EDA

Ignacio Vellido

11/13/2020

Intro

Para este trabajo contamos con dos datasets distintos: **autoMPG6** para aplicar Regresión y **haberman** para aplicar Clasificación.

Descripciones de los problemas

autoMPG6

http://lib.stat.cmu.edu/datasets/cars.desc

Este dataset codifica el consumo de gasolina de distintos coches (en millas por galón, Mpg) en base a las siguientes características:

1. Displacement: Indica la cilindrada del coche, la suma del volumen útil de los cilindros del motor, medido en pulgadas cúbicas.
2. Horse_power: Mide la potencia del coche.
3. Weight: Peso en libras.
4. Acceleration: Aceleración del coche de 0 a 60 millas por hora, medido en segundos.
5. Model_year: Indica las dos últimas cifras del año de producción.

El objetivo es poder predecir, en base a los cinco atributos, el consumo Mpg de un nuevo coche:

6. Mpg: Millas-por-galón, indica la cantidad de galones (1G +-= 3,78L) de fuél que consume un vehículo al recorrer una milla (1m +-= 1,6km).

El dataset contiene 392 instancias codificanto esta información.

Análisis Estadístico de Datos

autoMPG6

La descripción del problema nos da alguna información adicional sobre las variables:

1. Displacement: Variable numérica continua, contamos con valores reales en el rango [68.0,455.0].
2. Horse_power: Variable numérica continua, contamos con valores enteros en el rango [46,230].
3. Weight: Variable numérica continua, contamos con valores enteros en el rango [1613,5140].
4. Acceleration: Variable numérica continua, contamos con valores reales en el rango [8.0,24.8].
5. Model_year: Variable numérica discreta, contamos con valores enteros en el rango [70,82].
6. Mpg: Variable numérica continua, contamos con valores reales en el rango [9.0,46.6].

Hipótesis de partida

- H.1: Horse_power puede influir en Mpg: A más potencia, más consumo.

- H.2: Weight debe influir en Mpg: Un coche más pesado debería consumir más.
- H.3: Debería haber correlación entre displacement (cilindrada) con horse y acceleration
- H.4: Horse y acceleration podrían estar relacionadas
- H.5: Viendo que contamos con un rango pequeño de años, no debería haber un cambio significativo de prestaciones entre años
- H.6: Pero debería existir una tendencia de mejora de prestaciones con los años, incluyendo aumento de Displacement, Horse_power y Acceleration.
- H.7: Model_year podría no mostrar relación con Mpg: Pese al paso de los años si contamos con diferentes tipos de vehículos (todoterrenos, familiares, deportivos...) podría haber un consumo dispar. (Si existiera tendencia, viendo que los años son de las últimas décadas del siglo XX, podría ir el consumo hacia abajo)
- H.8: Esta última hipótesis se puede aplicar al resto de variables, indicándonos que Model_year no debería tener relevancia para este problema de regresión.
- H.9: Horse_power podría depender de las variables Displacement y Weight

---

Cargamos los datos:

```
names <- c("Displacement", "Horse_power", "Weight", "Acceleration", "Model_year", "Mpg")

auto <- read_csv("Data/autoMPG6/autoMPG6.dat", comment = "@", col_names = names)
```

```
-- Column specification --------------------------------------------------------
cols(
  Displacement = col_double(),
  Horse_power = col_double(),
  Weight = col_double(),
  Acceleration = col_double(),
  Model_year = col_double(),
  Mpg = col_double()
)
```

Antes de comenzar a analizar las variables nos plantemos una cuestión: ¿Debemos considerar Model_year como una variable numérica o como un factor categórico? Aunque por la hipótesis H.7 podríamos acabar no eligiendo la variable para el problema, es necesario plantearse esta cuestión antes de comenzar.

Sabemos que las observaciones para esta variable cuenta con valores entre 72 y 82, por lo que tenemos información exacta del año (en comparación, por ejemplo, con agrupaciones mayores como la década o el siglo). El hecho de tratarla como categórica o quantitativa depende mucho del problema. En este caso, tenemos interés en cuestionarnos por valores entre años, por ejemplo, el consumo entre los años 75 y 76 (por otro lado, no tenemos información más precisa para los meses dentro del año)

En un principio, el dataset está planteado para regresión, por lo que tendríamos dos opciones: - Mantenerlo como categórico y generar variables dummy (Valores 0-1 para indicar si la instancia es de ese año). Suponiendo que tenemos al menos una instancia de cada año, esto nos generaría 12 variables nuevas. - Mantenerlo como numérico, pero teniendo cuidado de cómo interpretar el año.

Proseguimos con tanto dejando Model_year como variable numérica.

```
head(auto)
```

Análisis univariable

Displacement	Horse_power	Weight	Acceleration	Model_year	Mpg
91	70	1955	20.5	71	26.0
232	100	2789	15.0	73	18.0
350	145	4055	12.0	76	13.0
318	140	4080	13.7	78	17.5
113	95	2372	15.0	70	24.0
97	60	1834	19.0	71	27.0

Hacemos summary para sacar datos de relevancia

```
summary(auto)
```

```
  Displacement    Horse_power        Weight      Acceleration     Model_year
 Min.   : 68.0   Min.   : 46.0   Min.   :1613   Min.   : 8.00   Min.   :70.00
 1st Qu.:105.0   1st Qu.: 75.0   1st Qu.:2225   1st Qu.:13.78   1st Qu.:73.00
 Median :151.0   Median : 93.5   Median :2804   Median :15.50   Median :76.00
 Mean   :194.4   Mean   :104.5   Mean   :2978   Mean   :15.54   Mean   :75.98
 3rd Qu.:275.8   3rd Qu.:126.0   3rd Qu.:3615   3rd Qu.:17.02   3rd Qu.:79.00
 Max.   :455.0   Max.   :230.0   Max.   :5140   Max.   :24.80   Max.   :82.00
      Mpg
 Min.   : 9.00
 1st Qu.:17.00
 Median :22.75
 Mean   :23.45
 3rd Qu.:29.00
 Max.   :46.60
```

El dataset no cuenta con valores repetidos

```
sum(duplicated(auto))
```

```
[1] 0
```

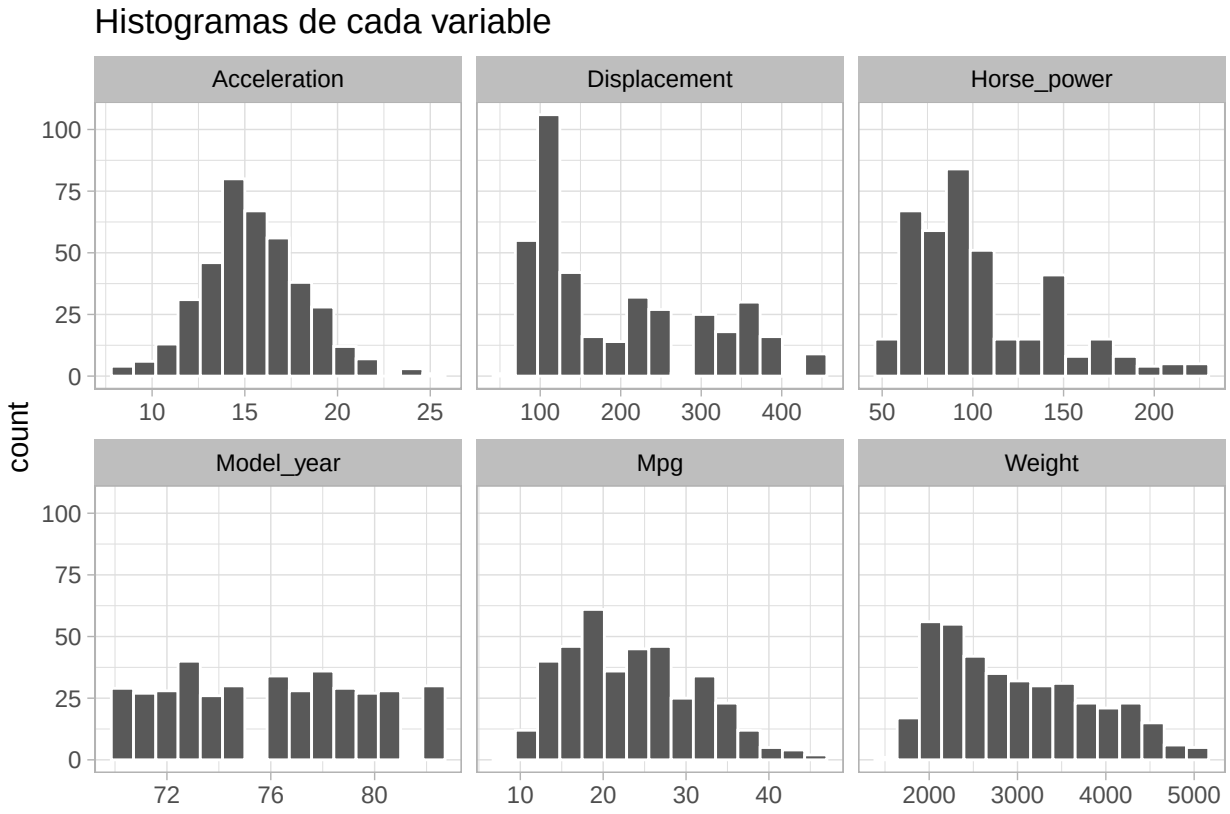
Ni missing values

```
sum(is.na(auto))
```

```
[1] 0
```

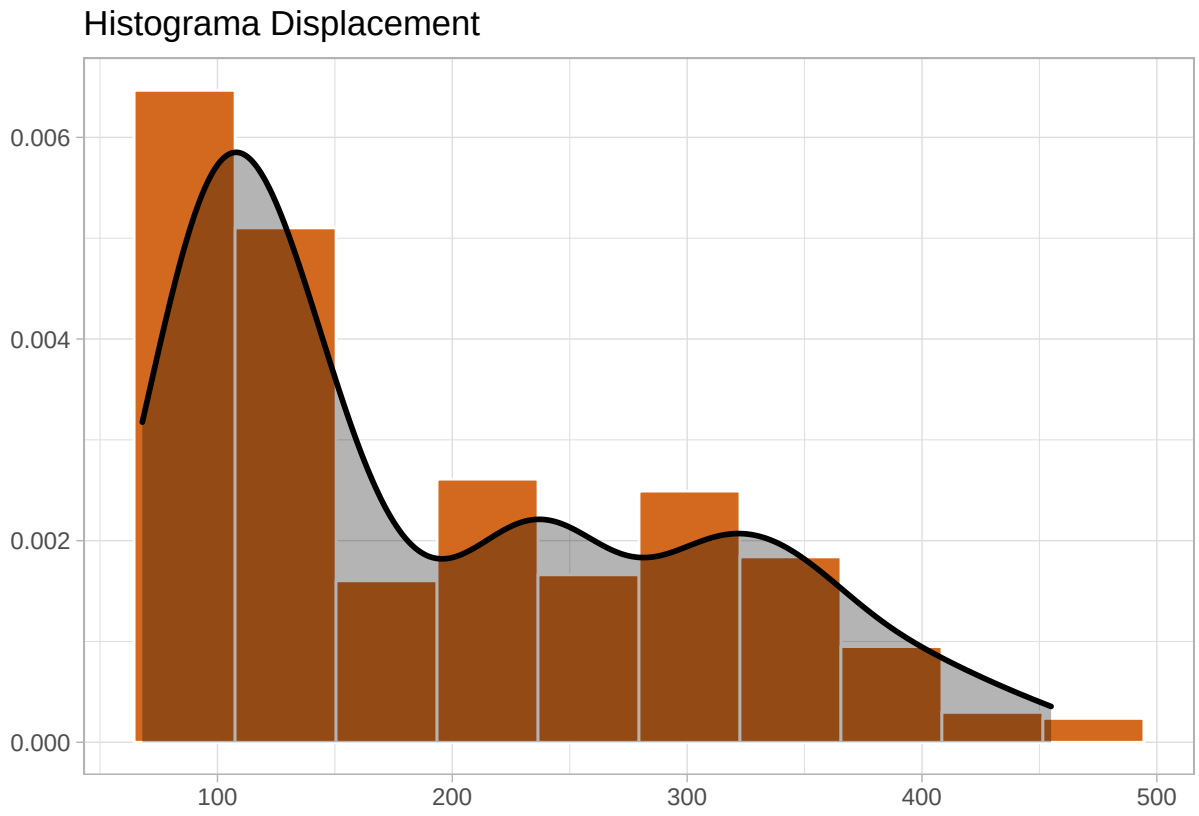
Vamos a sacar plots de cada variable para verlo mejor

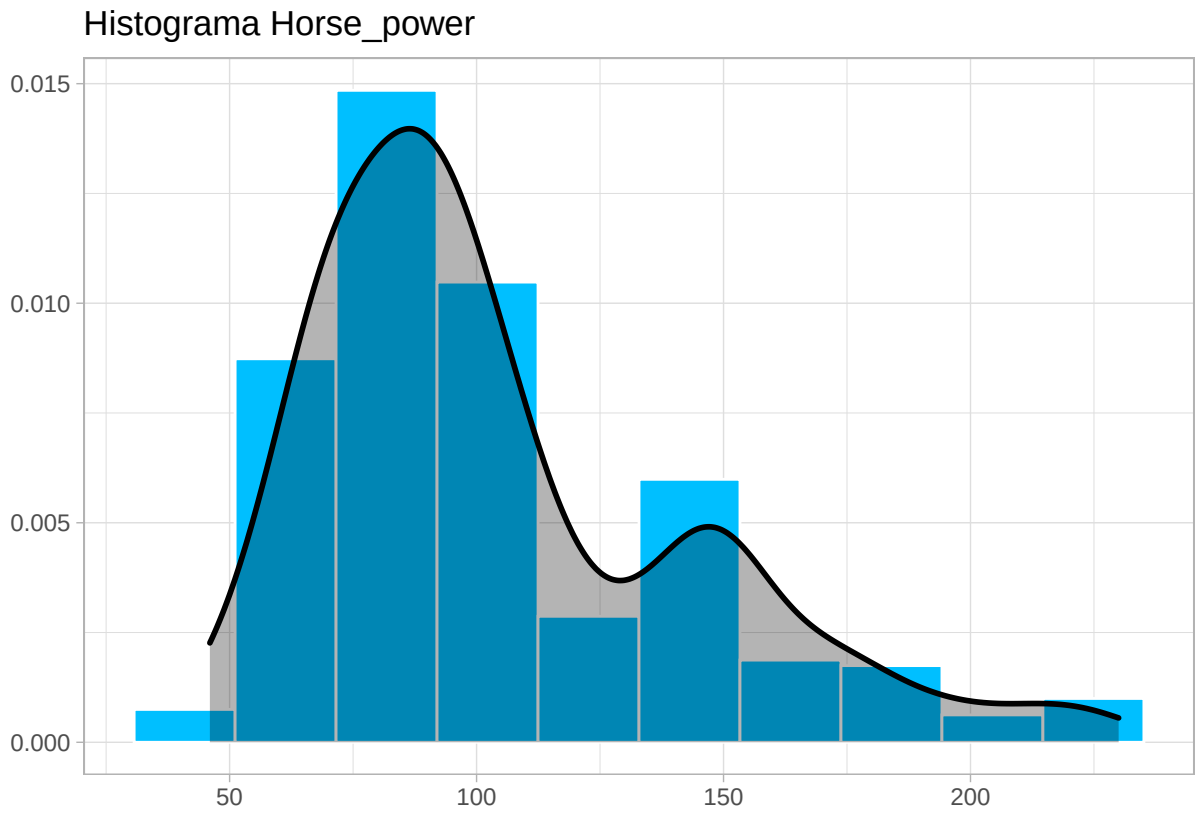
```
ggplot(gather(auto), aes(value)) +
  geom_histogram(bins = 15, color="white") +
  facet_wrap(~key, scales = 'free_x') +
  theme_light() +
  theme(strip.background = element_rect(fill="grey", size=2))+
  theme(strip.text = element_text(colour = 'black')) +
  labs(title="Histogramas de cada variable", x = "")
```

