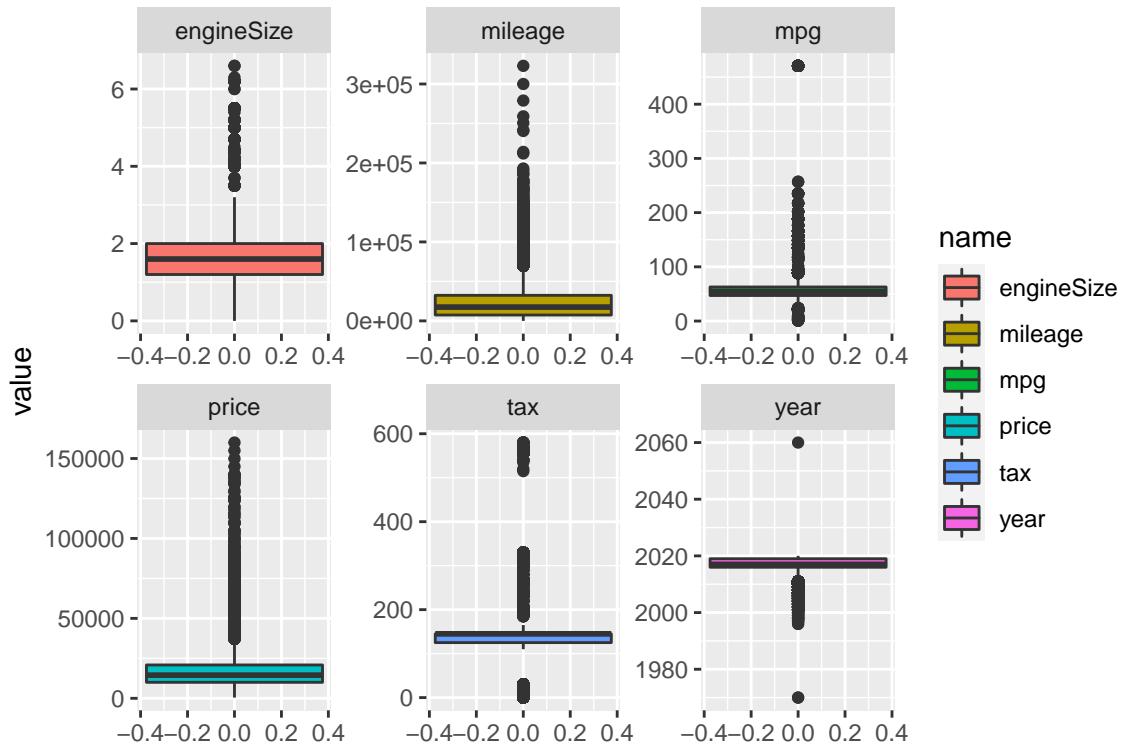
```
# Boxplot para cada una de las variables numéricas (vars)
vars <- c("year", "price", "mileage", "tax", "mpg", "engineSize")

used_cars %>% select(vars) %>% pivot_longer(cols=vars, values_drop_na = TRUE) %>% ggplot(.) + facet_wrap
```

```

## Note: Using an external vector in selections is ambiguous.
## i Use `all_of(vars)` instead of `vars` to silence this message.
## i See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This message is displayed once per session.

```



- Figura 1. Gráfico mostrando los valores extremos de las variables numéricas

3.2.1 Identificación y tratamiento de outliers

Un análisis más profundo nos permite identificar aquellos valores extremos y valorar su tratamiento en función de la información almacenada en dicha variable.

Para ello creamos la función `get_outlier` que toma como referencia el método de la diferencia intercuartil (IQR) (ver <https://www.r-bloggers.com/2020/01/how-to-remove-outliers-in-r/>)

```

# Creación de la función get_outliers para la identificación de los valores extremos de cada una de las
get_outliers <- function(x){
  q1 = quantile(x, c(0.25))
  q3 = quantile(x, c(0.75))
  iqr = q3-q1
  result = sapply(x, function(y){
    if(y < q1 - 1.5*iqr){
      y
    }else if(y > q3+1.5*iqr){
      y
    }else{
      NA
    }
  })
}

```

```

    }
}
return(list(result, q1 - 1.5*iqr, q3+1.5*iqr))
}

```

La ejecución de la función `get_outliers` nos permite identificar los valores extremos para cada una de las variables de interés, cuyo resultado mostramos en forma de tabla junto con el rango inferior, rango superior y el número total de outliers identificados y tres ejemplos para cada una de las variables.

```

# Ejecución de la función get_outliers y su resultado en forma de tabla
final <- list()
outliers_analysis <- used_cars %>% select(vars) %>% sapply(., get_outliers)
for(i in vars){
  ej <- na.omit(outliers_analysis[,i][[1]])

  final <- cbind(final, c(paste(sample(ej, 3), sep=" ", collapse=" / "), outliers_analysis[,i][[2]], o
}
colnames(final) = vars
rownames(final) = c("Ejemplos", "Inferior", "Superior", "Número total")
data.frame(final) %>% knitr::kable()

```

	year	price	mileage	tax	mpg	engineSize
Ejemplos	2010 / 2005 / 2003	46220 / 37444 / 48896	73856 / 82685 / 126000	20 / 235 / 30	88.3 / 470.8 / 141.2	4 / 4 / 6.6
Inferior	2011.5	-6307.5	-29946	95	23.55	-
Superior	2023.5	37176.5	69710	175	86.35	3.2
Número total	1737	3669	3902	28815	939	650

- Tabla 5. Valores extremos.

A continuación detallamos el tratamiento de los valores extremos en función del tipo de variable así como las inconsistencias encontradas tras su análisis.

- Year: El año de matriculación no puede ser superior al año de la recogida de los datos (Julio 2020), por lo que eliminaremos del dataset aquellos coches con fecha de matriculación superior al 2020. Por otra parte, aunque el análisis muestra como valores extremos los coches matriculados antes del 2011, consideraremos estos coches minoritarios pero válidos y los mantendremos en el dataset.

```

# Eliminamos los coches matriculados más tarde del 2020
original_used_cars <- used_cars
used_cars <- used_cars %>% filter(year <= 2020)

```

- Price: El análisis de valores extremos pone en evidencia una gran dispersión entre los distintos precios de los coches, con un rango inferior negativo. Dado que un valor negativo en el precio sería un claro error lo comprobamos y observamos que no existen. Sin embargo vemos que hay un porcentaje minoritario de coches, probablemente de alta gama, con un precio muy elevado sobre la media que en principio son reconocidos como valores extremos. Sin embargo, los consideramos válidos.

```
# Comprobamos que no hay coches con precio negativo
used_cars_neg_price <- used_cars %>% filter(price <= 0)
nrow(used_cars_neg_price)
```

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

- Mileage: de igual manera no podemos aceptar kilometraje negativo, por lo que lo comprobamos y observamos que no existen. Al igual que en la variable price, hay una gran dispersión en esta variable. Dado que se trata de compra-venta de coches de segunda mano asumimos una gran variedad de ofertas de coches y los consideramos válidos.

```
used_cars_neg_mil <- used_cars %>% filter(mileage <= 0)
nrow(used_cars_neg_mil)
```