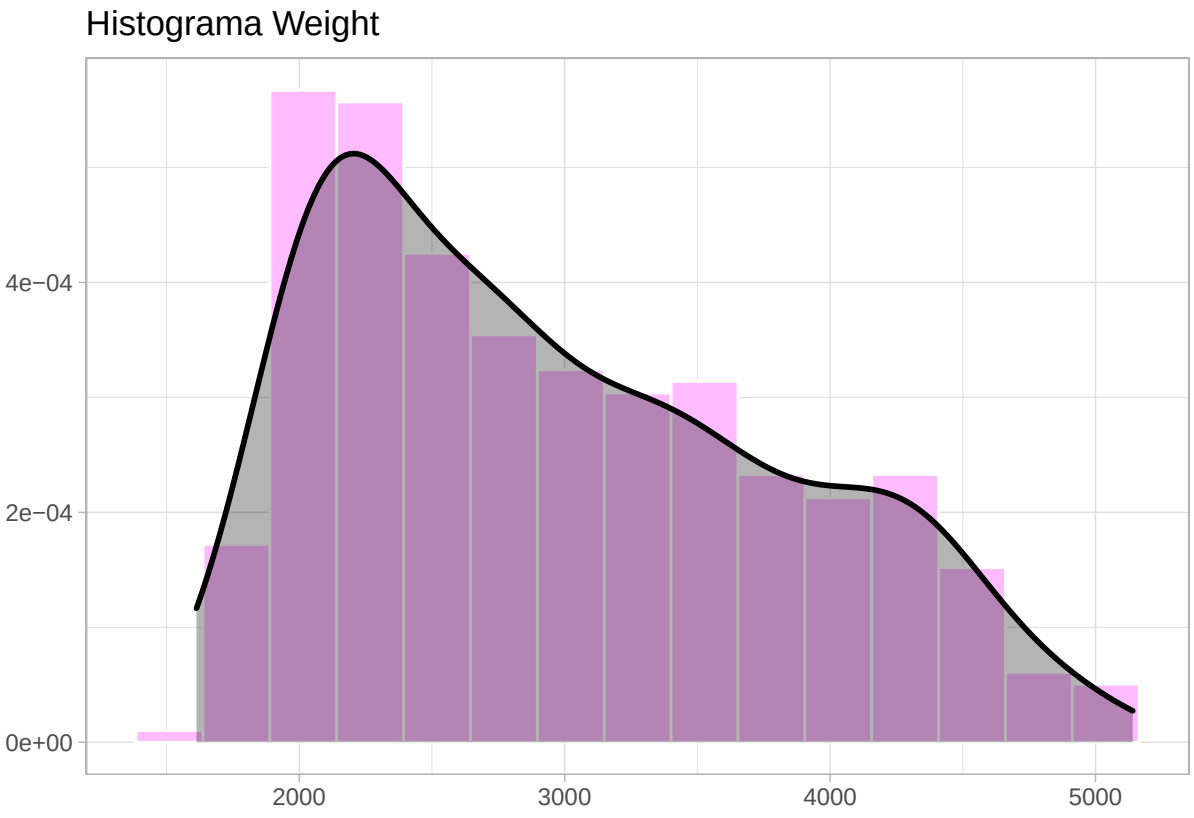


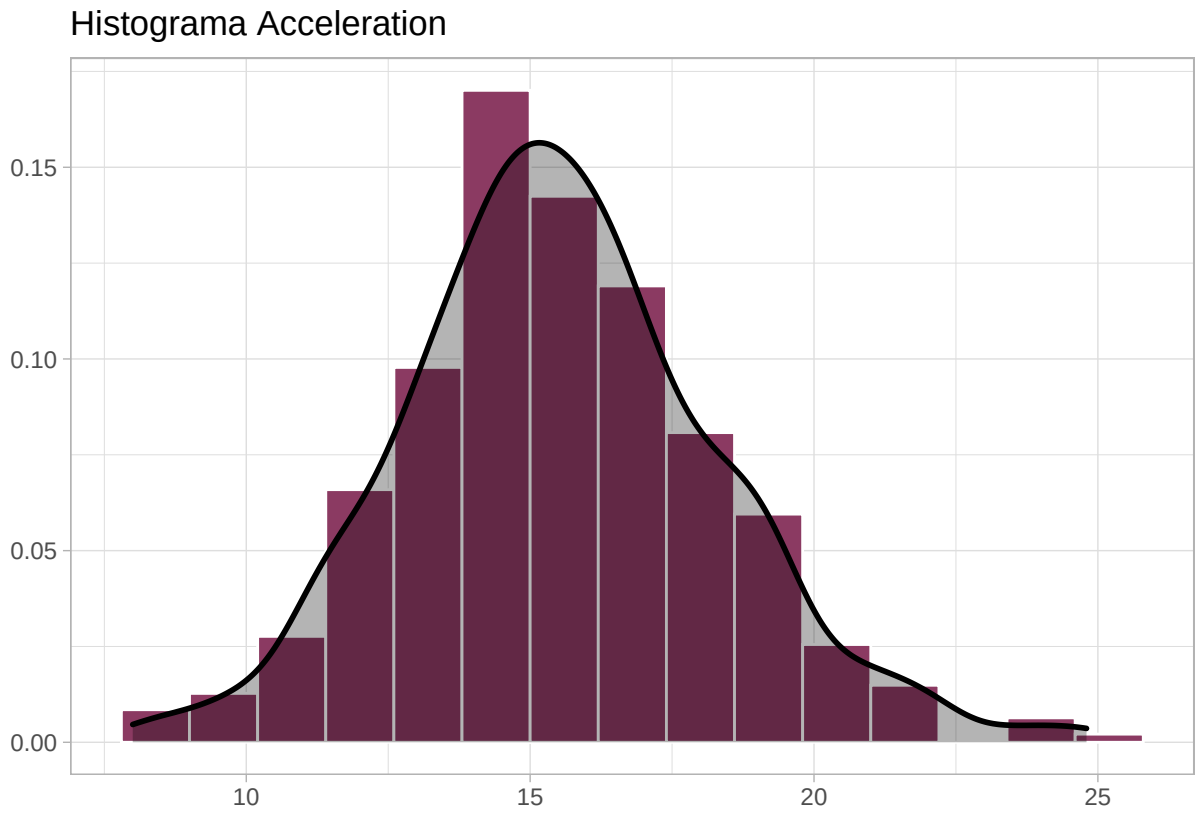
Una a una

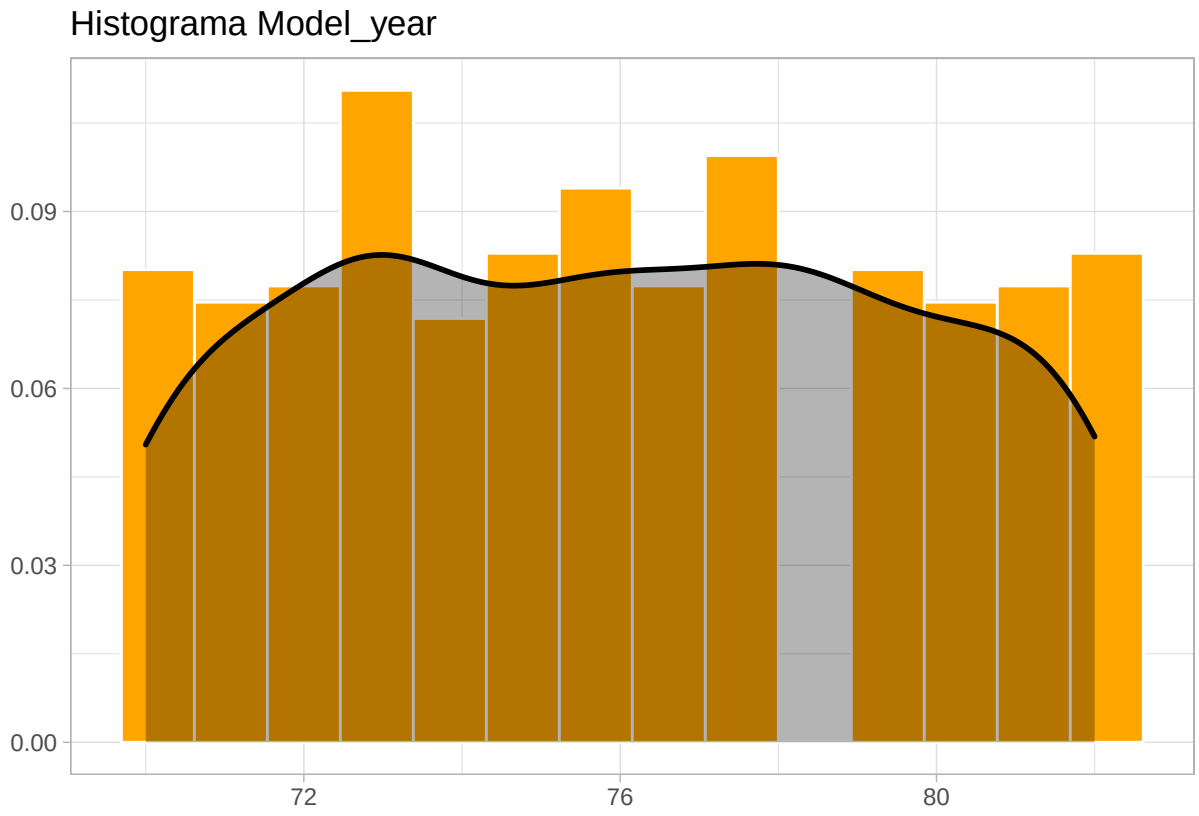
```
colors <- c("chocolate", "deepskyblue1", "plum1", "hotpink4", "orange", "springgreen4")
bins <- c(10,10,15,15,14,18)
plt <- list(length = length(names))

for (i in 1:length(names)) {
  ggplot(auto, aes_string(x=names[i])) +
    geom_histogram(aes(y=..density..), bins=bins[i], color="white", fill=colors[i]) +
    geom_density(alpha=.3, fill="black", size=1) +
    labs(title="", x="", y="") +
    theme_light() -> plt[[i]]

  print(plt[[i]] + labs(title=sprintf("Histograma %s", names[i]), x=""))
}
```

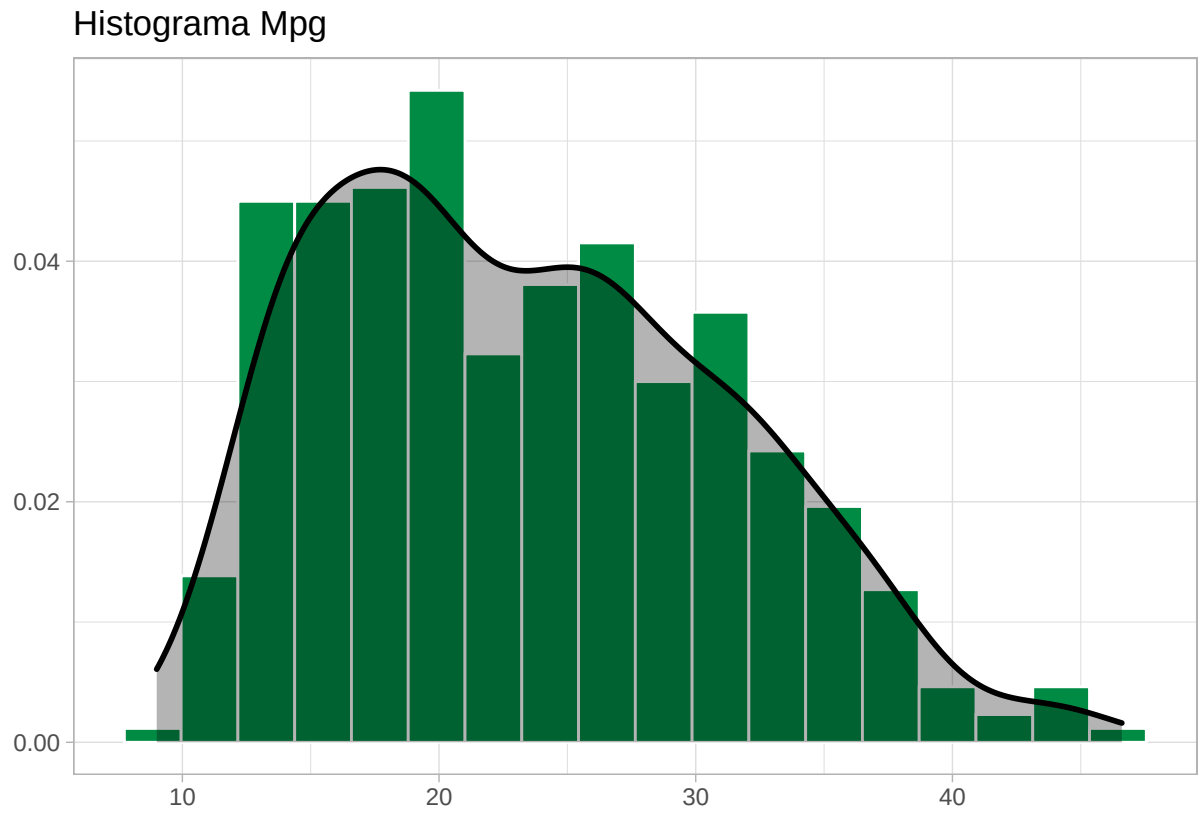

Histograma Displacement
0.006
0.004
0.002
0.000
100
200
300
400
500


Histograma Horse_power
0.015
0.010
0.005
0.000
50
100
150
200

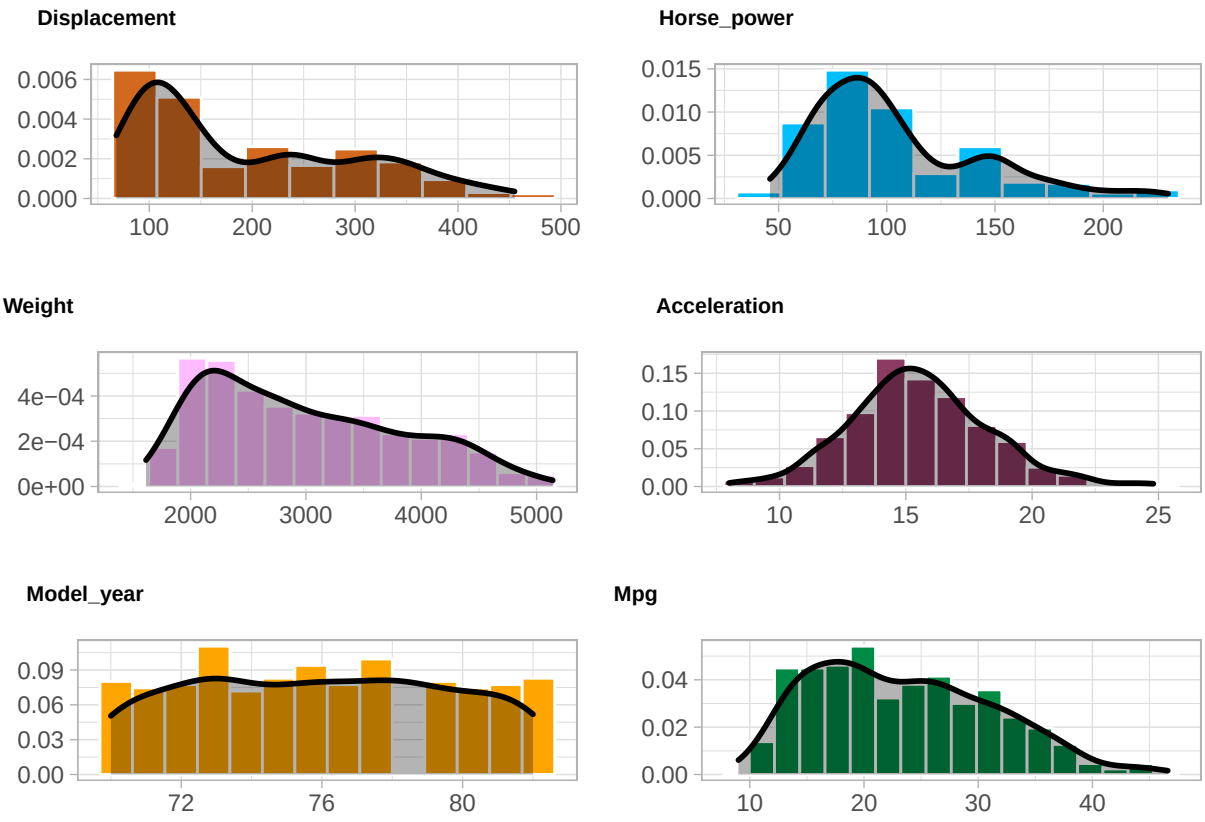

Histograma Weight
4e−04
2e−04
0e+00
2000
3000
4000
5000


Histograma Acceleration
0.15
0.10
0.05
0.00
10
15
20
25

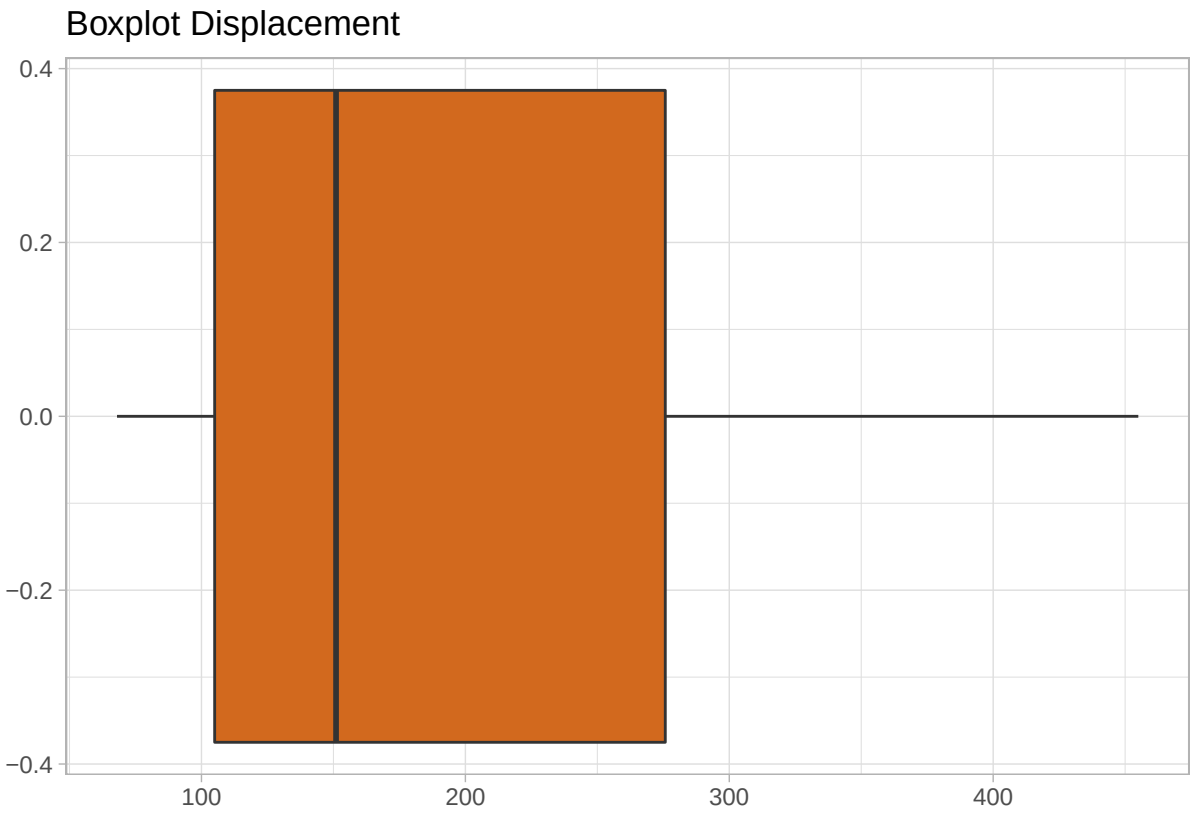

Histograma Model_year
0.09
0.06
0.03
0.00
72
76
80

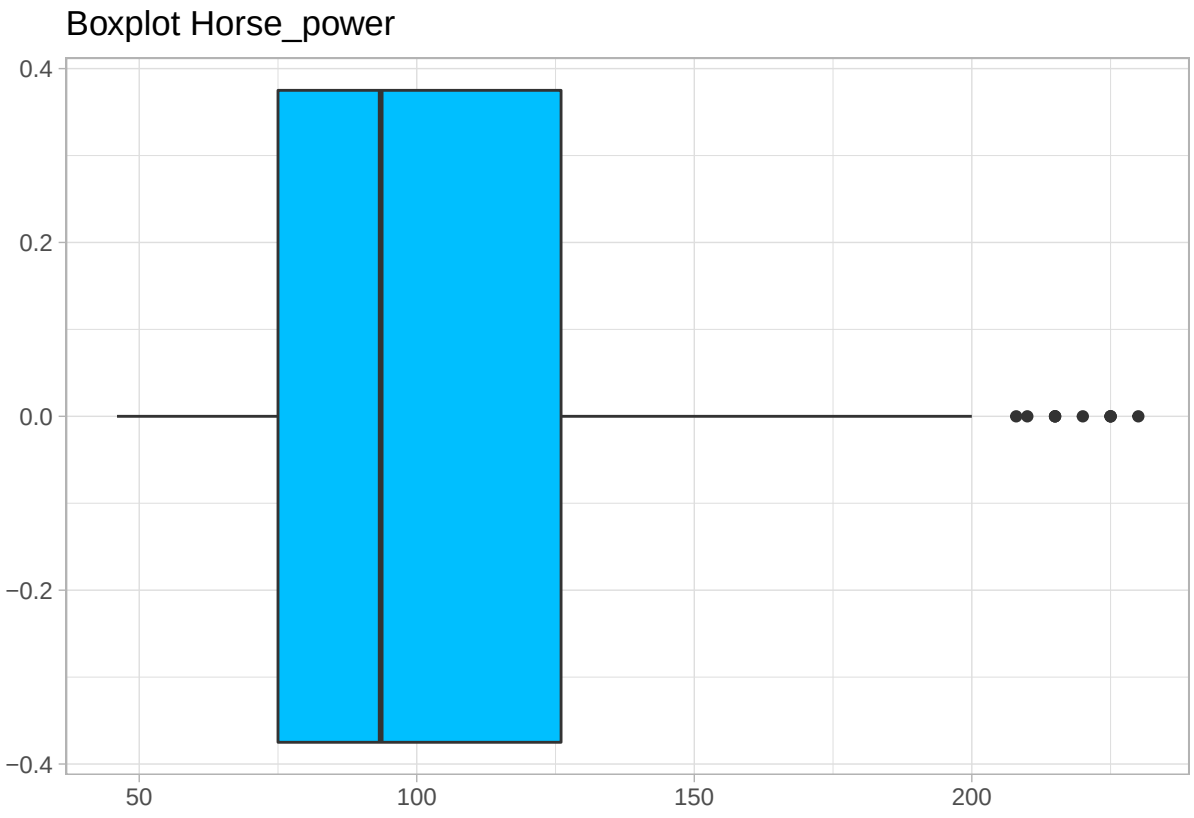


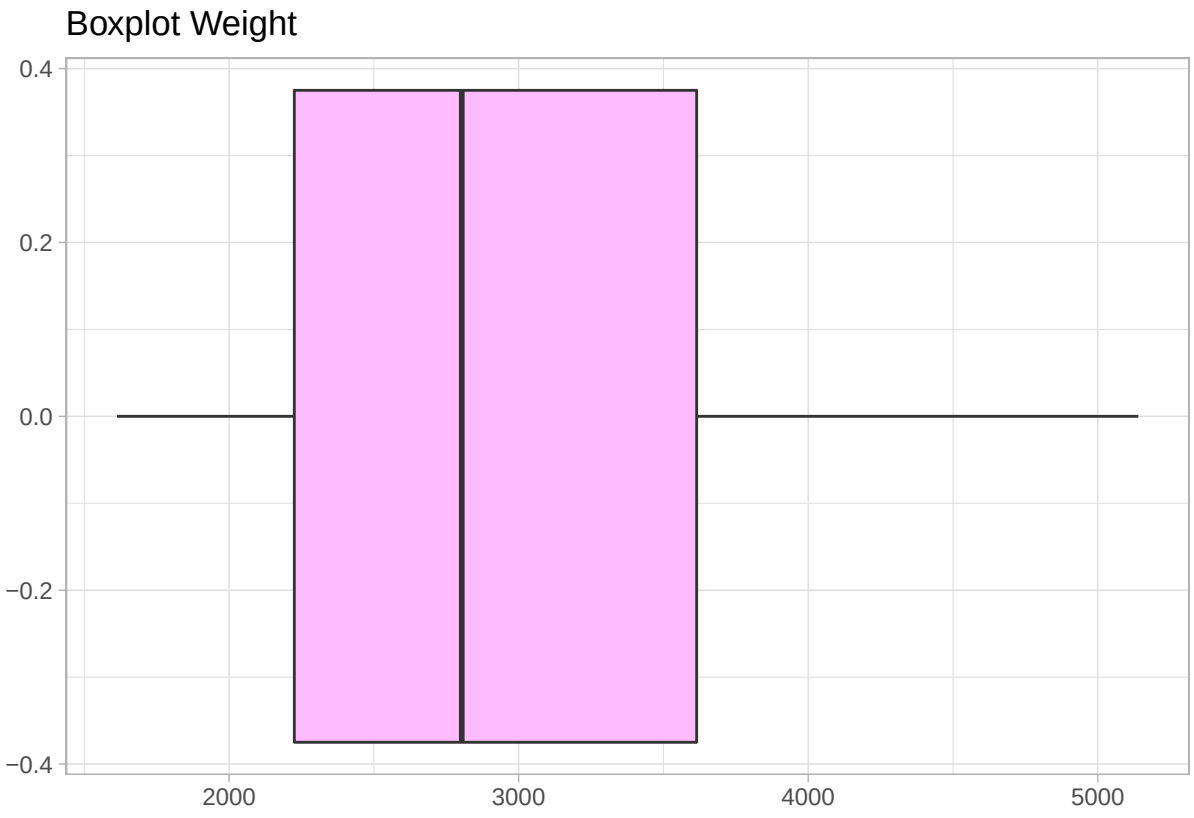

```
plot_grid(plotlist=plt, ncol=2, labels = names, label_size = 8)
```

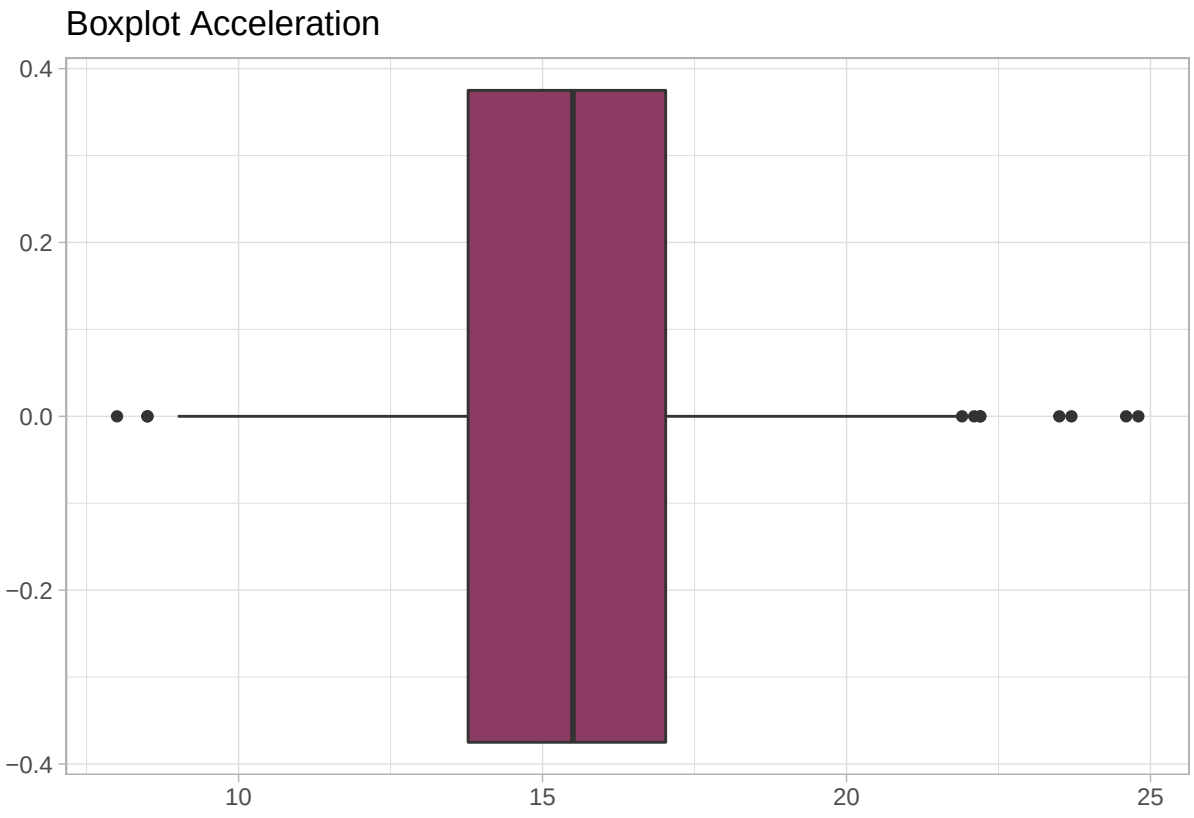


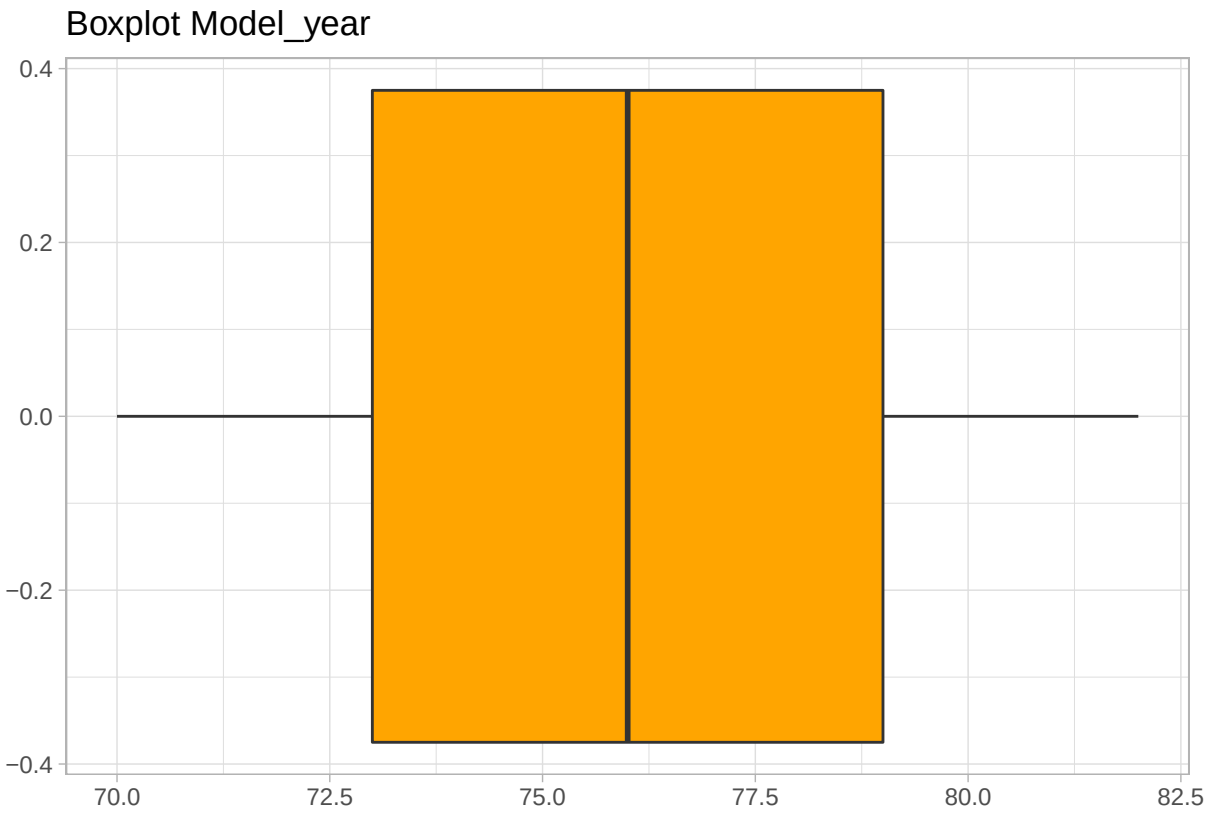

```r
colors <- c("chocolate", "deepskyblue1", "plum1", "hotpink4", "orange", "springgreen4")
plt <- list(length = length(names))

for (i in 1:length(names)) {
  ggplot(auto, aes_string(x=names[i])) +
    geom_boxplot(fill = colors[i]) +
    labs(title="", x="", y="") +
    theme_light() -> plt[[i]]

  print(plt[[i]] + labs(title=sprintf("Boxplot %s", names[i]), x=""))
}
```

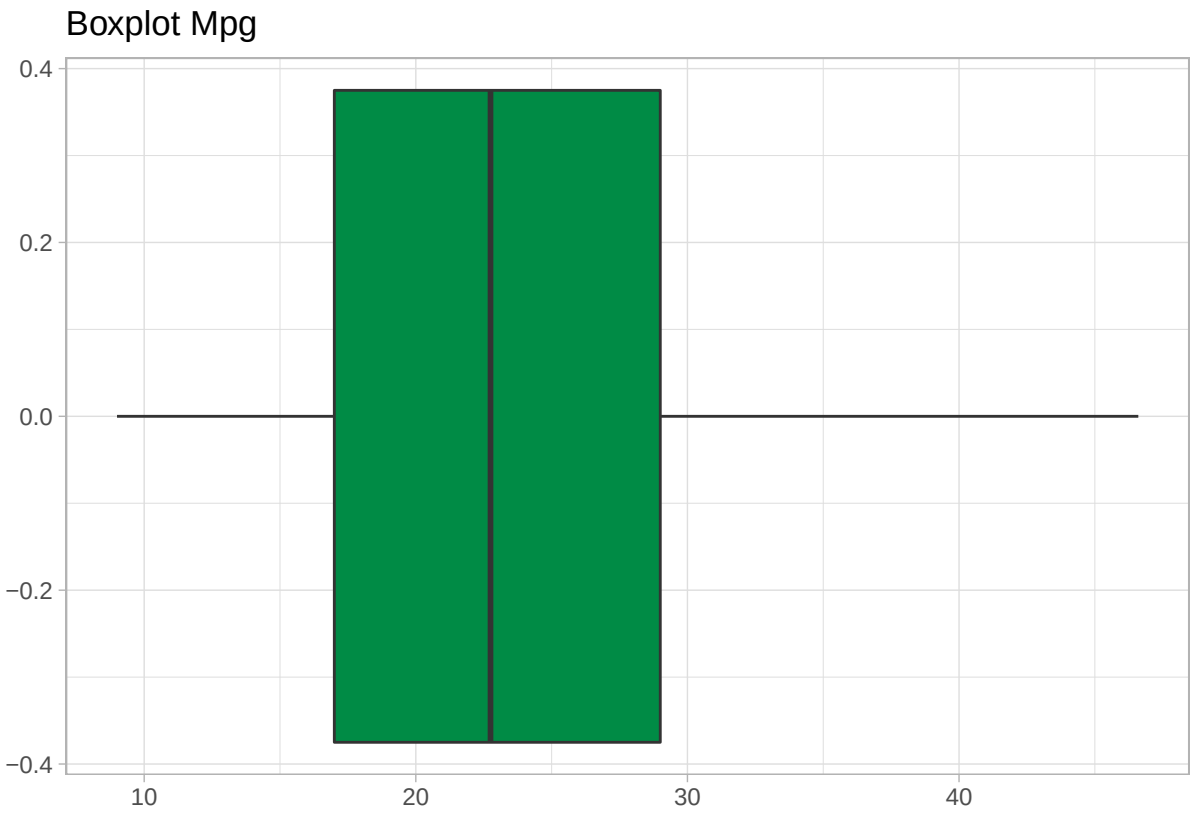

Boxplot Displacement
0.4
0.2
0.0
−0.2
−0.4
100
200
300
400


Boxplot Horse_power
0.4
0.2
0.0
−0.2
−0.4
50
100
150
200

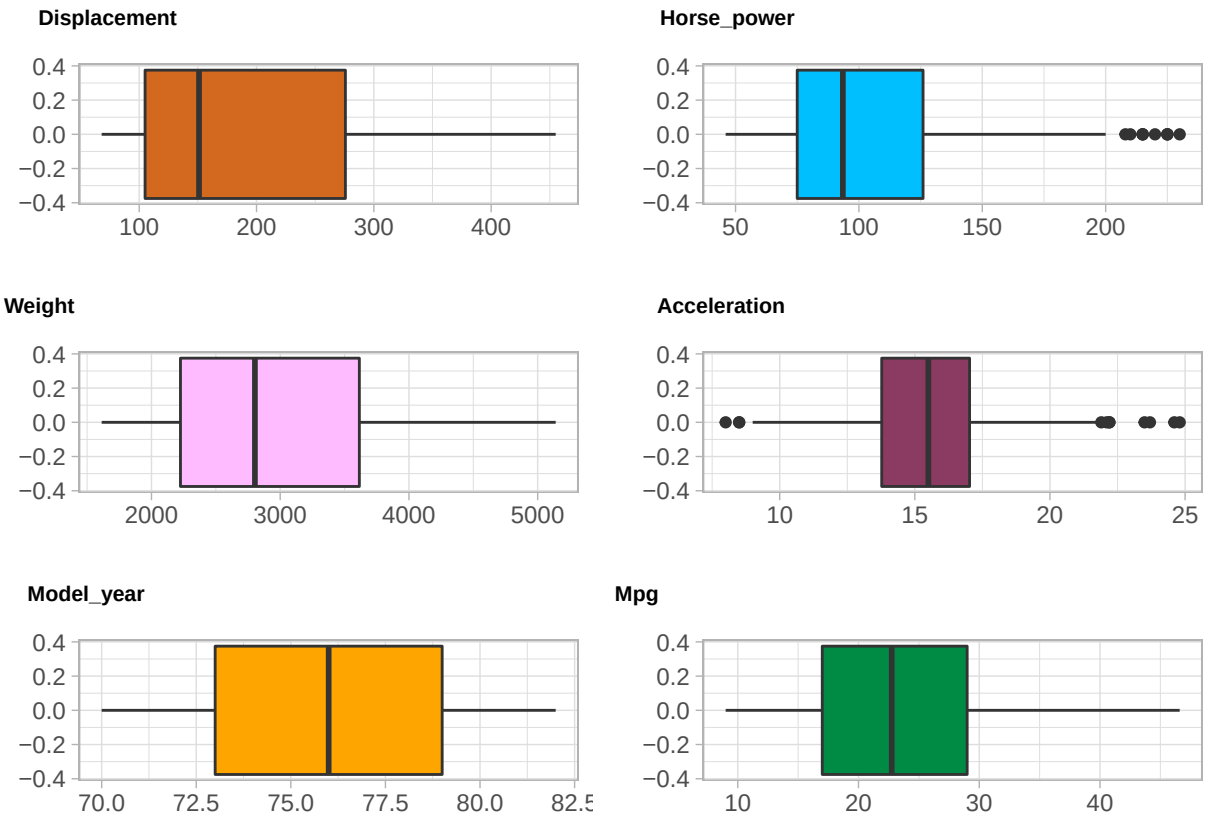

Boxplot Weight
0.4
0.2
0.0
−0.2
−0.4
2000
3000
4000
5000


Boxplot Acceleration
0.4
0.2
0.0
−0.2
−0.4
10
15
20
25


Boxplot Model_year
0.4
0.2
0.0
−0.2
−0.4
70.0
72.5
75.0
77.5
80.0
82.5

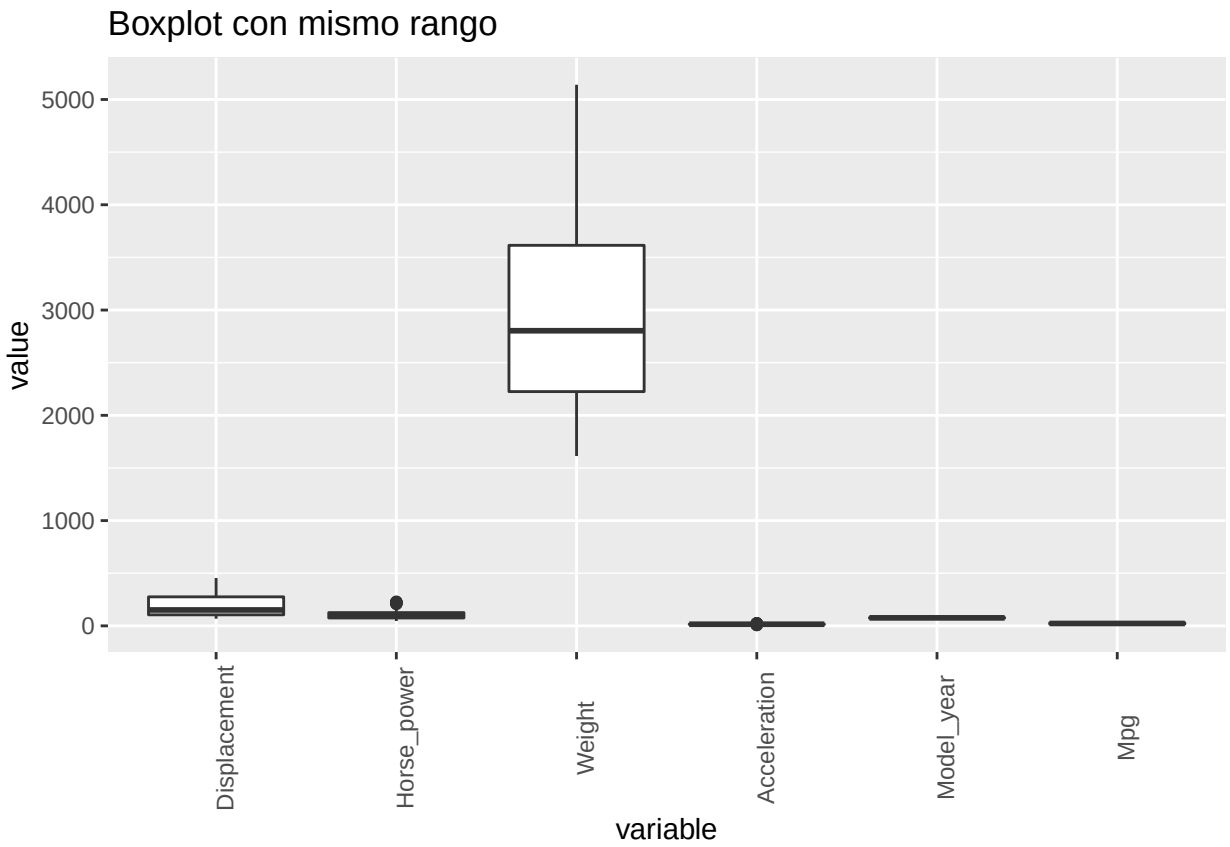



```
plot_grid(plotlist=plt, ncol=2, labels = names, label_size = 8)
```