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

- Tax: El diagrama de barras muestra una clara estratificación en los impuestos en el que observamos tres tramos. Consideramos válidos los valores extremos mostrados en el análisis. Sin embargo eliminamos los valores iguales a 0.

```
used_cars <- used_cars %>% filter(tax > 0)
```

- Mpg: debemos tener en cuenta que nuestro dataset contiene tanto coches híbridos como eléctricos por lo que habrá que mantener los datos cuyo mpg sea bajo o 0. Se podría hacer una manipulación distinta de los outliers para los diferentes segmentos de fuelType pero para no aumentar complejidad la mantendremos tal como está.
- engineSize: de igual manera un motor eléctrico deberá tener volumen 0, otro sería considerado una inconsistencia. Comprobamos por tanto que aquellos coches con volumen 0 sean eléctricos.

```
# Comprobación del tipo de coche con tamaño de EngineSize =0
used_cars <- used_cars %>% filter(engineSize != 0.0 | (engineSize == 0.0 & fuelType == "Electric"))
```

El análisis de los valores extremos delata una gran dispersión de los datos esperable dado el tipo de dataset seleccionado.

Tras el análisis de outliers el conjunto de datos numérico que tenemos tiene la siguiente estadística descriptiva:

```
sk2 <- skim(used_cars)
sk2 %>% yank('numeric') %>% select(c(skim_variable, mean, sd, p25, p50, p75, p0, p100)) %>% rename(Vari...
```

Variable	Media	Desviación Típica	Q1	Q2/Mediana	Q3	Mínimo	Máximo
year	2017.184	2.129	2016.0	2017.0	2019.0	1970.0	2020.0
price	17262.101	9985.671	10495.0	14995.0	21399.0	450.0	159999.0
mileage	22184.935	20917.333	6868.5	16330.0	31131.0	1.0	323000.0
tax	128.445	56.655	125.0	145.0	145.0	10.0	580.0
mpg	53.608	12.484	45.6	54.3	60.1	0.3	470.8
engineSize	1.687	0.558	1.3	1.6	2.0	0.0	6.6

- Tabla 5. Estadística descriptiva de las variables numéricas tras la limpieza

4 Análisis de los datos.

Una vez realizada la integración y limpieza de los datos vamos a proceder al análisis de los datos. Para ello vamos a realizar dos análisis completos que usaremos como ejemplos de tipo de consultas que se pueden resolver con este dataset.

En el primer análisis mostramos una serie de pasos que tienen como finalidad ayudar en la toma de decisión de un cliente que está interesado en comprarse un coche con una antigüedad igual o inferior a 5 años junto con otros requisitos que detallaremos más adelante

En el segundo análisis realizamos un modelo de regresión lineal para identificar aquellos coches con un valor por debajo de su precio de mercado para comprarlos y revenderlos a un precio de mercado.

- 1. Análisis para la toma de decisión de la compra de un coche con 5 años de antigüedad.

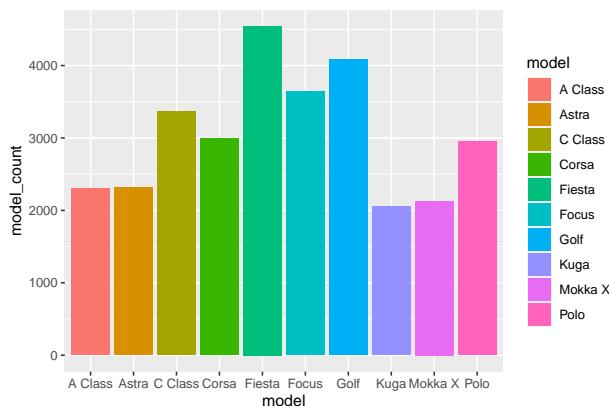
Un cliente quiere comprarse un coche de segunda mano. Para ello nos contacta para que le hagamos un análisis del mercado de segunda mano. Sus principales requisitos son 1) que sea un coche con una antigüedad igual o inferior a 5 años, 2) con un precio de hasta 6000 £, 3) que tenga muchas opciones de compra, es decir, que si se decide por un modelo concreto tenga donde elegir y 4) con una buena relación calidad-precio. El fabricante y/o modelo de coche, tipo de transmisión y tipo de combustible no es en principio importante para el cliente siempre y cuando cumpla con los requisitos anteriores.

A continuación detallamos los pasos realizados para dicho análisis de mercado que tendrá como finalidad recomendar al cliente un coche que cumpla con sus requisitos.

4.1 Selección de los grupos de datos que se quieren analizar/comparar (planificación de los análisis a aplicar).

1. Dado que el cliente quiere tener amplia oferta de compra exploramos los 10 modelos de coches con mayor oferta y con una antigüedad igual o inferior a cinco años y mostramos el gráfico visualmente.

```
# Selección de los 10 modelos más frecuentes de coches matriculados a partir del 2015 hasta la actualidad
top_10_used_cars <- used_cars %>% filter(year >= 2015) %>% group_by(manufacturer, model) %>% summarize(
  n = n(),
  .by_group = TRUE)
top_10_used_cars[,2:3] %>% ggplot(.) + geom_col(aes(model, n), fill=model)
```



- Figura 2. Modelos de coches con mayor oferta en el mercado de segunda mano en Julio 2020

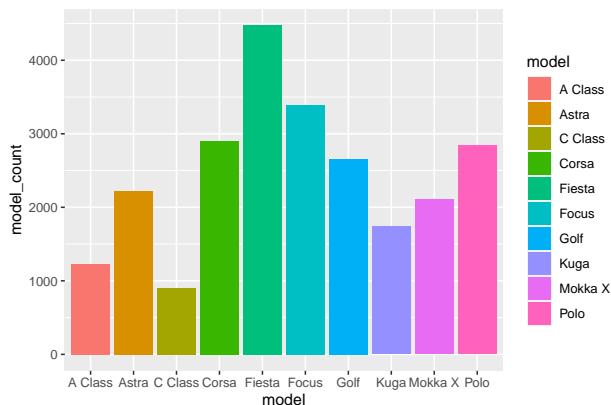
Una vez obtenida dicha información seleccionamos los modelos en nuestro dataset original aplicando los requisitos de antigüedad y precio y mostramos el resultado mediante una serie de gráficos.

```
# Filtrado de los coches que cumplen los requisitos
#top_10_cars <- used_cars %>% filter((manufacturer %in% top_10_used_cars$manufacturer) & (model %in% top_10_models))

top_10_cars <- used_cars %>% filter((manufacturer %in% top_10_used_cars$manufacturer) & (model %in% top_10_models))

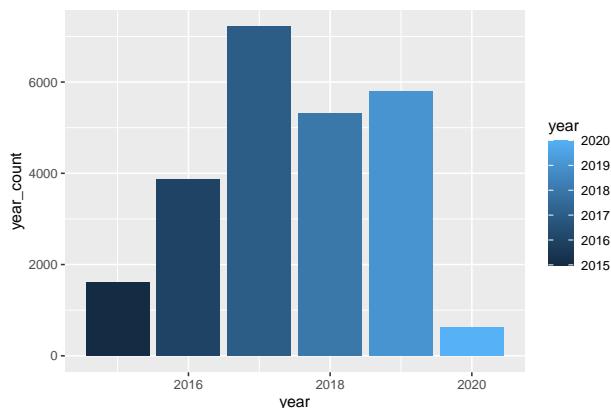
par(mfrow=c(3,3))

# 1. Gráfico que muestra el número de coches por modelo
car_model <- top_10_cars %>% group_by(manufacturer, model) %>% summarize(model_count=n()) %>% arrange(-model)
car_model[,2:3] %>% ggplot(.) + geom_col(aes(model, model_count, fill=model))
```



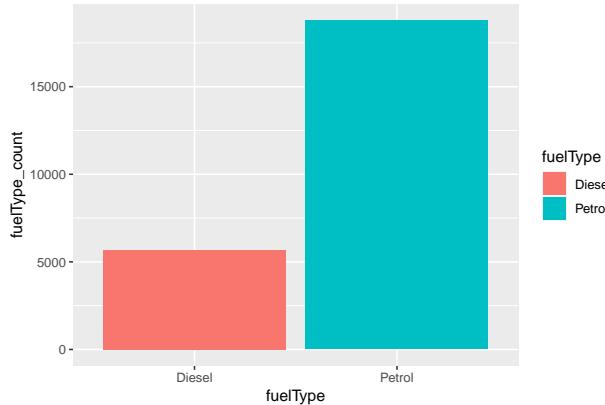
```
# 2. Gráfico que muestra el número de coches por año
car_year <- top_10_cars %>% group_by(manufacturer, year) %>% summarize(year_count=n()) %>% arrange(-year)

car_year[,2:3] %>% ggplot(.) + geom_col(aes(year, year_count, fill=year))
```

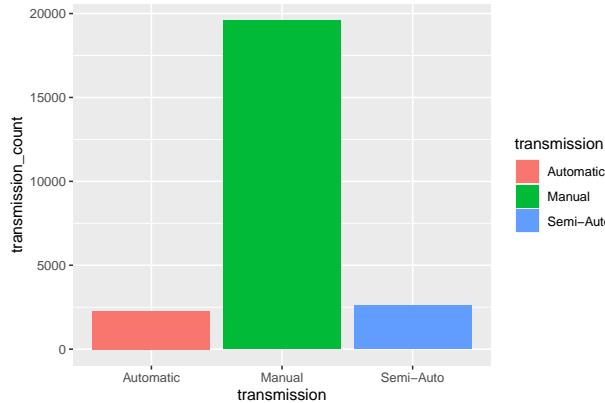


```
# 3. Gráfico que muestra el número de coches por fuelType
car_fuel <- top_10_cars %>% group_by(manufacturer, fuelType) %>% summarize(fuelType_count=n()) %>% arrange(-fuelType)

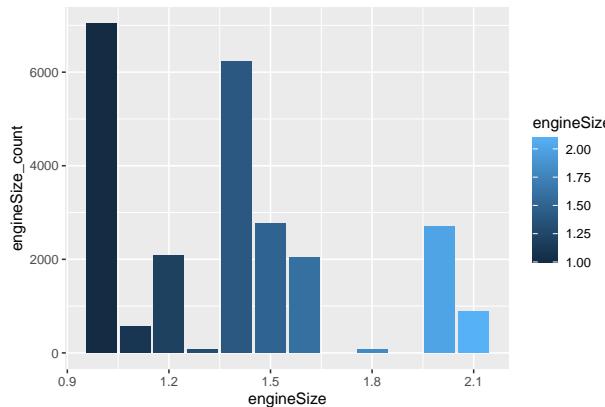
car_fuel[,2:3] %>% ggplot(.) + geom_col(aes(fuelType, fuelType_count, fill=fuelType))
```



```
# 4. Gráfico que muestra el número de coches por tipo de transmisión
car_tras <- top_10_cars %>% group_by(manufacturer, transmission) %>% summarize(transmission_count=n())
car_tras[,2:3] %>% ggplot(.) + geom_col(aes(transmission, transmission_count, fill=transmission))
```



```
# 5. Gráfico que muestra el número de coches por tipo de engineSize
car_motor <- top_10_cars %>% group_by(manufacturer, engineSize) %>% summarize(engineSize_count=n()) %>%
car_motor[,2:3] %>% ggplot(.) + geom_col(aes(engineSize, engineSize_count, fill=engineSize))
```



- Figura 3. Resumen de las características de los coches con mayor oferta en el mercado

```
# Tabla estadística descriptiva de las variables cuantitativas
sk3 <- skim(top_10_cars)
sk3 %>% yank('numeric') %>% select(c(skim_variable, mean, sd, p25, p50, p75)) %>% rename(Variable=skim_
```

Variable	Media	Desviación Típica	Q1	Q2/Mediana	Q3
year	2017.479	1.261	2017.00	2017.0	2019.0
price	12847.554	3475.807	10000.00	12450.0	15500.0
mileage	18280.471	11705.361	9334.25	16257.0	26078.0
tax	121.491	50.178	125.00	145.0	145.0
mpg	55.738	9.159	48.70	55.4	61.4
engineSize	1.382	0.337	1.00	1.4	1.5

- Tabla 6. Estadística descriptiva de las variable cuantitativas de los coches con mayor oferta.

```
# Boxplot para cada una de las variables numéricas (vars)
vars <- c("year", "price", "mileage", "tax", "mpg", "engineSize")

top_10_cars %>% select(vars) %>% pivot_longer(cols=vars, values_drop_na = TRUE) %>% ggplot(.) + facet_w
```