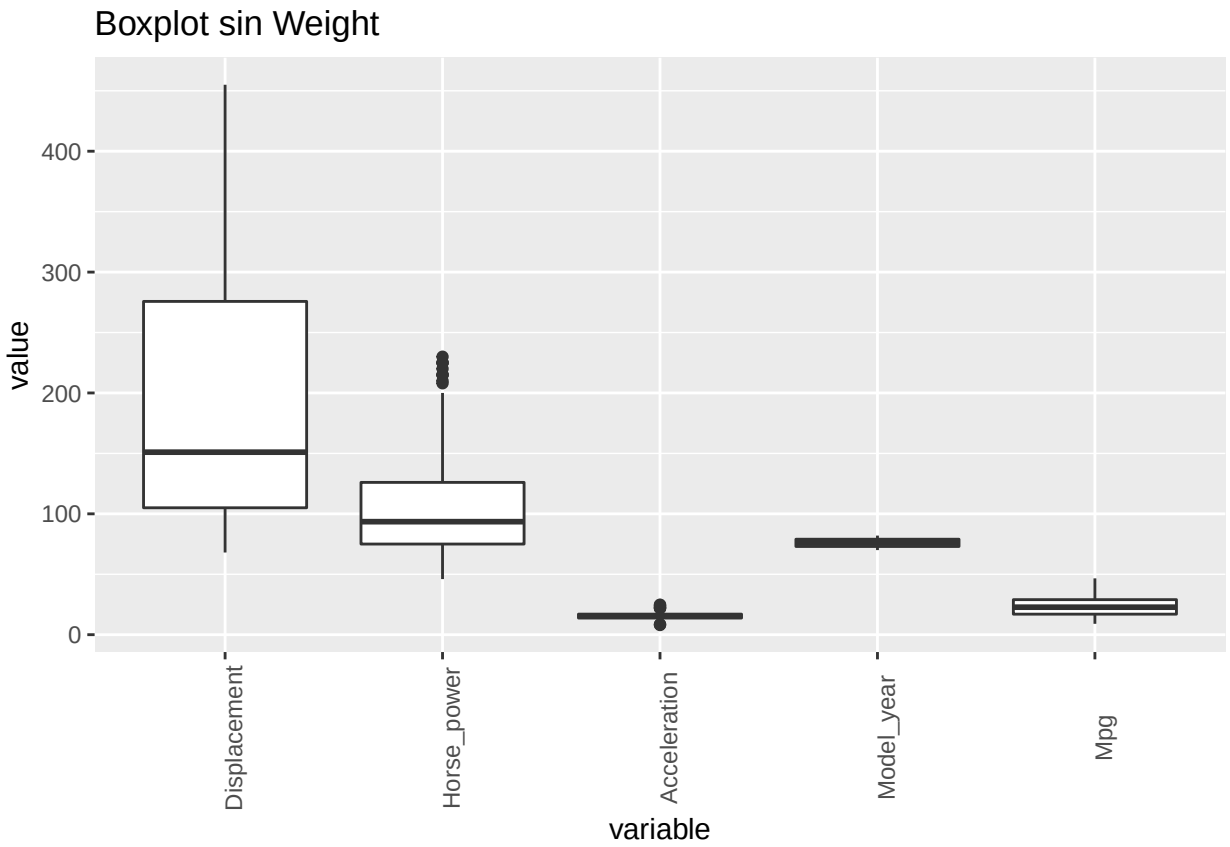

```
ggplot(melt(auto), aes(x=variable, y=value)) +
  geom_boxplot() +
  labs(title="Boxplot con mismo rango") +
  theme(axis.text.x = element_text(angle = 90))
```

```
No id variables; using all as measure variables
```




```
auto %>% dplyr::select(-Weight) %>%
  melt() %>%
  ggplot(aes(x=variable, y=value)) +
    geom_boxplot() +
    labs(title="Boxplot sin Weight") +
    theme(axis.text.x = element_text(angle = 90))
```

```
No id variables; using all as measure variables
```



Ya la descripción del problema nos lo decía, los rangos en los que se distribuyen los datos son muy diferentes dependiendo de la variable. Se pueden estandarizar los datos para solucionar este problema, aunque para regresión lineal no es necesario (sí lo es para KNN)

Podemos comparar los rangos intercuartiles si estandarizamos antes el dataset

```
scale(auto) %>% apply(2, IQR)
```

```
Displacement  Horse_power       Weight Acceleration   Model_year          Mpg
    1.631723     1.324980     1.635856     1.178021     1.628781     1.537475
```

También podemos ver la distancia entre mínimos y máximos

```
scale(auto) %>% apply(2, range) %>% apply(2, dist)
```

```
Displacement  Horse_power       Weight Acceleration   Model_year          Mpg
    3.698253     4.780318     4.152330     6.089463     3.257562     4.817420
```

Displacement: (coger los plots de la variable sola)

La cilindrada vemos con una desviación grande y una gran concentración en los valores inferiores. Desviado a la izquierda, no parece seguir una distribución normal. Existe una alta concentración en torno al valor 125, muy por encima del recuento que alcanzan el resto de valores

Horse_power Similar a Displacement pero cuenta con una mayor dispersión y algunos valores muya altos. A día de hoy los coches suelen rondar los 120 en turismos y los 200 en SUVs. Aquí contamos con predominancia en el rango aproximado [70, 125] con algunas instancias por encima de los 200. Desviado a la izquierda, no parece seguir una distribución normal.

Weight Una distribución más achatada que las anteriores, también ladeada hacia la izquierda. Un rango mayor

Acceleration Valores altamentes concentrados pero en general con un rango alto. Parece seguir una distribución normal.

Model_year Aunque no se vea bien en las gráficas, contamos con valores de todos los años, más o menos equitativamente

```
table(auto$Model_year)
```

```
70 71 72 73 74 75 76 77 78 79 80 81 82
29 27 28 40 26 30 34 28 36 29 27 28 30
```

Análisis sobre las distribuciones

Hemos comentado antes que no apreciamos semejanzas con una distribución normal en algunas de las variables, lo comprobamos con un test estadístico (Shapiro-Wilk test):

```
normality(auto) %>% filter(p_value < 0.05)
```

```
Warning: `cols` is now required when using unnest().
Please use `cols = c(statistic)`
```

vars	statistic	p_value	sample
Displacement	0.8818359	0.0000000	392
Horse_power	0.9040975	0.0000000	392
Weight	0.9414661	0.0000000	392
Acceleration	0.9918671	0.0305289	392
Model_year	0.9469666	0.0000000	392
Mpg	0.9671696	0.0000001	392

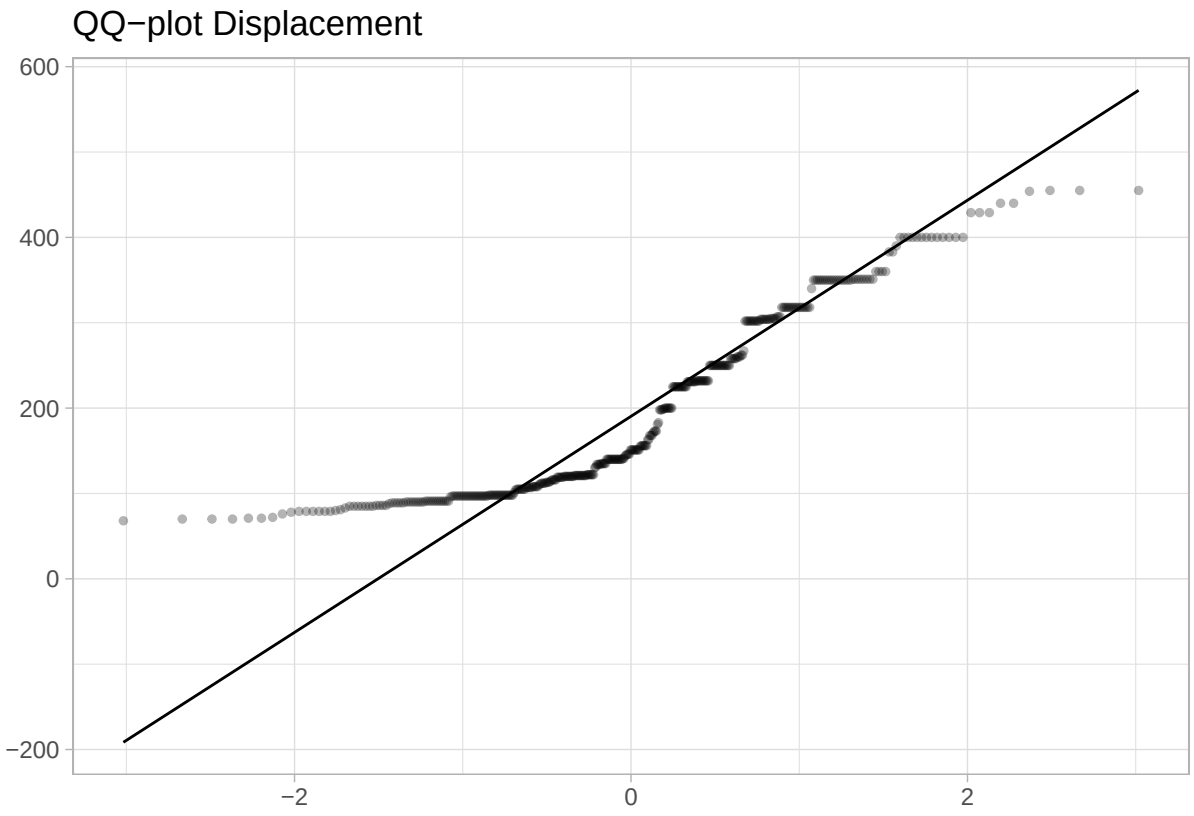
El test de Shapiro nos dice que ninguna variable sigue una distribución normal, con bastante certeza excepto en Acceleration.

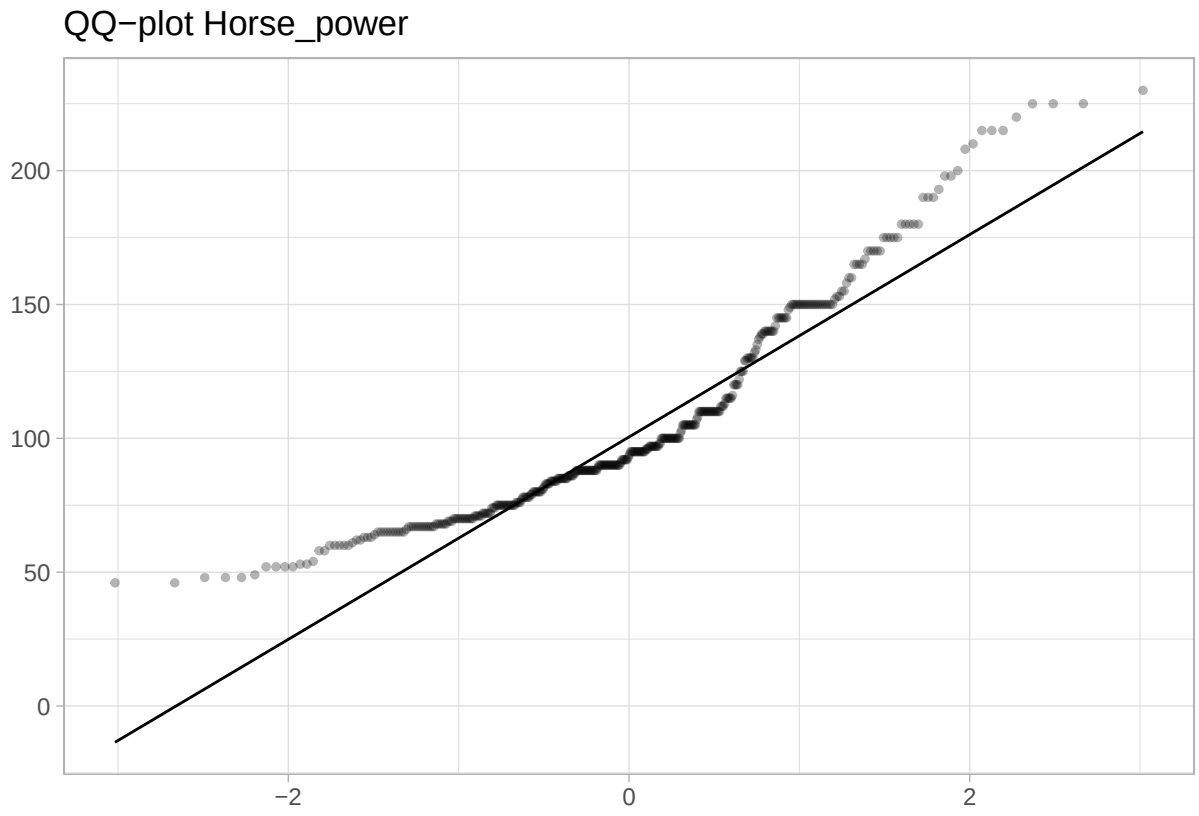
Para regresión aún así no es necesario.

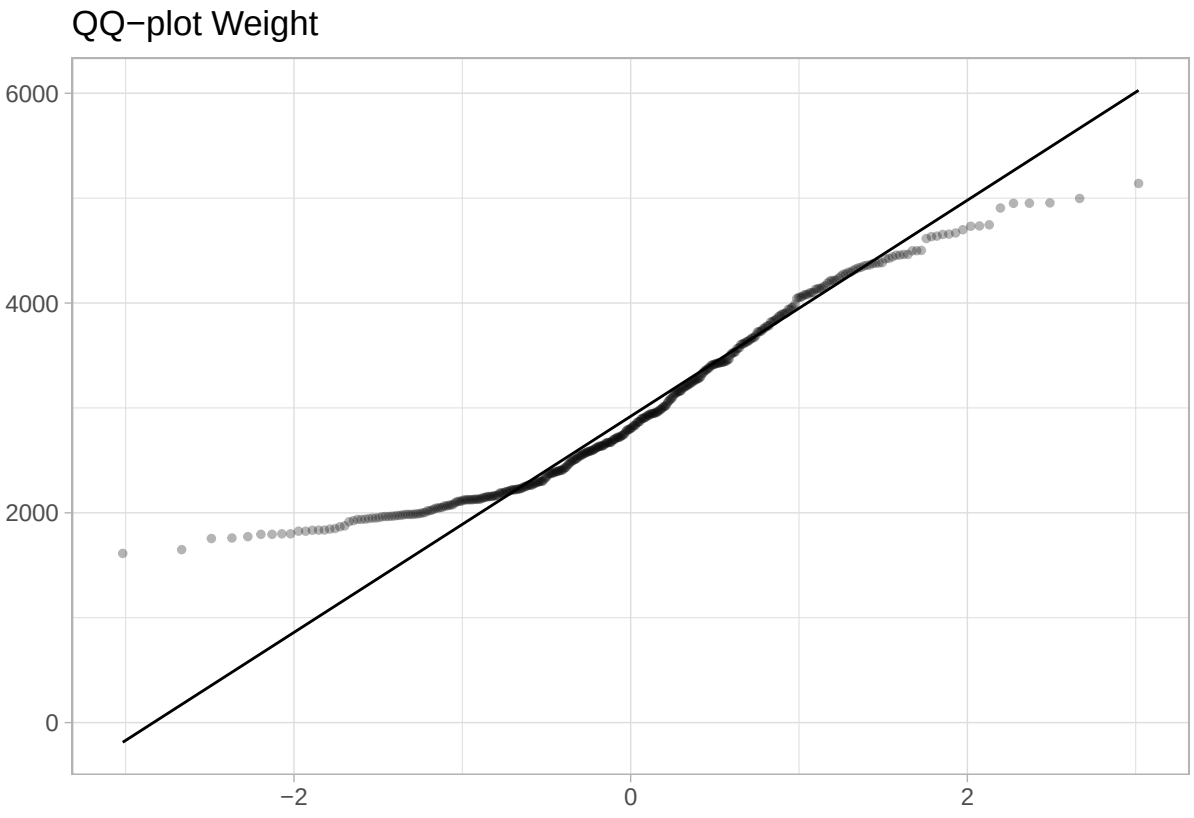
Se muestra aquí como no hay que dejarse engañar por los gráficos, puesto que Acceleration parecía seguirla. El p-value de Acceleration está muy cerca del umbral (0.03 vs 0.05). Es bastante probable de que la parte central derecha de la distribución sea la causante de no asegurar la normalidad.