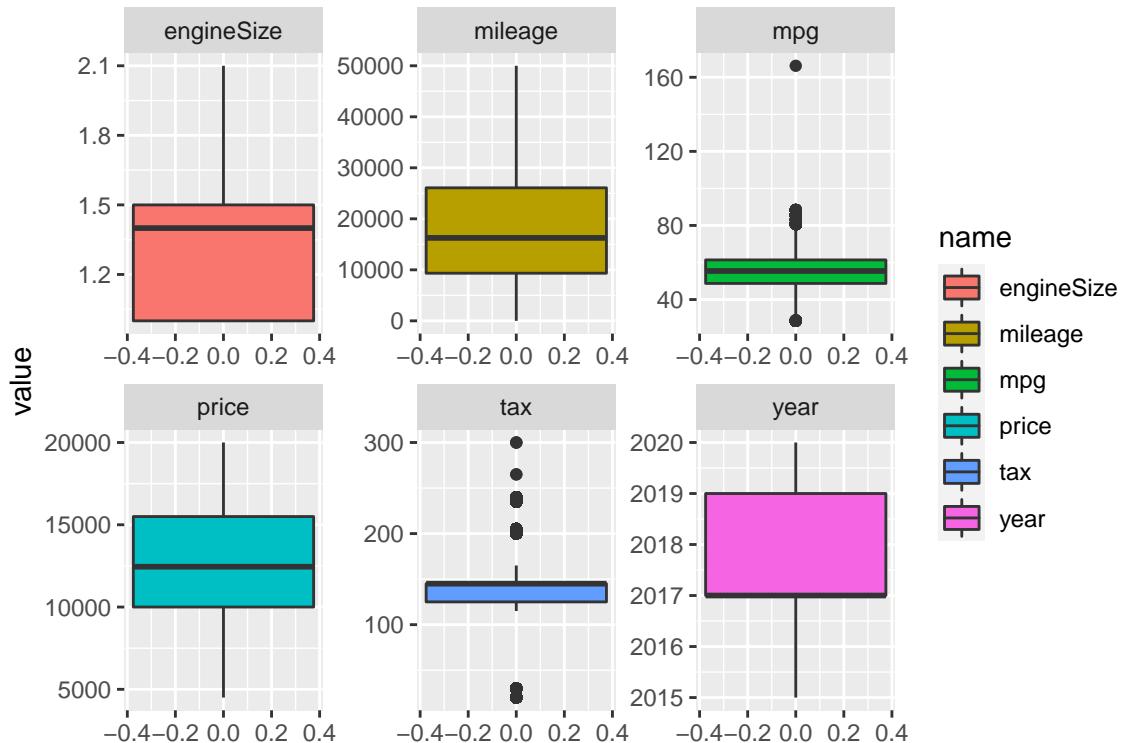


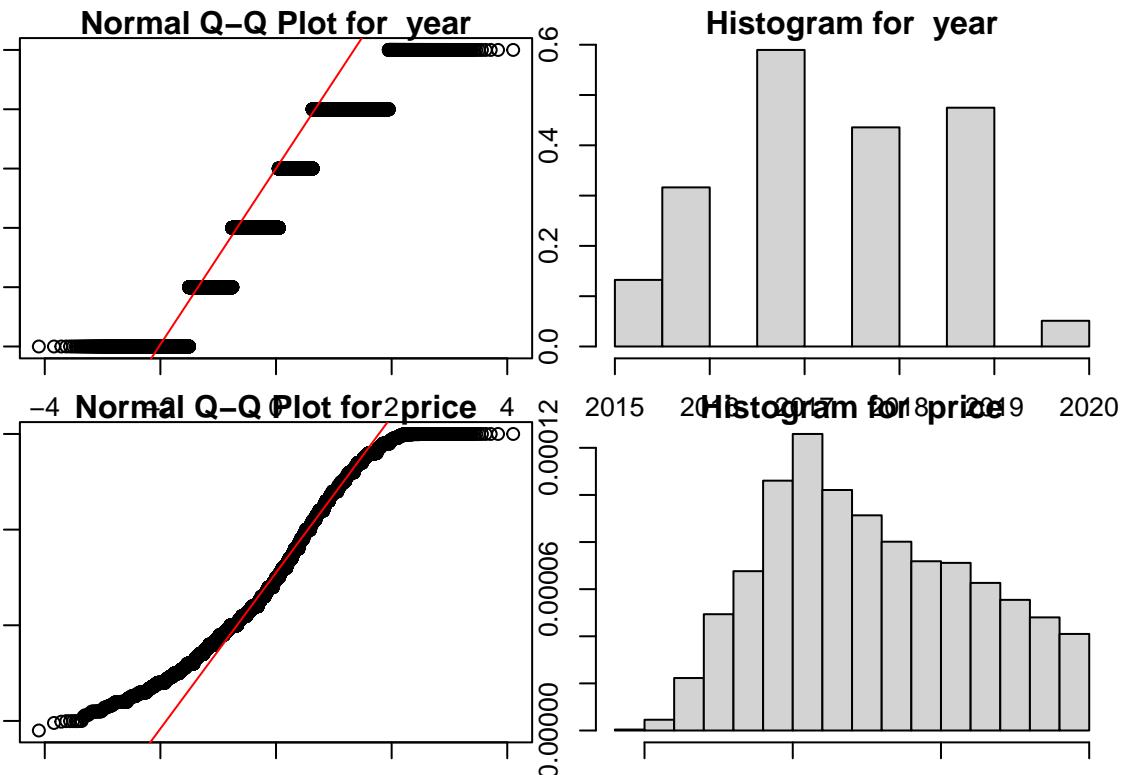
Figura 4. Diagrama de barras de las características de los coches con mayor oferta

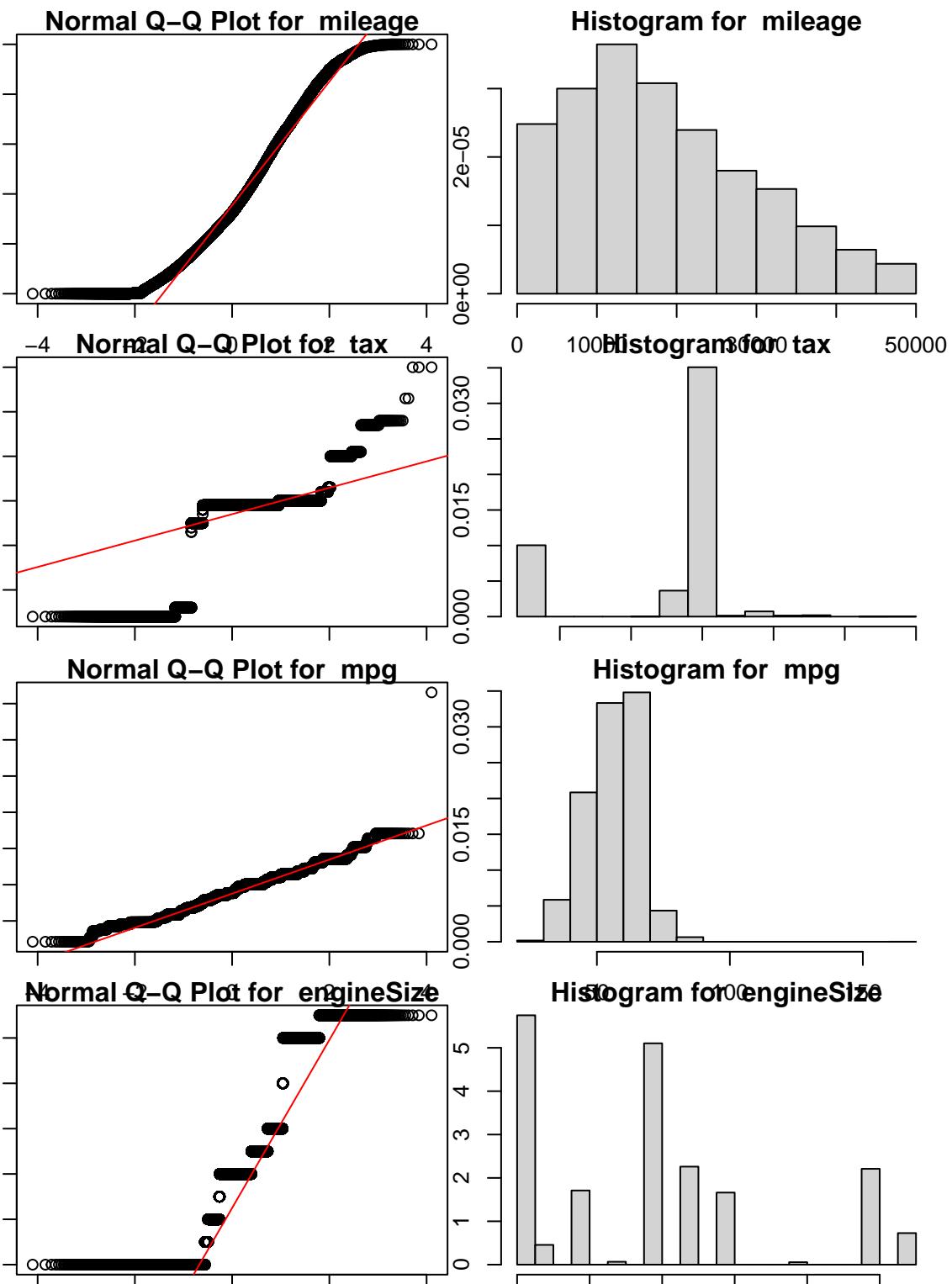
4.2 Comprobación de la normalidad y homogeneidad de la varianza.

Antes de aplicar ningún análisis estadístico comprobamos la normalidad y homogeneidad de los datos. Para ello elegimos dos estrategias, una basada en la representación gráfica de las variables numéricas y la otra en el test de Lilliefors..

- 1. Gráficos Q-Q e Histograma Representamos los datos mediante un histograma y los gráficos de cuantiles teóricos (Gráficos Q-Q) y observamos que, salvo la variable “mileage” (kilometraje), todas las variables se alejan bastante de una distribución normal.

```
par(mfrow=c(2,2), par(mar=c(1,1,1,1)))
for(i in 1:ncol(top_10_cars)) {
  if (is.numeric(top_10_cars[,i])){
    qqnorm(top_10_cars[,i],main = paste("Normal Q-Q Plot for ",colnames(top_10_cars)[i]))
    qqline(top_10_cars[,i],col="red")
    hist(top_10_cars[,i],
         main=paste("Histogram for ", colnames(top_10_cars)[i]),
         xlab=colnames(top_10_cars)[i], freq = FALSE)
  }
}
```





- Figura 5. Gráficos Q-Q e Histogramas de las variables numéricas de interés
- 2. Test de Lilliefors para las variables numéricas.

Nuestro dataset, una vez filtrado con los requisitos del cliente tienen un total de 24466 entradas por lo que

aplicamos el test de Lilliefors a cada una de las variables numéricas, el cual asume una media y varianza desconocida.

El resultado es similar al observado visualmente. Todas las variables tienen un valor p ($p\text{-value}$) < 0.05 , por lo que se rechaza la hipótesis nula y las variables no se consideran que siguen una distribución normal.

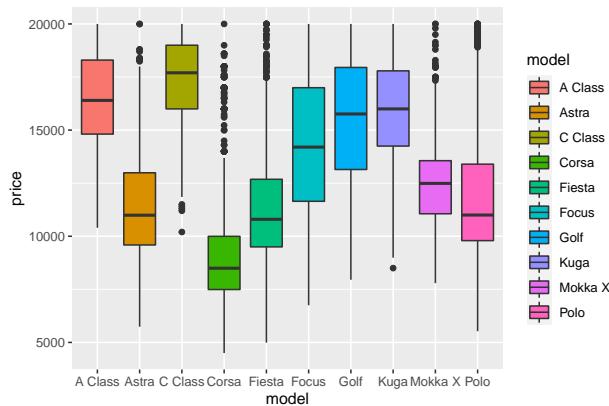
Sin embargo, aplicando el Teorema Central, asumiremos normalidad a la hora de aplicar los métodos estadísticos detallados a continuación.

4.3 Análisis Estadístico

Para ayudar a nuestro cliente a tomar una decisión vamos a aplicar una serie de test estadísticos y su representación gráfica.

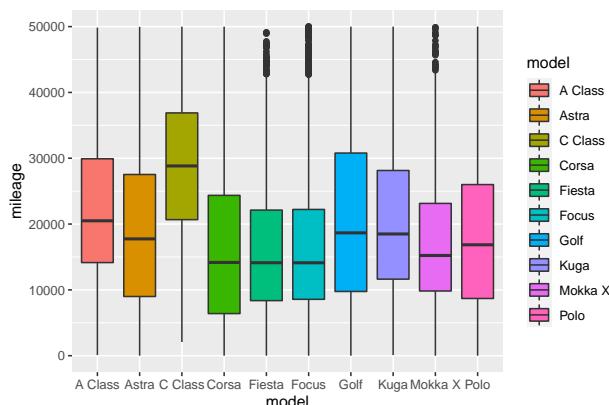
- Comparación precio medio por modelo.

```
# Gráfico del precio medio y la desviación estándar por modelo de coche
ggplot(top_10_cars, aes(x=model, y=price, fill = model)) + geom_boxplot()
```



Del gráfico anterior observamos una gran diferencia del precio medio en función del modelo que nos permite agruparlos en tres categorías en función del precio: alto (A class, C Class), medio (Focus, Golf, Kuga) y bajo (Astra, Corsa, Fiesta, Mokka X y Polo).

```
# Gráfico del kilometraje medio y la desviación estándar por modelo de coche
ggplot(top_10_cars, aes(x=model, y=mileage, fill = model)) + geom_boxplot()
```



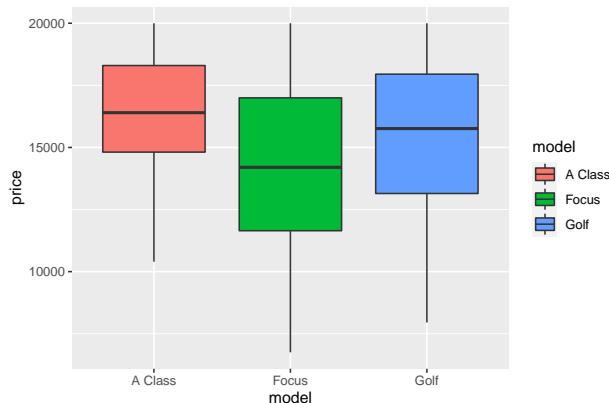
Dado que el cliente quiere un coche de confianza, del anterior análisis seleccionamos tres modelos (A Class, Focus, Golf) que dadas sus características creemos que le pueden interesar al comprador y estudiamos:

- 1. Análisis del impacto del modelo de coche sobre la media de las variables aplicando un análisis de la varianza sobre las variables precio, kilometraje y consumo medio.
- 2. El impacto del kilometraje en el precio del coche aplicando una correlación.
- 3. El impacto del tipo de combustible (petrol vs. diesel) en el precio y kilometraje.