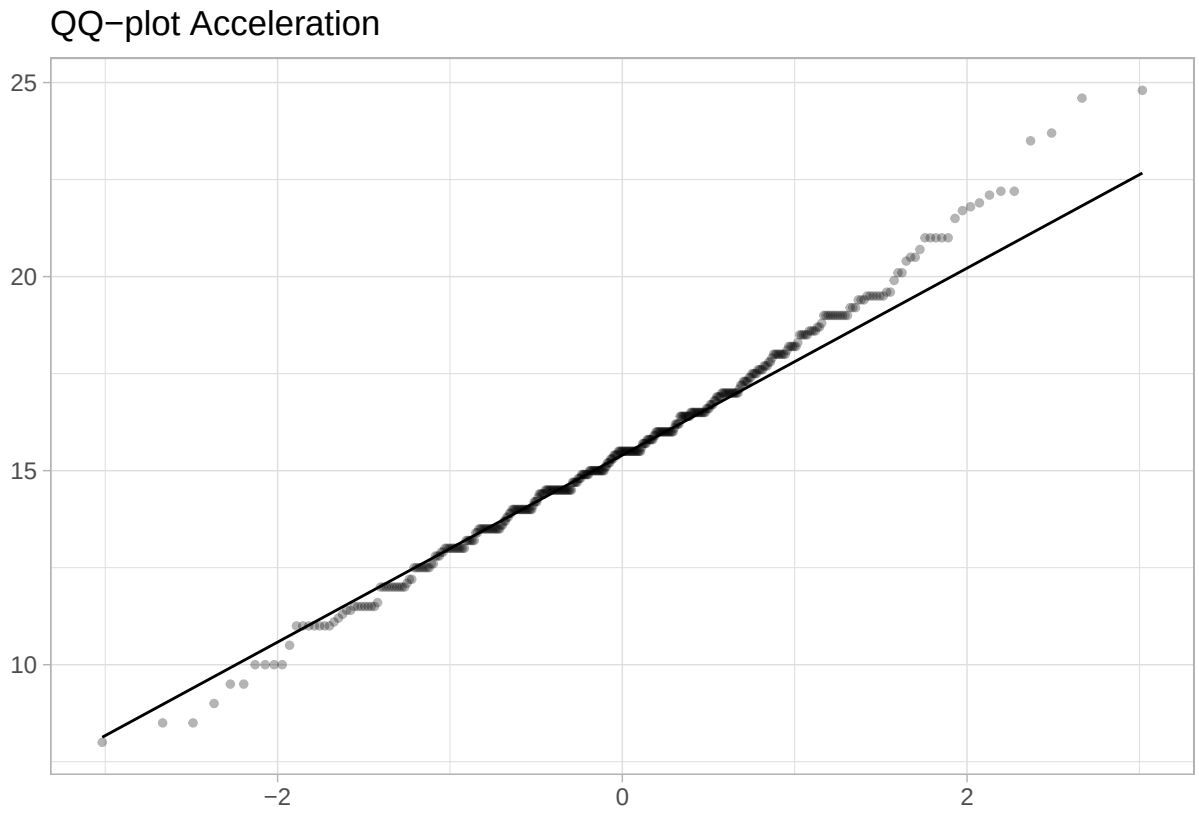
Vamos a mostrarlo con gráficos Q-Q para verlo mejor:

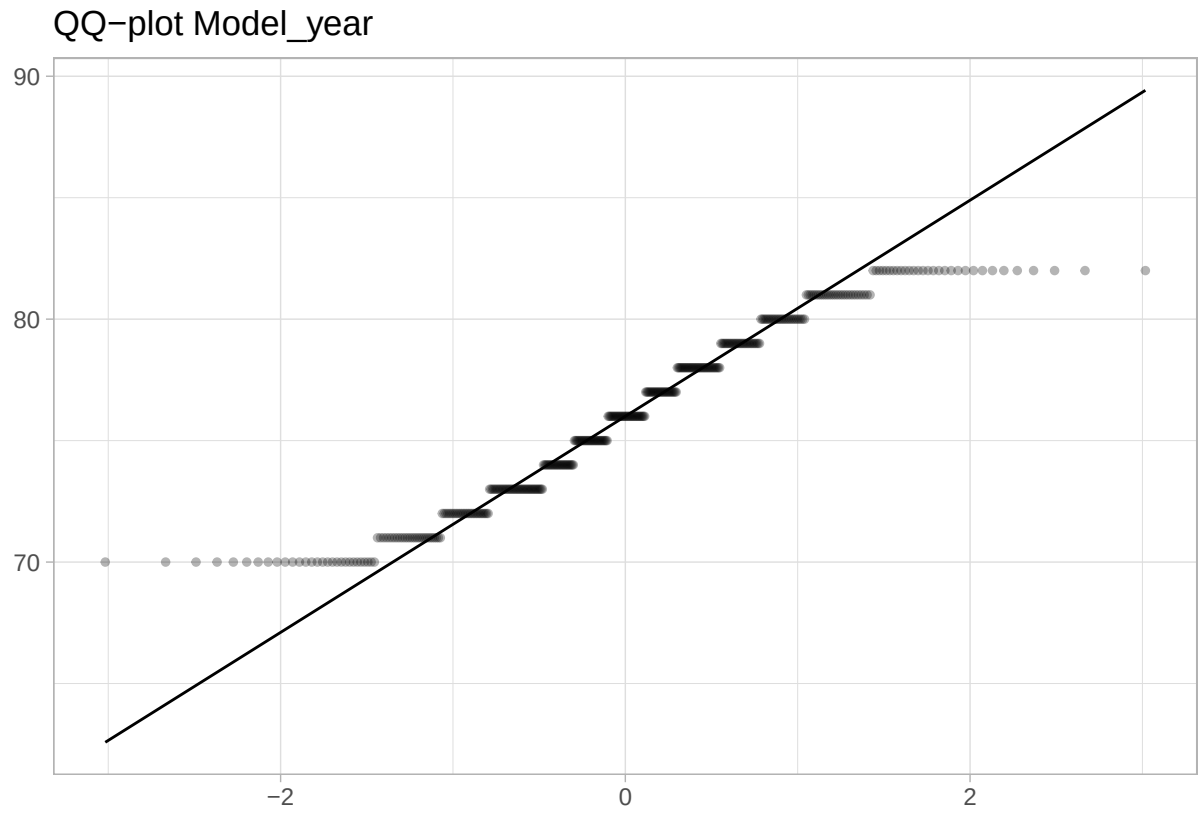
```
plt <- list(length = length(names))

x<-rnorm(100, mean=0, sd=1)

for (i in 1:length(names)) {
  ggplot(auto, aes_string(sample=names[i])) +
    stat_qq(alpha=.3, fill=colors[i], size=1) +
    stat_qq_line() +
    labs(title="", x="", y="") +
    theme_light() -> plt[[i]]

  print(plt[[i]] + labs(title=sprintf("QQ-plot %s", names[i]), x=""))
}
```

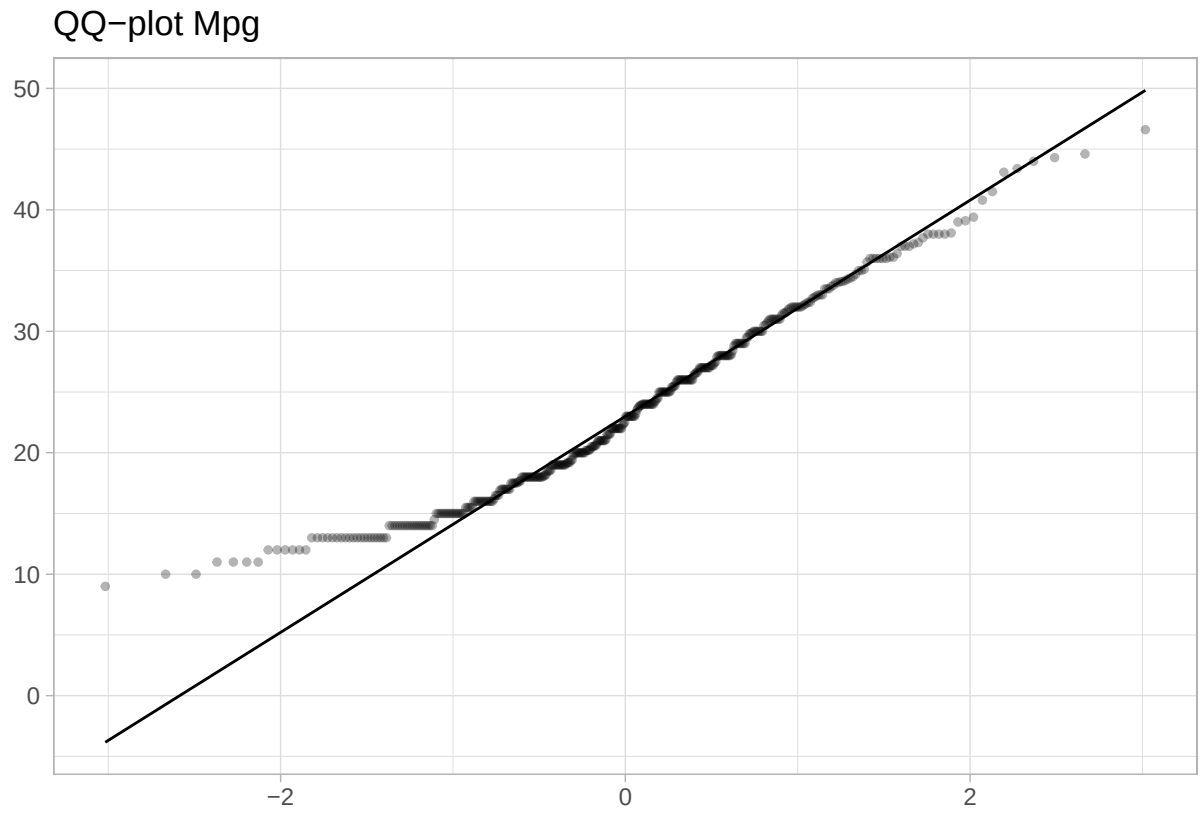

QQ−plot Displacement
600
400
200
0
−200
−2
0
2


QQ−plot Horse_power
200
150
100
50
0
−2
0
2

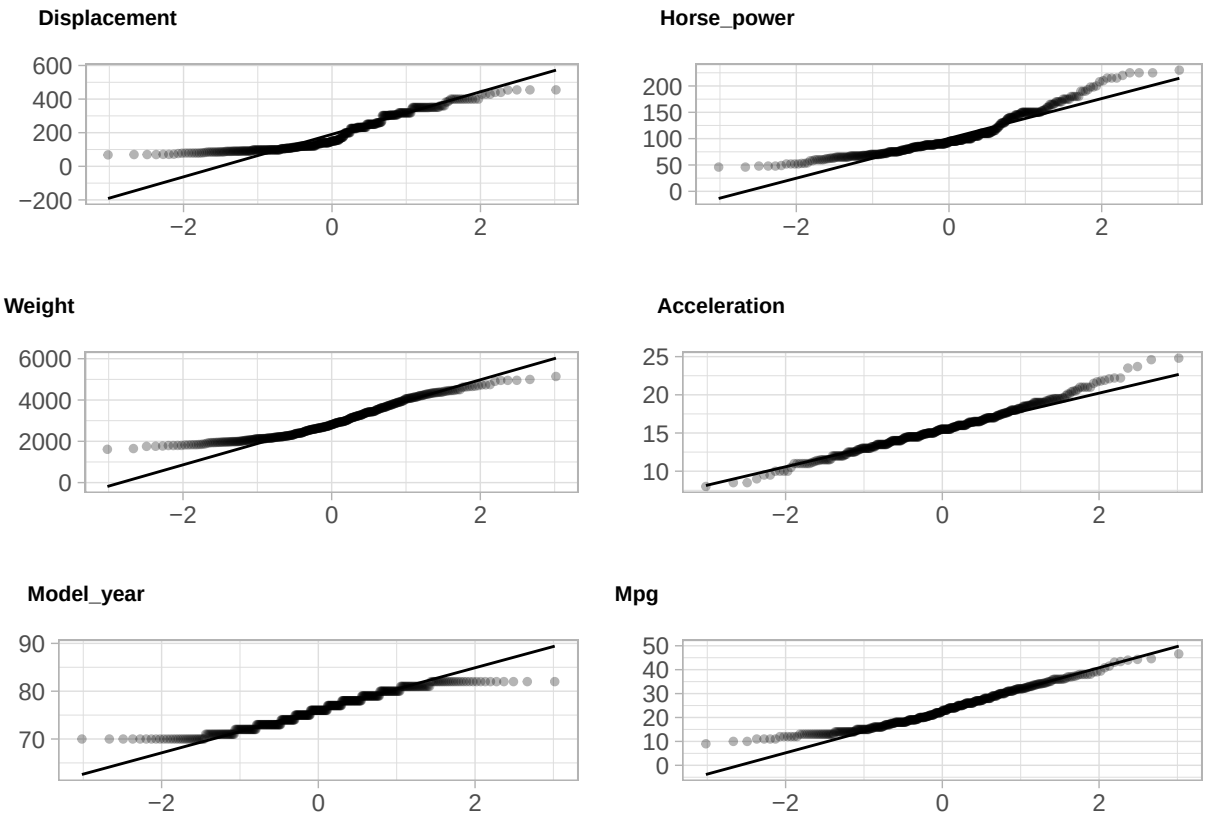

QQ−plot Weight
6000
4000
2000
0
−2
0
2


QQ−plot Acceleration
25
20
15
10
−2
0
2


QQ−plot Model_year
90
80
70
−2
0
2




```
plot_grid(plotlist=plt, ncol=2, labels = names, label_size = 8)
```



Estos gráficos Q-Q nos muestran más claramente que las variables no siguen distribuciones normales. La distribución de Acceleration es la que más se asemeja y eso lo vemos en el estadístico de Shapiro, pero en la cola superior existe una diferencia significativa que hace que el test rechace.

Skewness:

```
skewCols <- find_skewness(auto)
colnames(auto)[skewCols]
```

```
[1] "Displacement" "Horse_power"  "Weight"
```

```
cat("Displacement: ")
skewness(auto$Displacement)
cat("Horse_power: ")
skewness(auto$Horse_power)
cat("Weight: ")
skewness(auto$Weight)
cat("Mpg: ")
skewness(auto$Mpg)
```

```
Displacement: [1] 0.6989813
Horse_power: [1] 1.083161
Weight: [1] 0.5175953
Mpg: [1] 0.4553414
```

Sobre la skewness, tal y como se había visto en las gráficas, algunas de las variables la tienen, en los 3 casos positivas (hacia la izquierda).

Los plots nos han dado idea de que Mpg tiene cierta skewness, pero cae por debajo del umbral de 0.5.

Transformaciones

Tampoco vemos necesario crear variables nuevas a partir de las vistas, por el conocimiento que tenemos del problema parece que las variables son coherentes.

Las transformaciones necesarias para pasar a una distribución normal dependen de la variable en cuestion. Primero debemos averiguar que tipo de distribución siguen.

De todas maneras, los métodos utilizados para regresión (regresión lineal y KNN) no asumen ninguna forma para la distribución de los datos, por lo que no es necesario aplicar nada.

Algunas parecen tener una distribución exponencial

```
# auto_transform <- preProcess(auto[,skewCols], method=c("YeoJohnson"))
# auto_norm <- predict(auto_transform, auto[,skewCols])

# auto_transform <- preProcess(auto[,1:6], method=c("scale", "center"))
auto_transform <- preProcess(auto[,1:6], method=c("YeoJohnson","scale", "center"))
auto_norm <- predict(auto_transform, auto[,1:6])

summary(auto_norm)
```

```
  Displacement      Horse_power          Weight          Acceleration
 Min.   :-1.8856   Min.   :-2.59657   Min.   :-2.19582   Min.   :-3.08413
 1st Qu.:-0.8847   1st Qu.:-0.77280   1st Qu.:-0.88990   1st Qu.:-0.61672
 Median :-0.1367   Median :-0.07186   Median :-0.02988   Median : 0.02508
 Mean   : 0.0000   Mean   : 0.00000   Mean   : 0.00000   Mean   : 0.00000
 3rd Qu.: 0.9411   3rd Qu.: 0.77159   3rd Qu.: 0.84752   3rd Qu.: 0.56609
 Max.   : 1.7103   Max.   : 2.16086   Max.   : 1.95352   Max.   : 3.03480
   Model_year           Mpg
 Min.   :-1.6431   Min.   :-2.45780
 1st Qu.:-0.8047   1st Qu.:-0.79553
 Median : 0.0172   Median : 0.04412
 Mean   : 0.0000   Mean   : 0.00000
 3rd Qu.: 0.8236   3rd Qu.: 0.78143
 Max.   : 1.6154   Max.   : 2.32155
```

Para la variable Acceleration aplicando una transformación de YeoJohnson es suficiente para pasarla a una normal.

```
normality(auto)
```

```
Warning: `cols` is now required when using unnest().
Please use `cols = c(statistic)`
```

vars	statistic	p_value	sample
Displacement	0.8818359	0.0000000	392
Horse_power	0.9040975	0.0000000	392
Weight	0.9414661	0.0000000	392
Acceleration	0.9918671	0.0305289	392
Model_year	0.9469666	0.0000000	392
Mpg	0.9671696	0.0000001	392

Aunque para regresión lineal no es absolutamente necesario, podemos estandarizar los datos a media 0 y dev 1, facilitando un poco los cálculos. La inferencia estadística de la regresión no va a variar, por lo que es conveniente hacerlo. Haciendo esto debemos tener cuidado a la hora de interpretar los resultados de la regresión para no confundirnos.

Outliers

Como hemos visto anteriormente en los boxplots, las únicas variables con valores muy alejados del centro de la distribución son Acceleration y Horse_power.

Por el significado del problema, probablemente estos posibles outliers correspondan a coches de alta gama o potentes en la época. Esto tampoco lo podemos asegurar puesto que contamos con pocas características, pero se considera un razonamiento coherente. Además, puesto que los valores caen dentro de los rangos posibles para coches de la época, podemos descartar que sean errores de medida.

Deberíamos decidir si mantener o no estas instancias. Como en nuestro caso se nos ha pedido predecir el consumo Mpg, sin darnos consideraciones sobre los tipos/gamas de coches a los que se enfoca, proseguimos dejándo estas filas.