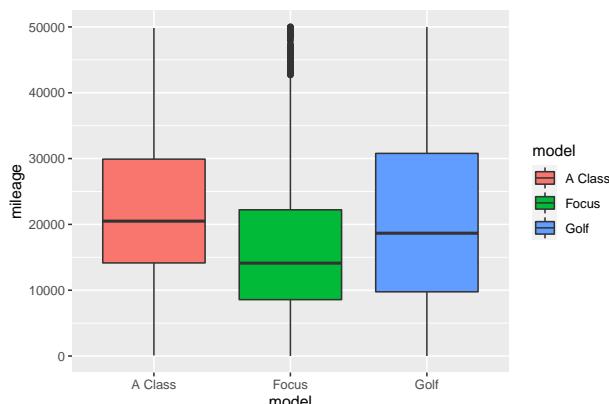
```
# Selección de tres modelos de coches
three_cars <- c('A Class', 'Focus', 'Golf')
final_cars <- top_10_cars %>% filter((model %in% three_cars))
```

- 1. Análisis del impacto del modelo sobre la media de las variables 1) precio, 2) kilometraje, 3) impuesto de circulación, 4) antigüedad media y 5) consumo medio:

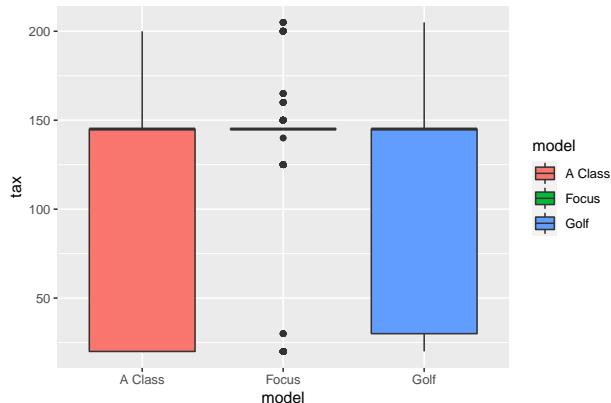
```
# Gráficos
# 1. Precio medio
ggplot(final_cars, aes(x=model, y=price, fill = model)) + geom_boxplot()
```



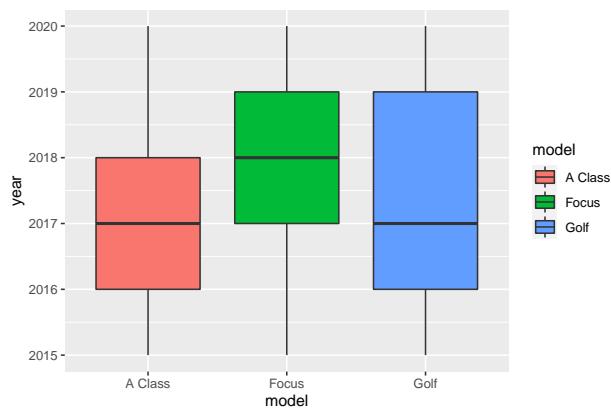
```
# 2. kilometraje medio
ggplot(final_cars, aes(x=model, y=mileage, fill = model)) + geom_boxplot()
```



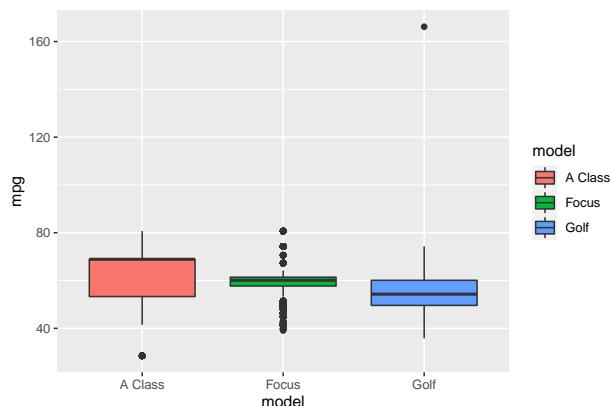
```
# 3. Impuesto de circulación medio
ggplot(final_cars, aes(x=model, y=tax, fill = model)) + geom_boxplot()
```



```
# 4. Antigüedad media
ggplot(final_cars, aes(x=model, y=year, fill = model)) + geom_boxplot()
```



```
# 5. Consumo medio
ggplot(final_cars, aes(x=model, y=mpg, fill = model)) + geom_boxplot()
```



- Figura 7. Impacto del modelo en la media de las variables cuantitativas

El análisis de la varianza para cada variable numérica muestra un alto grado de significancia en función del modelo.

```
# 1. Cálculo de la varianza de un solo factor (ONE-WAY ANOVA) para precio
res.aov_price <- aov(price ~ model, data = final_cars)
# Resumen del análisis
summary(res.aov_price)
```

```
##           Df   Sum Sq  Mean Sq F value Pr(>F)
## model       2 5.047e+09 2.523e+09   310.4 <2e-16 ***
## Residuals  7274 5.913e+10 8.129e+06
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Comparación multiple
TukeyHSD(res.aov_price)
```

```
## Tukey multiple comparisons of means
## 95% family-wise confidence level
##
## Fit: aov(formula = price ~ model, data = final_cars)
##
## $model
##      diff      lwr      upr p adj
## Focus-A Class -2127.4531 -2350.236 -1904.6699 0
## Golf-A Class   -803.5211 -1034.294  -572.7481 0
## Golf-Focus     1323.9320  1150.839  1497.0248 0
```

```
# 2. Cálculo de la varianza de un solo factor (ONE-WAY ANOVA) para mileage
res.aov_mileage <- aov(mileage ~ model, data = final_cars)
# Resumen del análisis
summary(res.aov_mileage)
```

```
##           Df   Sum Sq  Mean Sq F value Pr(>F)
## model       2 4.208e+10 2.104e+10   155.2 <2e-16 ***
## Residuals  7274 9.861e+11 1.356e+08
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Comparación multiple
TukeyHSD(res.aov_mileage)
```

```
## Tukey multiple comparisons of means
## 95% family-wise confidence level
##
## Fit: aov(formula = mileage ~ model, data = final_cars)
##
## $model
##      diff      lwr      upr p adj
## Focus-A Class -5980.719 -6890.511 -5070.928 0.0e+00
## Golf-A Class   -1973.776 -2916.196 -1031.356 2.8e-06
## Golf-Focus     4006.943  3300.075  4713.812 0.0e+00
```

```
# 5. Cálculo de la varianza de un solo factor (ONE-WAY ANOVA) para mpg
res.aov_mpg <- aov(mpg ~ model, data = final_cars)
# Resumen del análisis
summary(res.aov_mpg)
```

```
##          Df Sum Sq Mean Sq F value Pr(>F)
## model      2 65488   32744     399 <2e-16 ***
## Residuals 7274 596878       82
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Comparación multiple
TukeyHSD(res.aov_mpg)
```

```
## Tukey multiple comparisons of means
## 95% family-wise confidence level
##
## Fit: aov(formula = mpg ~ model, data = final_cars)
##
## $model
##        diff      lwr      upr p adj
## Focus-A Class -4.299427 -5.007261 -3.591594 0
## Golf-A Class  -8.538872 -9.272091 -7.805653 0
## Golf-Focus    -4.239445 -4.789401 -3.689489 0
```

- 2. Impacto del kilometraje en el precio del coche en cada uno de los modelos seleccionados.