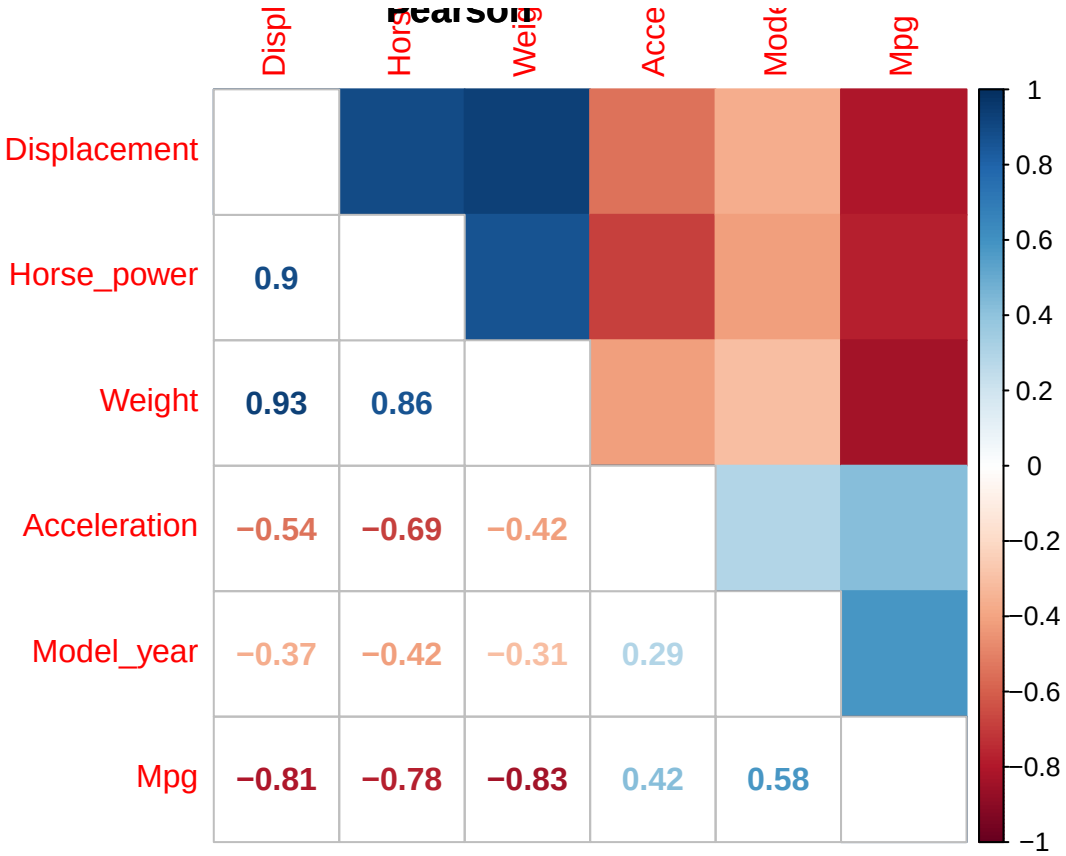
Análisis de correlación

Tenemos que tener en cuenta que las variables no siguen distribuciones normales. Aunque el coeficiente de Pearson no asume normalidad (si asume varianza y covarianza finitas), podemos usar el coeficiente de Kendall para los cálculos. Independientemente del método usado vamos a obtener las mismas correlaciones en este dataset, solo varía la fuerza con la que se dan.

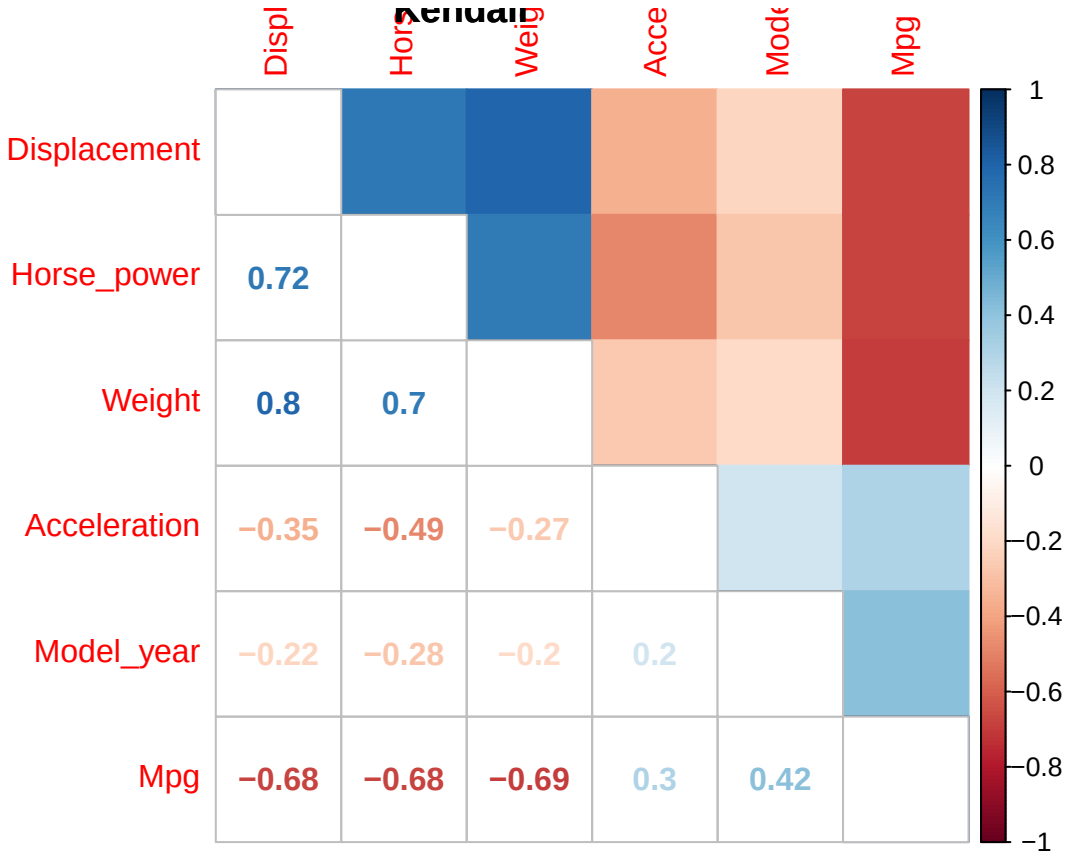
Para regresión la correlación en los datos no es preocupante. Al contrario, podría haber información (poca, pero alguna cantidad) que se aporte y nos ayude en el problema. Además, la propia metodología de selección de variables en el modelo multivariable nos ayudará a descartar aquellas variables que no sean necesarias como regresor.

Corrplot

```
corrplot.mixed(cor(auto), tl.pos="lt", upper="color", title="Pearson")
```



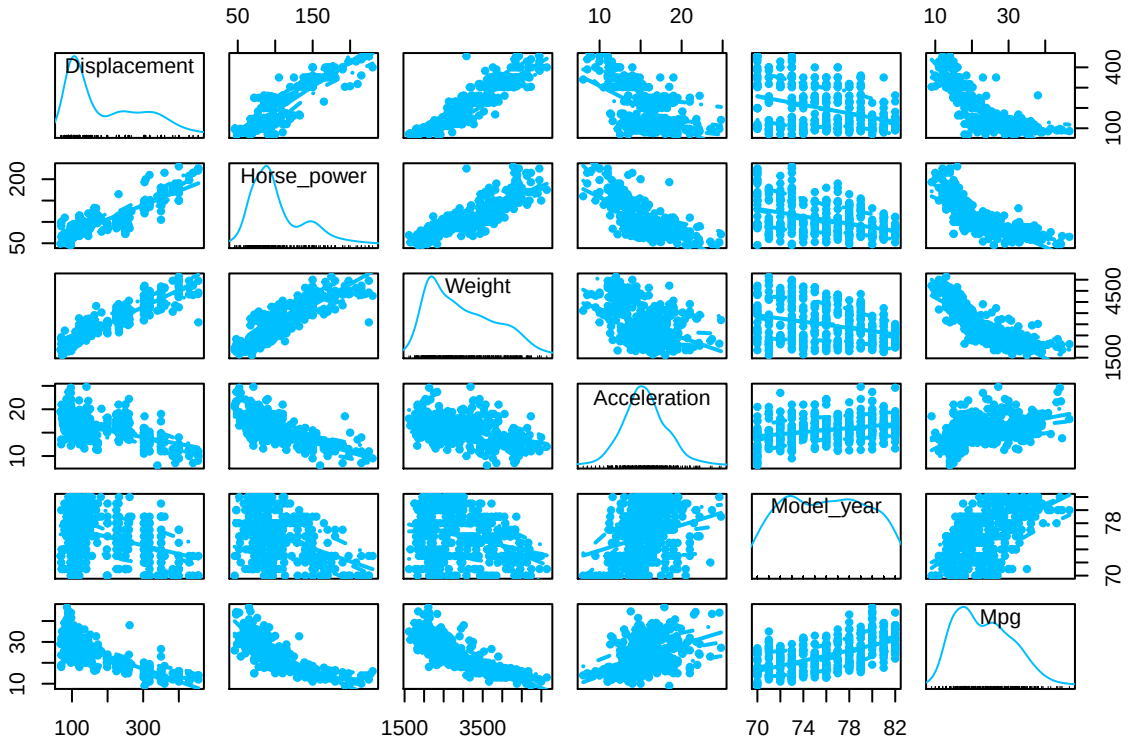

```
corrplot.mixed(cor(auto, method="kendall"), tl.pos="lt", upper="color", title="Kendall")
```



Estas gráficas nos dicen que existe una alta correlación en el dataset, generalmente entre todas las variables (a excepción de Model_year), pero extremadamente fuerte en las parejas:

1. Horse_power & Displacement
2. Weight & Displacement
3. Weight & Horse_power
4. Acceleration & Horse_power
5. Mpg & Horse_power
6. Mpg & Displacement
7. Mpg & Weight

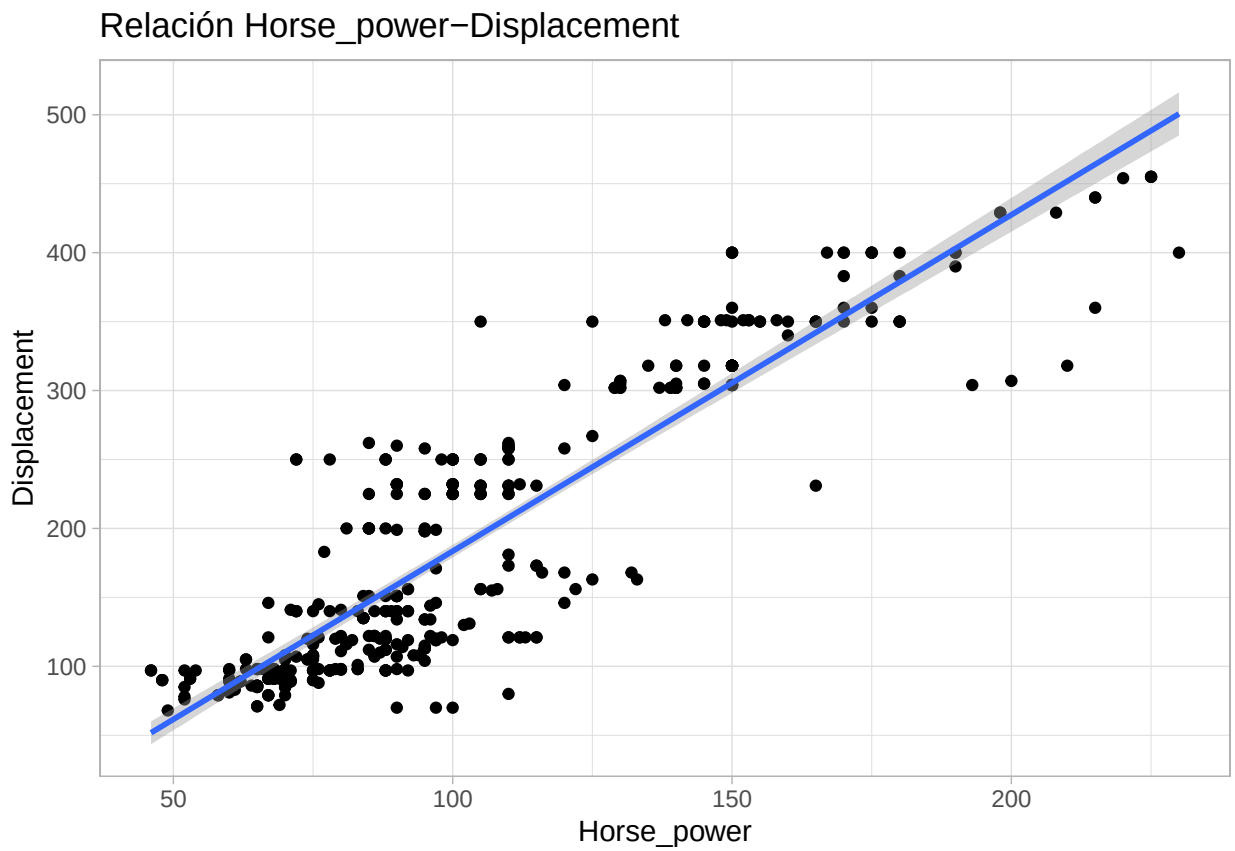
```
scatterplotMatrix(auto, pch=20, col="deepskyblue")
```



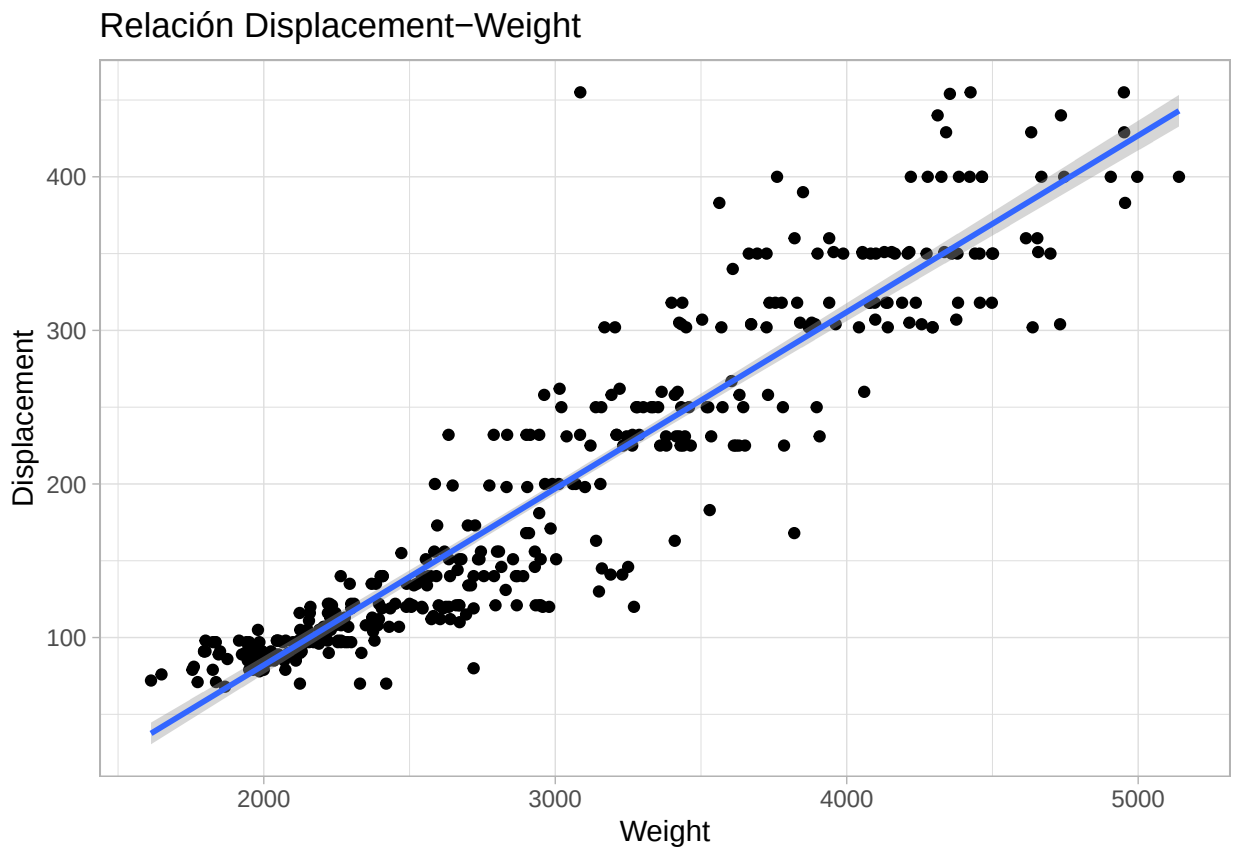
El scatterplot anterior nos muestra mejor la forma de estas correlaciones. Vemos que en todos los casos en los que se da una correlación positiva existe una tendencia lineal entre los datos de ambas variables, y en las negativas una tendencia logarítmica.

Vamos a mostrar algunas Positivas:

```
ggplot(auto, aes(x=Horse_power, y=Displacement)) +
  geom_point() +
  geom_smooth(formula = y~x, method=glm) +
  labs(title="Relación Horse_power-Displacement") +
  theme_light()
```

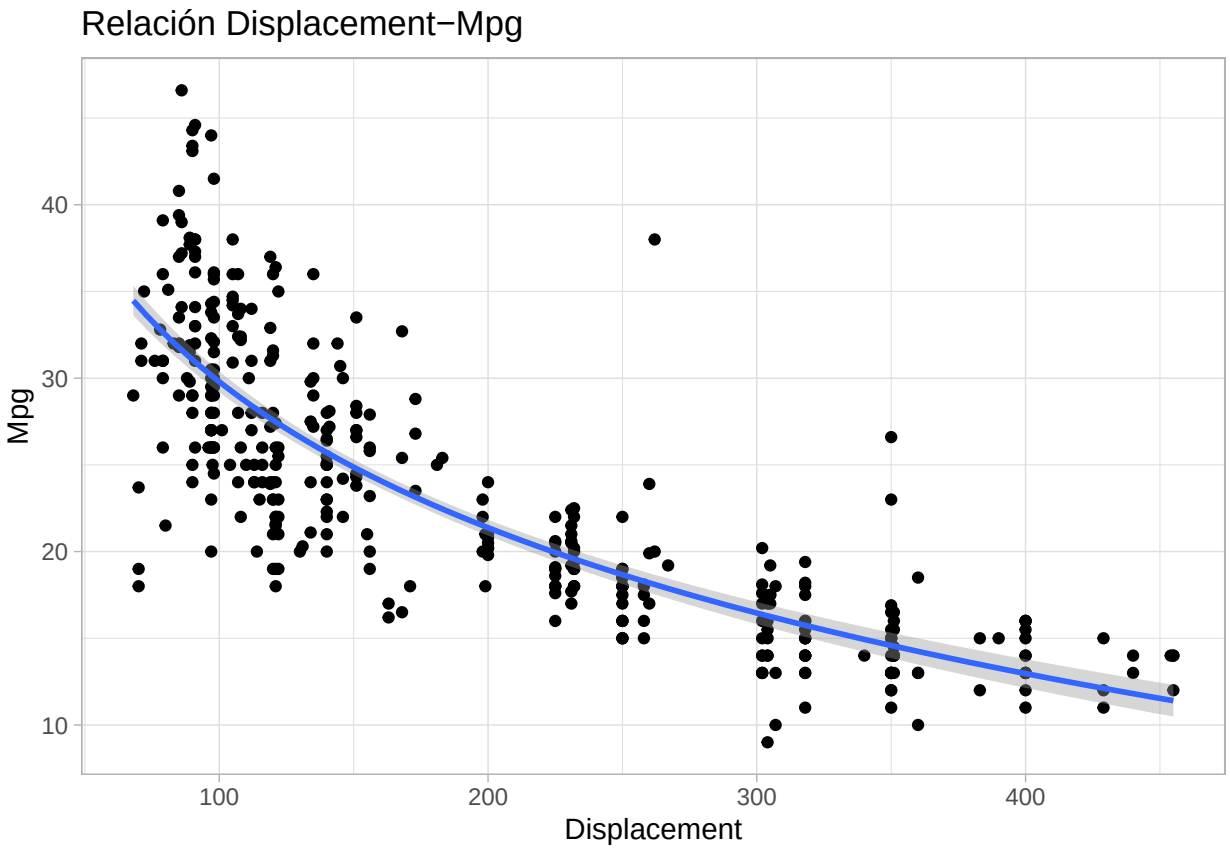



```
ggplot(auto, aes(x=Weight, y=Displacement)) +
  geom_point() +
  geom_smooth(formula = y~x, method=glm) +
  labs(title="Relación Displacement-Weight") +
  theme_light()
```

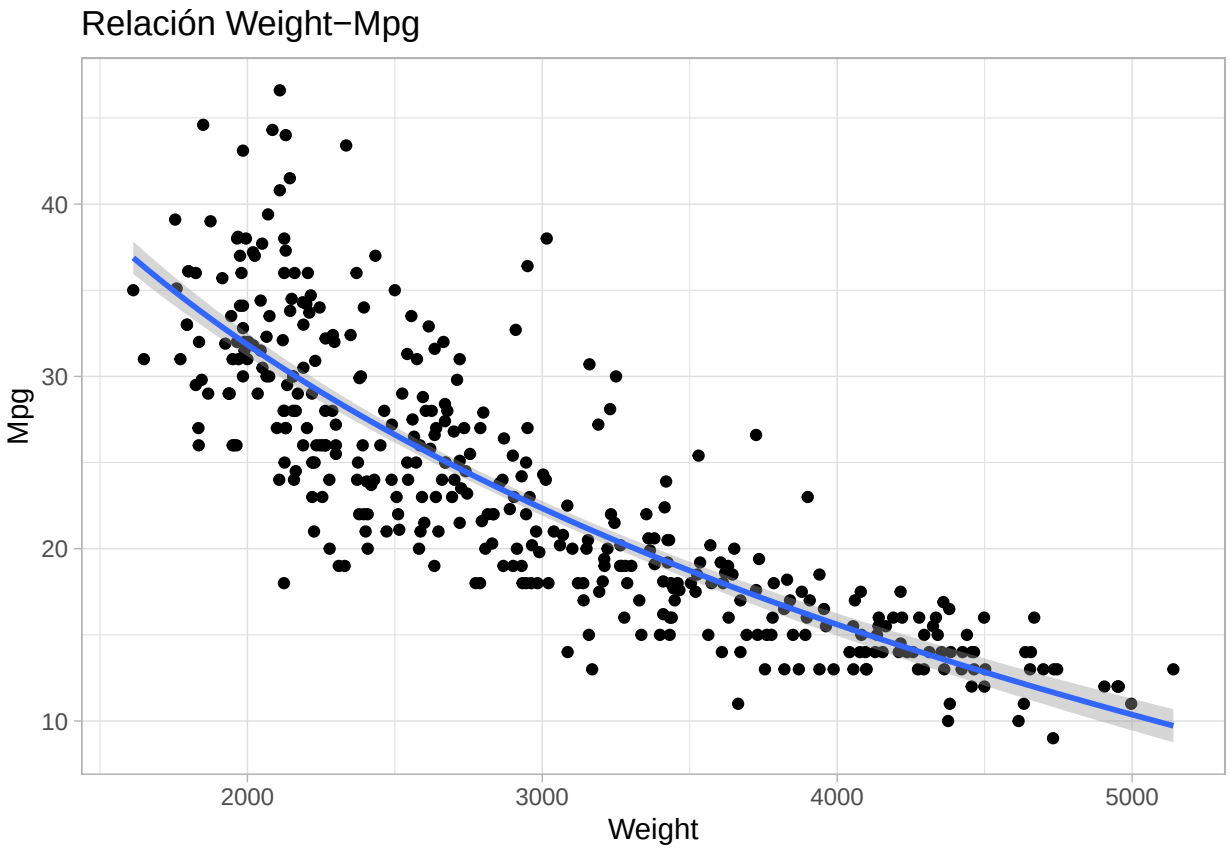


Negativas

```
ggplot(auto, aes(x=Displacement, y=Mpg)) +
  geom_point() +
  geom_smooth(formula = y~log(x), method=glm) +
  labs(title="Relación Displacement-Mpg") +
  theme_light()
```

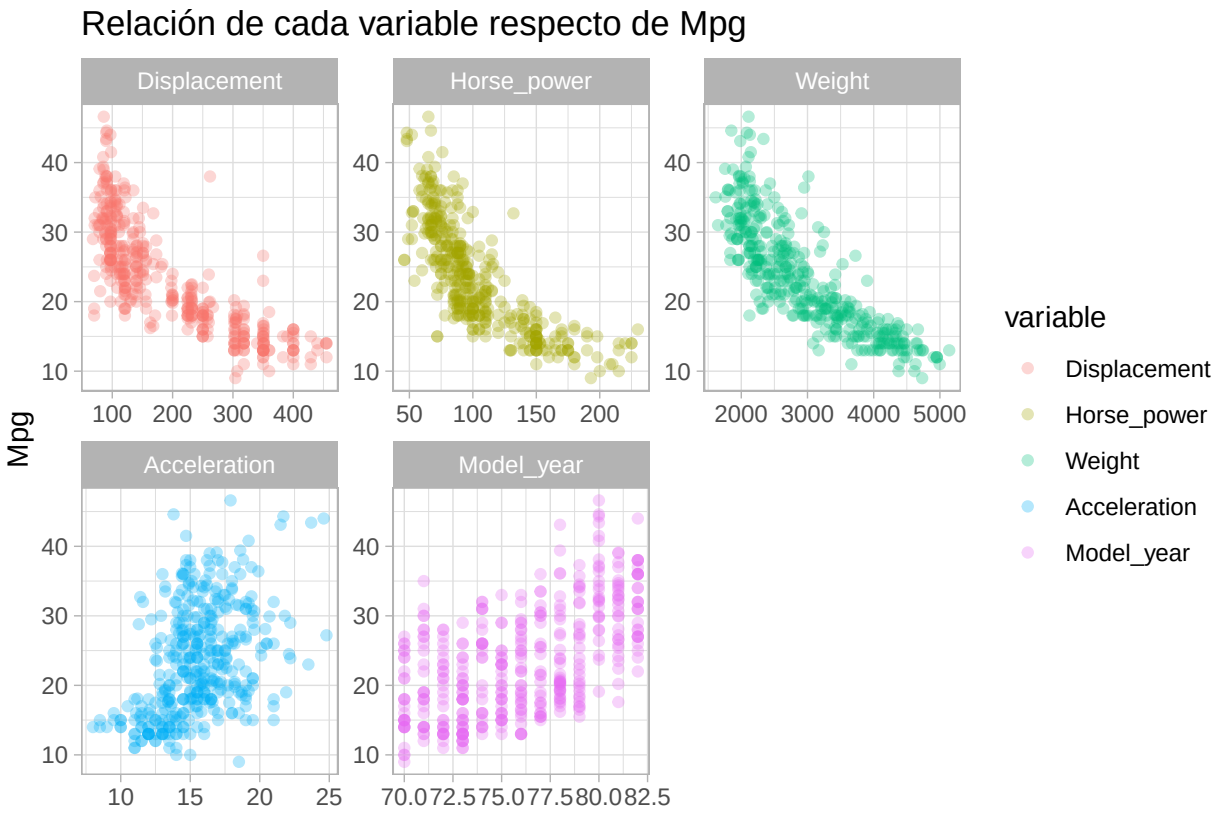



```
ggplot(auto, aes(x=Weight, y=Mpg)) +
  geom_point() +
  geom_smooth(formula = y~log(x), method=glm) +
  labs(title="Relación Weight-Mpg") +
  theme_light()
```



Previsualicación de las variables respecto a la salida

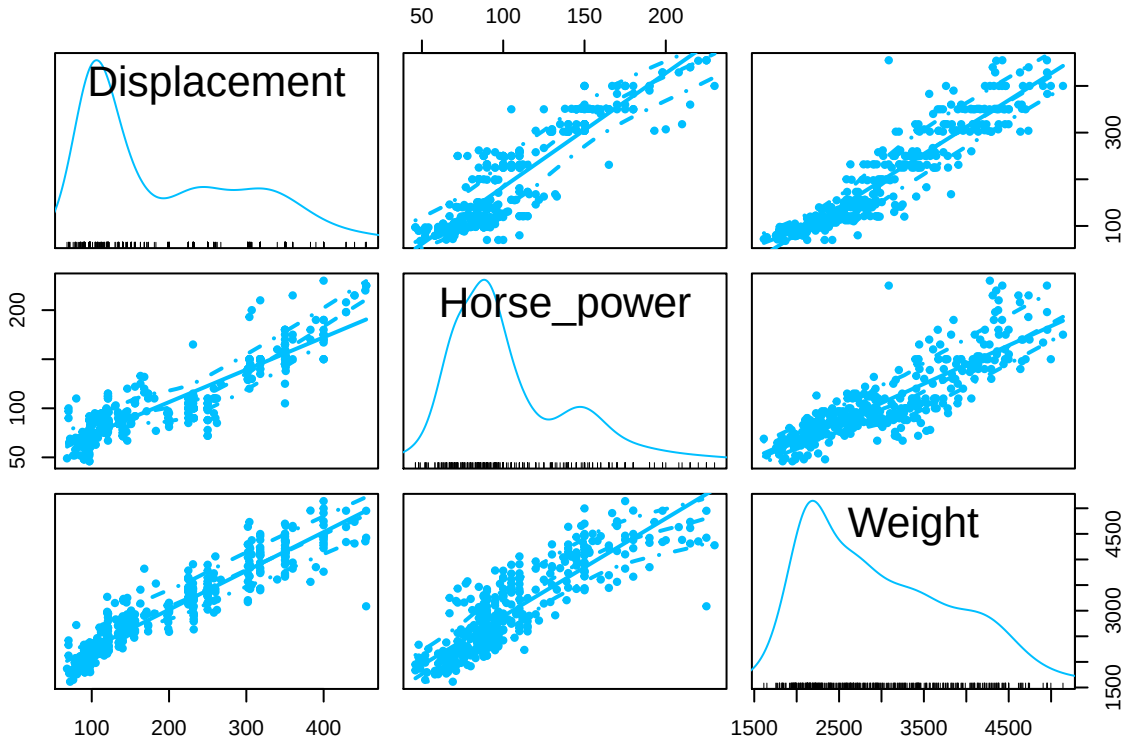
```
ggplot(melt(auto, "Mpg"), aes(x=value, y=Mpg, color=variable)) +
  geom_point(alpha=0.3) +
  facet_wrap(.~variable, scale="free") +
  labs(title="Relación de cada variable respecto de Mpg", x="") +
  theme_light()
```



Se aprecia alta correlación entre Displacement, Horse_power, Weight respecto de la salida.

Como habíamos supuesto en la hipótesis H.9, Horse_power podría depender de Displacement y Weight. Esta claro que la potencia de un motor va a depender de la cilindrada y el peso que tenga.

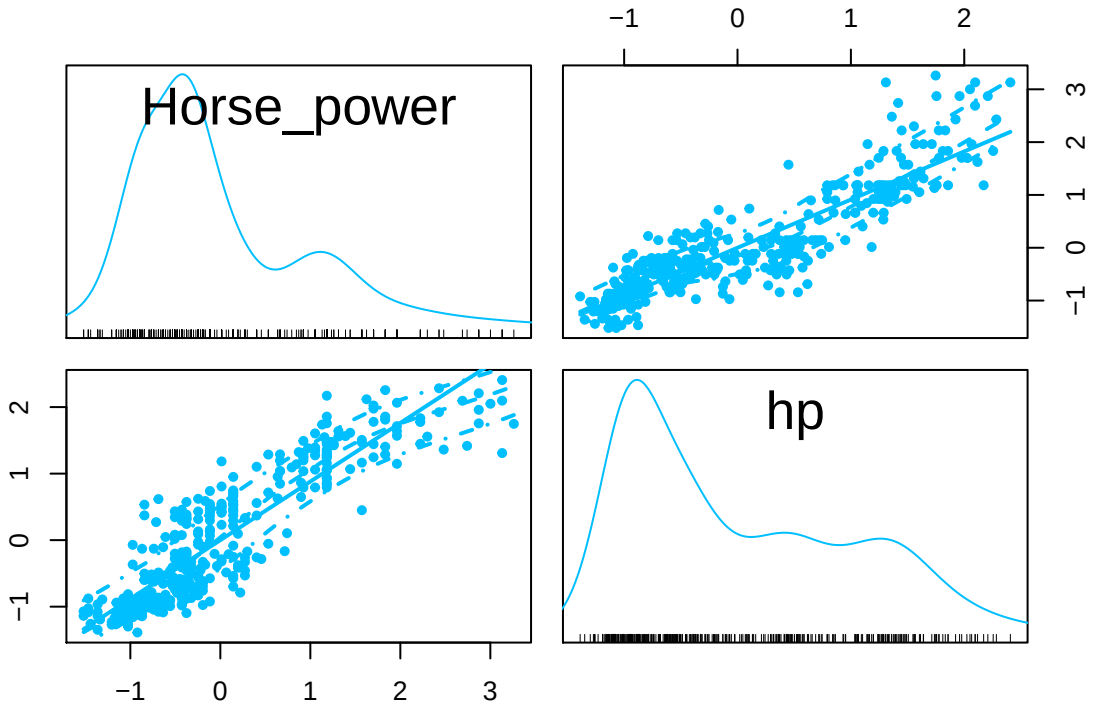
```
auto %>%
  dplyr::select(Displacement, Horse_power, Weight) %>%
  scatterplotMatrix(pch=20, col="deepskyblue")
```



Podemos apreciar como la función de densidad de Horse_power parece una ("MEDIANIZACIÓN") de las otras dos.

Vamos a intentar comprobarlo

```
scale(auto) %>%
  as.data.frame() %>%
  mutate(hp = (Displacement+Weight) / 2) %>%
  dplyr::select(Horse_power, hp) %>%
  scatterplotMatrix(pch=20, col="deepskyblue")
```



Viendo que no son tan similares como creíamos, buscamos diferentes fórmulas para el cálculo de los caballos de vapor, y vemos que las fórmulas son un poco más complejas y no tenemos exactamente los datos necesarios para utilizarlas (no se descarta que no se puedan deducir, pero no sería un cálculo evidente)

(poner fórmulas https://www.ajdesigner.com/phphorsepower/horsepower_equation_trap_speed_method_increase_horsepower.php#:~:text=Solving%20for%20the%20change%20in,the%20vehicle%2C%20driver%20and%20passenger.)

Tratamiento de variables

Para este dataset, al ser casi todas las variables numéricas continuas, existen pocos tratamientos que aplicar.

No tenemos variables categóricas que transformar.

Para añadir interpretabilidad, podríamos agrupar la variable Weight en intervalos, pero puesto que vamos a aplicar regresión sería más conveniente realizarlo con los resultados finales.

Ordenaciones

Volvemos a mostrar la cabecera de los datos:

```
head(auto)
```

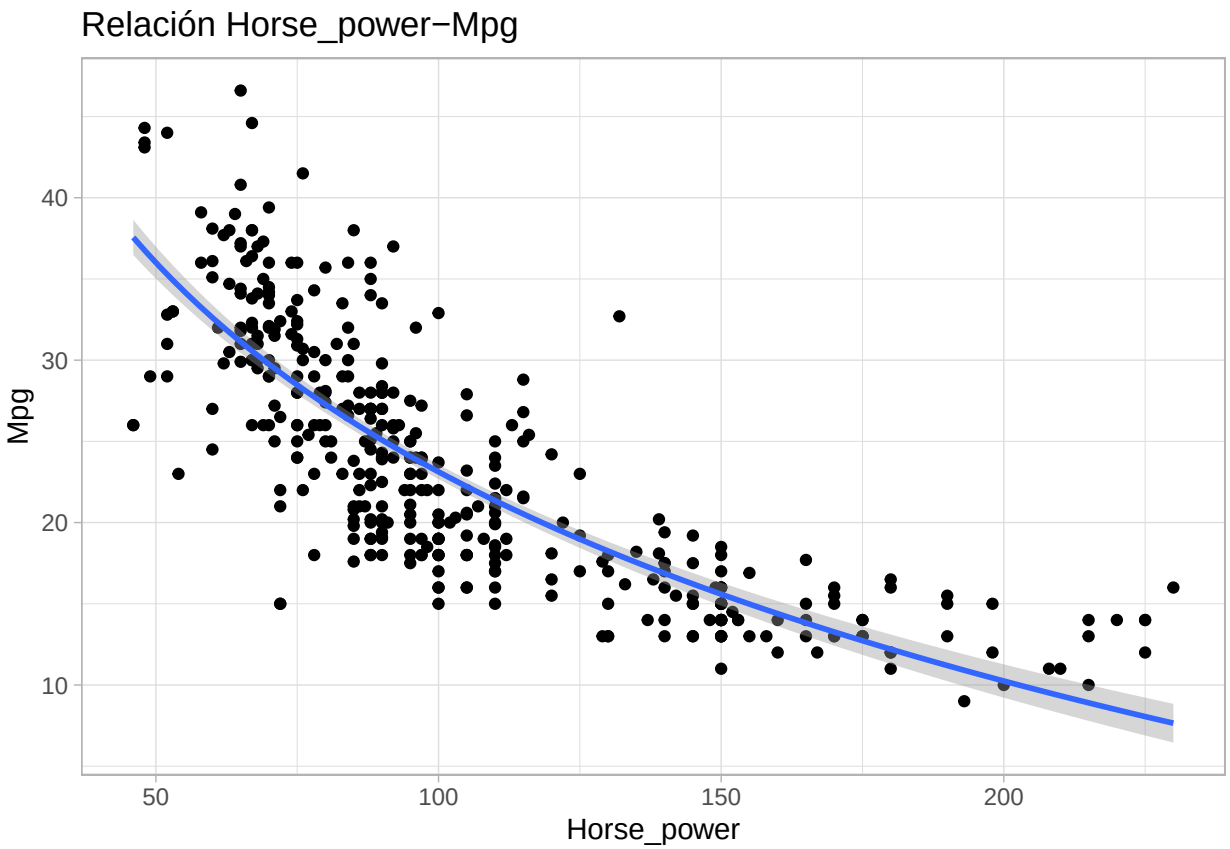
Displacement	Horse_power	Weight	Acceleration	Model_year	Mpg
91	70	1955	20.5	71	26.0
232	100	2789	15.0	73	18.0
350	145	4055	12.0	76	13.0
318	140	4080	13.7	78	17.5
113	95	2372	15.0	70	24.0
97	60	1834	19.0	71	27.0

En este caso no es necesario aplicar ninguna reorganización. Cada variable ocupa su propia columna, y contiene un único tipo de información, con unidades de observación diferentes No existe ninguna relación entre variables sobre la información que codifican (en el sentido de que podrían agruparse).