En este caso estudiamos dicho impacto en cada uno de los modelos por separado por lo que creamos los correspondientes datasets.

```
# Crear un dataset por cada modelo
A_Class <- top_10_cars %>% filter(model=='A Class')
Golf <- top_10_cars %>% filter(model=='Golf')
Focus <- top_10_cars %>% filter(model=='Focus')
```

La correlación entre el kilometraje y el precio es mayor para el coche Class A (-0.65), que indica que a menor kilometraje mayor precio. Para los modelos Golf (-0.50) y Focus (-0.57) esta correlación no es tan clara.

```
# Correlación entre el kilometraje y el precio
# 1. A_Class
A_Class.cor <- cor(x=A_Class$mileage, y=A_Class$price, method='pearson')
# 2. Golf
Golf.cor <- cor(x=Golf$mileage, y=Golf$price, method='pearson')
# 3. Focus
Focus.cor <- cor(x=Focus$mileage, y=Focus$price, method='pearson')

A_Class.cor

## [1] -0.6530462
```

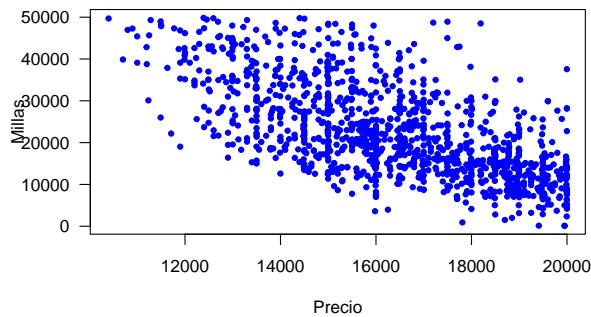
```
Golf.cor
```

```
## [1] -0.5003888
```

```
Focus.cor
```

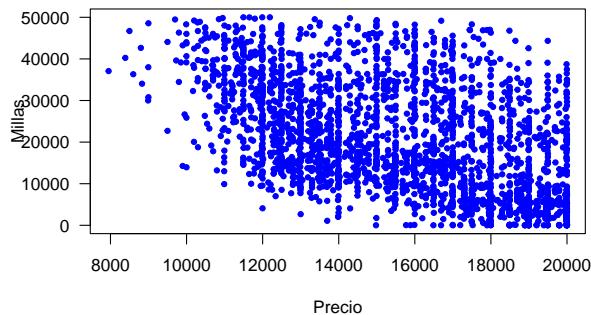
```
## [1] -0.5745208
```

```
# Gráficos de dispersión
# 1. A_Class
with(A_Class, plot(x=price, y=mileage, pch=20, col='blue',
                    xlab='Precio', las=1,
                    ylab='Millas'))
```



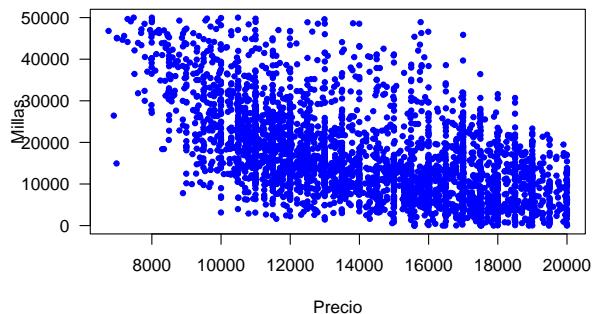
```
# 2. Golf
```

```
with(Golf, plot(x=price, y=mileage, pch=20, col='blue',
                  xlab='Precio', las=1,
                  ylab='Millas'))
```



```
# 3. Focus
```

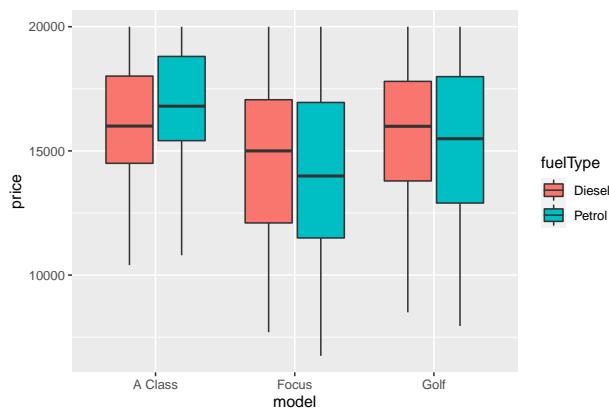
```
with(Focus, plot(x=price, y=mileage, pch=20, col='blue',
                  xlab='Precio', las=1,
                  ylab='Millas'))
```



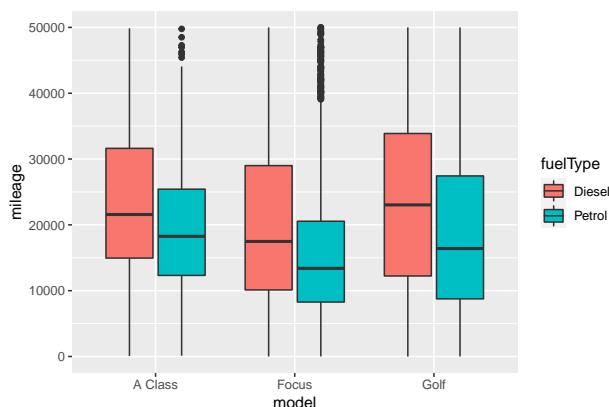
- Figura 8. Diagramas de dispersión entre el precio y el kilometraje

*3. El impacto del tipo de combustible (petrol vs. diesel) en el precio y kilometraje

```
# Gráficos
# 1. Precio vs tipo combustible
ggplot(final_cars, aes(x=model, y=price, fill = fuelType)) + geom_boxplot()
```



```
#2. kilometraje vs combustible
ggplot(final_cars, aes(x=model, y=mileage, fill = fuelType)) + geom_boxplot()
```



- Figura 8. Precio y kilometraje en función del tipo de combustible por modelo de coche

Realizamos un análisis de la varianza de dos factores (TWO-WAY ANOVA) y observamos diferencias significativas en función del modelo y del precio. Además hay una interacción entre las variables que nos indica que el precio o kilometraje en función del tipo de combustible depende del modelo.

Este análisis no aporta mucha información, sin embargo los gráficos nos permiten sacar las siguientes conclusiones:

1. Para el Focus y Golf el precio de los coches gasolina son más baratos mientras que el clase A es lo contrario.
2. El general los coches diesel tiene más kilometraje que los gasolina.

```
# Two-way ANOVA precio en función del modelo y tipo de combustible
res.aov <- aov(price ~ model * fuelType, data = final_cars)
summary(res.aov)
```

```
##          Df   Sum Sq Mean Sq F value    Pr(>F)
## model      2 5.047e+09 2.523e+09 313.06 < 2e-16 ***
## fuelType   1 1.539e+08 1.539e+08 19.09 1.27e-05 ***
## model:fuelType 2 3.664e+08 1.832e+08 22.73 1.45e-10 ***
## Residuals 7271 5.861e+10 8.060e+06
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
TukeyHSD(res.aov, which = "fuelType")
```

```
## Tukey multiple comparisons of means
## 95% family-wise confidence level
##
## Fit: aov(formula = price ~ model * fuelType, data = final_cars)
##
## $fuelType
##        diff      lwr      upr     p adj
## Petrol-Diesel -280.2895 -414.248 -146.331 4.15e-05
```

```
# Two-way ANOVA precio en función del modelo y tipo de transmisión
res.aov2 <- aov(price ~ model * transmission, data = final_cars)
summary(res.aov2)
```

```
##          Df   Sum Sq Mean Sq F value    Pr(>F)
## model      2 5.047e+09 2.523e+09 331.29 <2e-16 ***
## transmission 2 1.993e+09 9.964e+08 130.82 <2e-16 ***
## model:transmission 4 1.775e+09 4.438e+08 58.26 <2e-16 ***
## Residuals 7268 5.536e+10 7.617e+06
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
TukeyHSD(res.aov2, which = "transmission")
```

```

## Tukey multiple comparisons of means
## 95% family-wise confidence level
##
## Fit: aov(formula = price ~ model * transmission, data = final_cars)
##
## $transmission
##          diff      lwr      upr p adj
## Manual-Automatic -1386.0603 -1617.2532 -1154.8673 0e+00
## Semi-Auto-Automatic -638.0686 -927.4135 -348.7236 7e-07
## Semi-Auto-Manual    747.9917  533.0984  962.8850 0e+00

```

5 Análisis 2. Caso práctico compra-venta

Somos una empresa que se dedica a hacer negocio comprando coches baratos, arreglándolos y volviéndolos a vender más caros. Queremos saber qué coches del conjunto de datos se venden al menos un 20% debajo de su precio de mercado (“chollos”) para poder comprarlos y revenderlos.

Para ello vamos a modelar el “mercado de compraventa” mediante un modelo de regresión lineal. Para ello utilizaremos la función “lm” la cual internamente transformará todas las clases de nuestras variables cualitativas en variables dicotómicas para poder obtener el valor estimado como la combinación lineal de todas las variables.

A continuación vemos los coeficientes:

```

used_cars[] <- lapply(used_cars, function(x) if(is.factor(x)) factor(x) else x)
lm_model <- lm(price~., used_cars)
#lm_model$coefficients Comentado por exceso de páginas

```

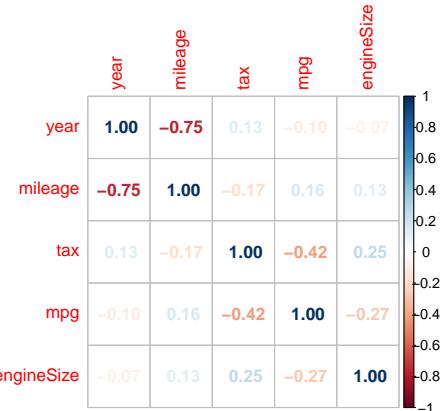
- Como podemos analizar por manufacturer los coches Mercedes “manufacturermerc” son más caros (“añaden” 1309€ a la estimación final) y los Toyota son más baratos (“restan” 9187 € a la estimación final).
- En cuanto a modelos por ejemplo Audi RS6 altera al alza el precio sumando 20120 € al precio.
- En cuanto a la transmision vemos que los Manuales son más baratos por lo general que los Semi-Auto.
- Y los coches híbridos son más caros que los de gasolina y diesel.
- Al ser más nuevo, el precio es mayor.
- Mientras más mayor es son las tasas el coche suele ser más barato (coches antiguos, contaminan más, mayor tasa)
- Al tener mayor “kilometraje” el precio es menor.
- El tamaño del motor determina positivamente el precio del coche.

Veamos también la correlación de las variables numéricas por si tuvieramos que eliminar alguna del modelo por ser redundante:

```

matriz_correl <- cor(used_cars %>% select(c("year", "mileage", "tax", "mpg", "engineSize")))
corrplot::corrplot(matriz_correl, method="number")

```



Vemos que “mileage” y “year” están correlacionadas inversamente, mientras más nuevo el coche, menor es el kilometraje. Igual sucede con el consumo y las tasas, mientras mayor distancia recorre con menos combustible, menor es la tasa (menor consumo, menor tasa). A pesar de todo no hablamos de correlaciones muy altas. Por lo que podemos aceptar el modelo.

Veamos la calidad de nuestro modelo mediante el RMSE (raíz de suma de mínimos cuadrados) para tenerlo el error en unidades de precio:

```
sqrt(mean(lm_model$residuals^2))
```

```
## [1] 3704.997
```

Vemos que no es un error despreciable y tendriamos que refinar un poco más nuestro modelo. Con el presente modelo haremos responderemos a la pregunta anterior ¿Cuáles son los chollos del conjunto, coches el 20% por debajo del precio de mercado?

```
mean((lm_model$fitted.values-(used_cars$price*1.2)) > 0)
```

```
## [1] 0.1412599
```

Tenemos que el ~14% está al menos un 20% por debajo del mercado. Por ejemplo (un sampleo de 10):

```
used_cars[(lm_model$fitted.values-(used_cars$price*1.2)) > 0,] %>% sample_n(., 10) %>% knitr::kable()
```

manufacturer	model	year	price	transmission	mileage	fuelType	tax	mpg	engineSize
ford	Fiesta	2018	9495	Manual	15835	Petrol	150	64.2	1.1
merc	A Class	2016	16399	Manual	16088	Diesel	30	62.8	2.1
vw	Golf	2017	12999	Semi-Auto	28295	Diesel	20	58.9	1.6
vw	Golf	2019	14995	Semi-Auto	15110	Diesel	145	54.3	1.6
vw	Golf	2017	11500	Semi-Auto	57600	Petrol	30	55.4	1.4
audi	A4	2015	13495	Manual	1800	Diesel	30	65.7	2.0
ford	Fiesta	2017	8999	Manual	20589	Petrol	125	54.3	1.2
bmw	2 Series	2016	9500	Manual	59509	Diesel	20	68.9	2.0
ford	Focus	2016	9250	Manual	20288	Petrol	20	61.4	1.0
vauxhall	Corsa	2019	9990	Manual	100	Petrol	145	43.5	1.4

6 Conclusiones

Con cada uno de los análisis descritos en este proyecto queremos ayudar al comprador medio y a un vendedor en la toma de decisión a la hora de comprar un coche, tanto para su uso como para su posterior reventa.

En el caso del comprador le hemos propuesto tres modelos que cumplen con los requisitos del cliente. Una vez leido el informe le corresponderá al comprador hacer la elección. Sin embargo, todos los análisis muestran al Focus con una mejor relación-precio en comparación con el Clase A y el Golf.

El análisis del caso “Compra-venta” arroja que el 14% de los coches ofertados están por debajo del 20% del precio de mercado. Pero podemos reconocer que el modelo es mejorable, siendo más restringentes a la hora de seleccionar datos (eliminar outliers, valores altos) y probando con otros modelos como árboles de decisión.

7 Exportación del código en R y de los datos producidos.

El código ha sido producido con Rstudio y se produce tanto el PDF como el fichero html ejecutando fichero Rmd con el procesador Knit.

8 Contribución

En este proyecto han participado Marta Gómez Galán Y Juan Francisco Nieto Mendoza a partes iguales tanto en la investigación previa como en la redacción y creación de código.