Resolución de hipótesis Nos habíamos planteado las siguientes hipótesis

- H.1: Horse_power puede influir en Mpg: A más potencia, más consumo.

```
ggplot(auto, aes(x=Horse_power, y=Mpg)) +
  geom_point() +
  geom_smooth(formula = y~log(x), method=glm) +
  labs(title="Relación Horse_power-Mpg") +
  theme_light()
```

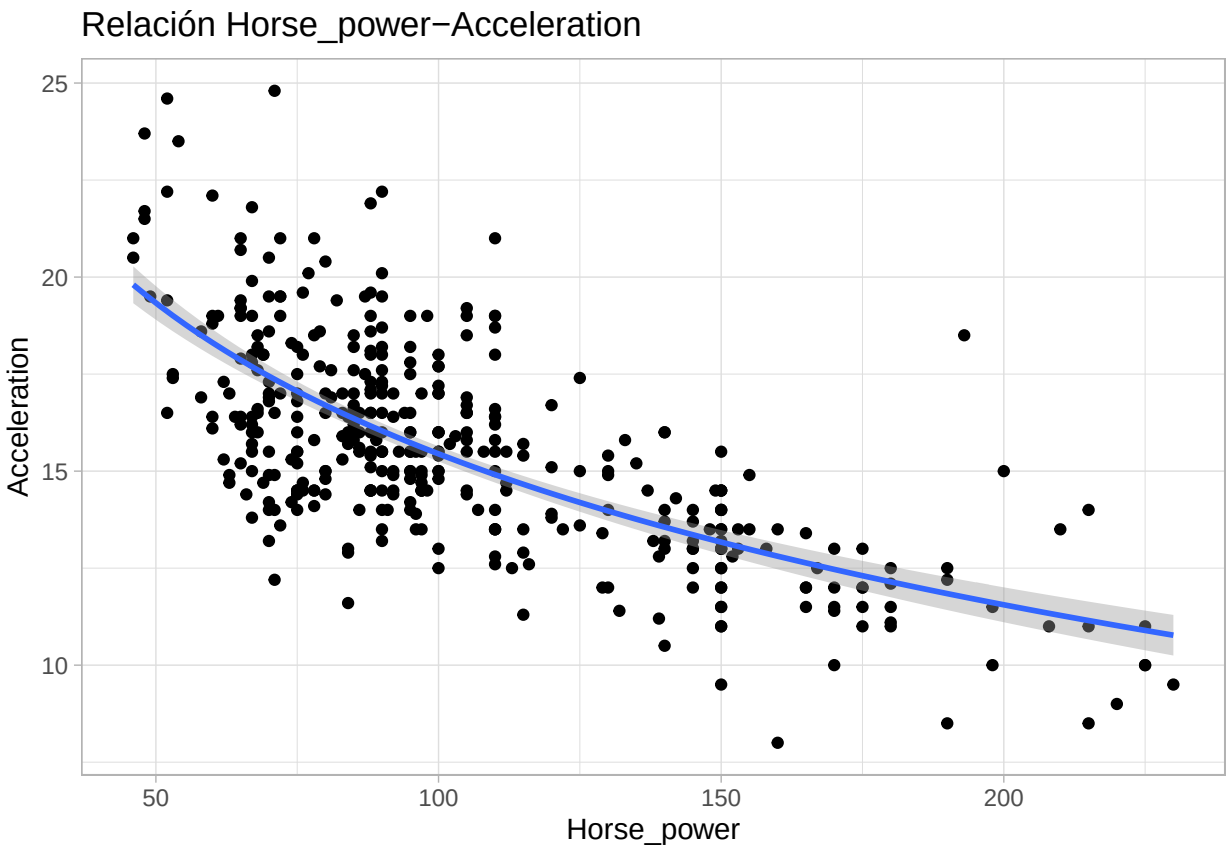


Con el plot y los resultados de la matriz de correlación queda claro que existe una correlación negativa entre estas dos variables. Por tanto, podemos considerar Horse_power como un buen candidato para la regresión

- H.2: Weight debe influir en Mpg: Un coche más pesado debería consumir más idem. a la hipótesis anterior, lo hemos visto anteriormente en la figura X

- H.3: Debería haber correlación entre displacement (cilindrada) con horse y acceleration La hemos referenciado anteriormente
- H.4: Horse y acceleration podrían estar relacionadas

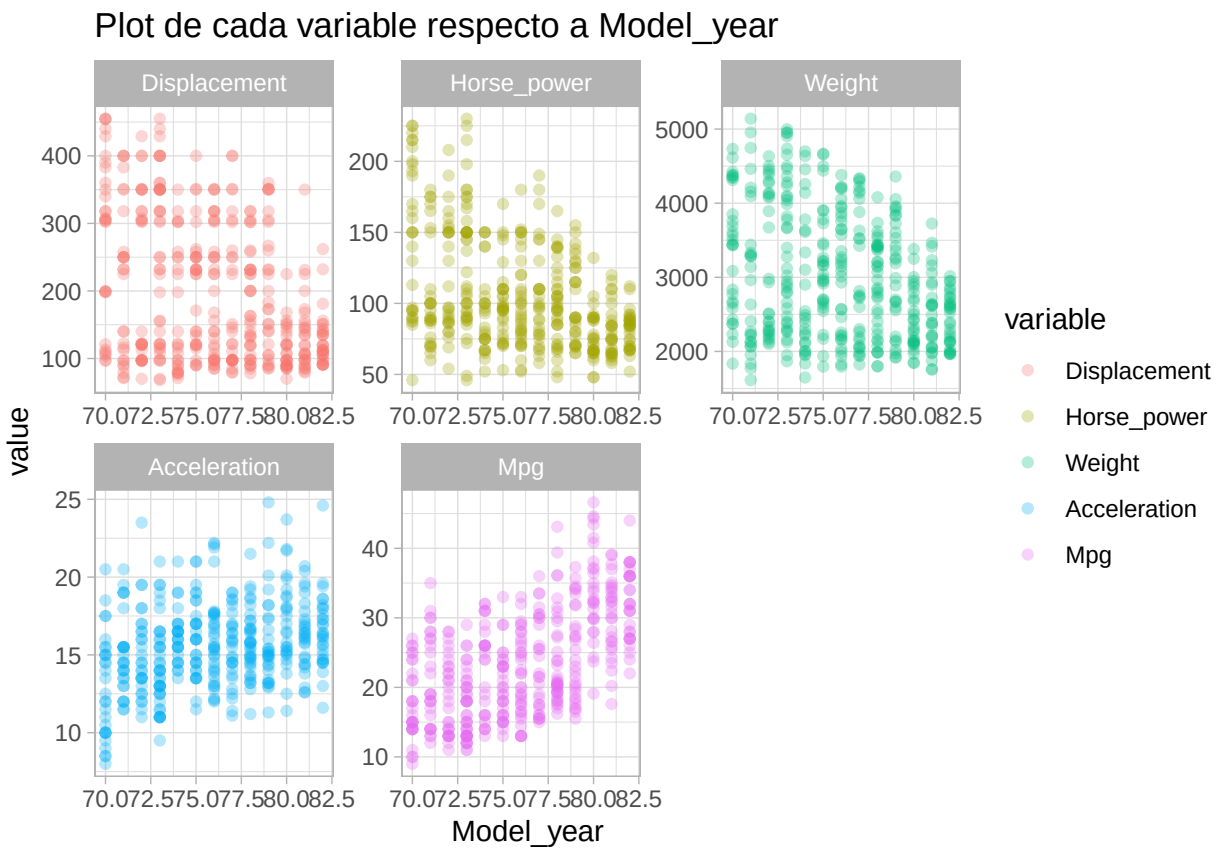
```
ggplot(auto, aes(x=Horse_power, y=Acceleration)) +
  geom_point() +
  geom_smooth(formula = y~log(x), method=glm) +
  labs(title="Relación Horse_power-Acceleration") +
  theme_light()
```



idem. se aprecia una correlación logarítmica entre las dos variables. Similarmente a lo ocurrido con la hipótesis anterior, esto puede ser un problema para nuestro problema de regresión.

- H.5: Viendo que contamos con un rango pequeño de años, no debería haber un cambio significativo de prestaciones entre años.

```
ggplot(melt(auto, "Model_year"), aes(y=value, x=Model_year, color=variable)) +
  geom_point(alpha=0.3) +
  facet_wrap(.~variable, scale="free") +
  labs(title="Plot de cada variable respecto a Model_year") +
  theme_light()
```



Existe una alta dispersión de los datos en cada una de las variables, pero aún así se aprecia tendencias en las variables. Acceleartion y Mpg tienden a aumentar, y Displacement, Horse_power y Weight tienden a disminuir. También vemos que la dispersión en las prestaciones de los coches disminuyen ligeramente.

Podemos creer en principio que puede deberse a un decremento del número de instancias con el paso de los años, pero recordamos que en general los datos están repartidos equitativamente

```
table(auto$Model_year)
```

```
70 71 72 73 74 75 76 77 78 79 80 81 82
29 27 28 40 26 30 34 28 36 29 27 28 30
```

Podemos ver cómo varían los rangos para cada año

```
years <- auto %>% group_split(Model_year)

for (y in years) {
  cat("Year: ")
  y$Model_year[1] %>% cat()
  y %>% apply(2, range) %>% as.data.frame() %>% print()
}
```

```
Year: 70  Displacement Horse_power Weight Acceleration Model_year Mpg
1           97          46   1835          8.0         70   9
2          455         225   4732         20.5         70  27
Year: 71  Displacement Horse_power Weight Acceleration Model_year Mpg
1           71          60   1613         11.5         71  12
2          400         180   5140         20.5         71  35
```

```
Year: 72  Displacement Horse_power Weight Acceleration Model_year Mpg
1           70          54   2100         11.0         72  11
2          429         208   4633         23.5         72  28
Year: 73  Displacement Horse_power Weight Acceleration Model_year Mpg
1           68          46   1867          9.5         73  11
2          455         230   4997         21.0         73  29
Year: 74  Displacement Horse_power Weight Acceleration Model_year Mpg
1           71          52   1649         13.5         74  13
2          350         150   4699         21.0         74  32
Year: 75  Displacement Horse_power Weight Acceleration Model_year Mpg
1           90          53   1795         11.5         75  13
2          400         170   4668         21.0         75  33
Year: 76  Displacement Horse_power Weight Acceleration Model_year Mpg
1           85          52   1795         12.0         76  13
2          351         180   4380         22.2         76  33
Year: 77  Displacement Horse_power Weight Acceleration Model_year Mpg
1           79          58   1825         11.1         77  15
2          400         190   4335         19.0         77  36
Year: 78  Displacement Horse_power Weight Acceleration Model_year  Mpg
1           78          48   1800         11.2         78 16.2
2          318         165   4080         21.5         78 43.1
Year: 79  Displacement Horse_power Weight Acceleration Model_year  Mpg
1           85          65   1915         11.3         79 15.5
2          360         155   4360         24.8         79 37.3
Year: 80  Displacement Horse_power Weight Acceleration Model_year  Mpg
1           70          48   1845         11.4         80 19.1
2          225         132   3381         23.7         80 46.6
Year: 81  Displacement Horse_power Weight Acceleration Model_year  Mpg
1           79          58   1755         12.6         81 17.6
2          350         120   3725         20.7         81 39.1
Year: 82  Displacement Horse_power Weight Acceleration Model_year Mpg
1           91          52   1965         11.6         82  22
2          262         112   3015         24.6         82  44
```

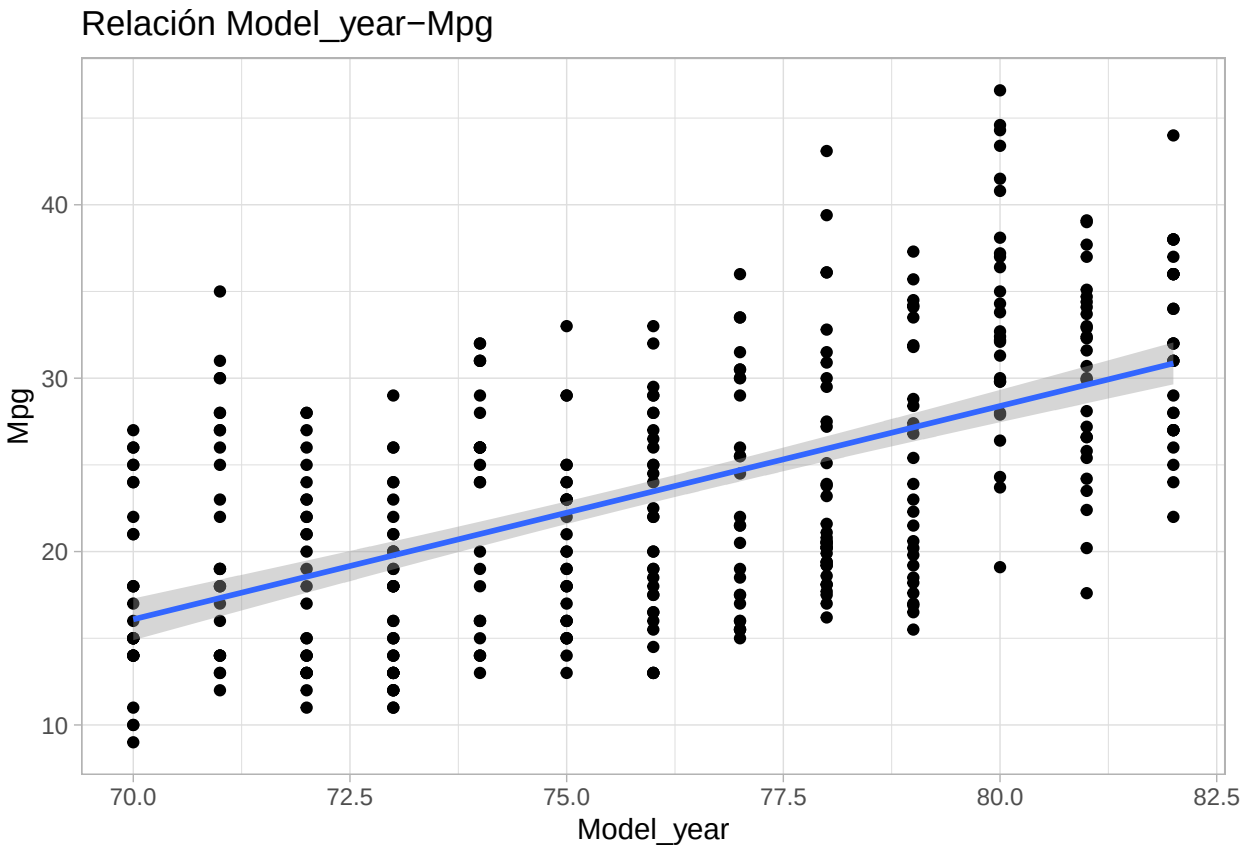
- H.6: Pero debería existir una tendencia de mejora de prestaciones con los años, incluyendo aumento de Displacement, Horse_power y Acceleration.

Ciertamente. Se ha comprobado en la hipótesis anterior.

- H.7: Model_year podría no mostrar relación con Mpg: Pese al paso de los años si contamos con diferentes tipos de vehículos (todoterrenos, familiares, deportivos...) podría haber un consumo dispar. (Si existiera tendencia, viendo que los años son de las últimas décadas del siglo XX, podría ir el consumo hacia abajo)

Hemos visto que existe tendencia, lineal con gran dispersión, y positiva.

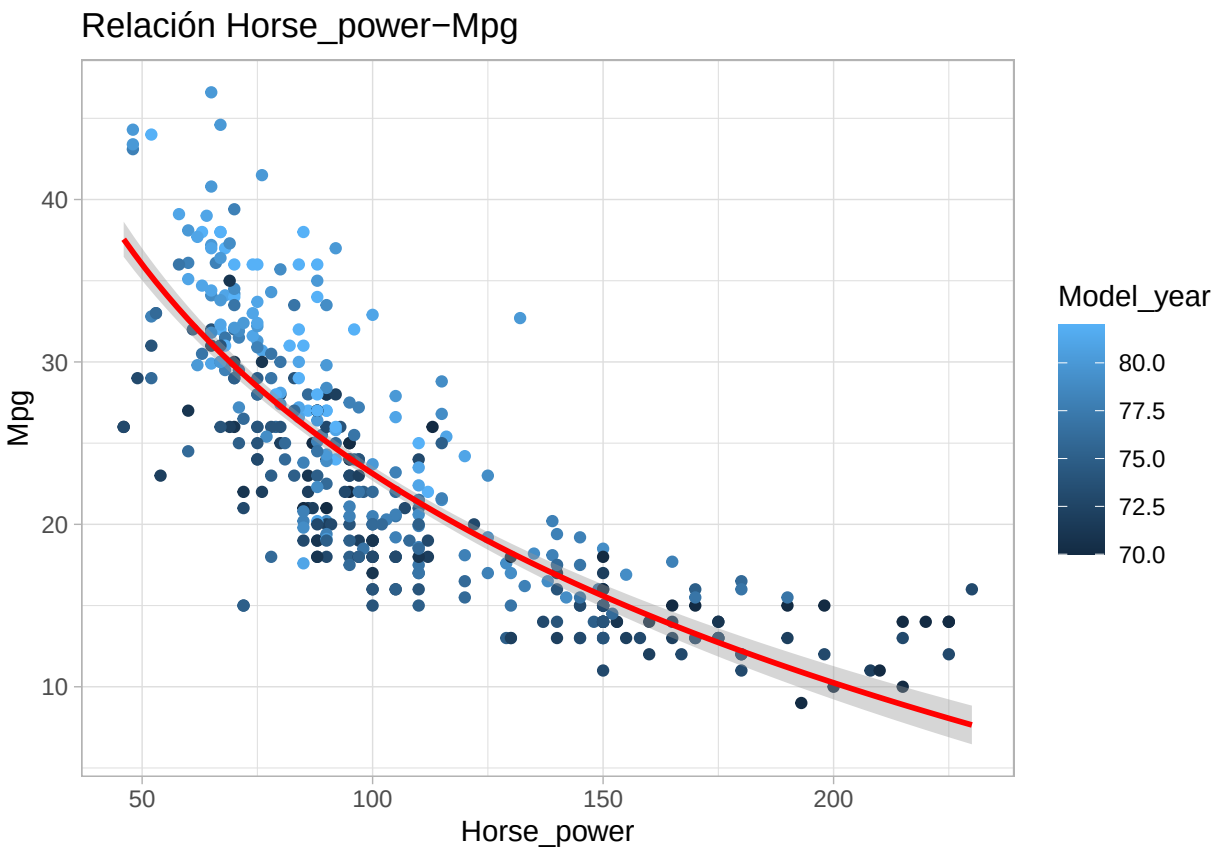
```
ggplot(auto, aes(x=Model_year, y=Mpg)) +
  geom_point() +
  geom_smooth(formula = y~x, method=glm) +
  labs(title="Relación Model_year-Mpg") +
  theme_light()
```



Por desgracia no contamos información sobre los modelos de los coches

Podemos ver como se ubican los diferentes años en un plot Horse_power vs Mpg

```
ggplot(auto, aes(x=Horse_power, y=Mpg, color=Model_year)) +
  geom_point() +
  geom_smooth(formula = y~I(log(x)), color="red", method=glm) +
  labs(title="Relación Horse_power-Mpg") +
  theme_light()
```



Y vemos que no se puede afirmar la hipótesis, los coches están entremezclados por diferentes años

- H.8: Esta última hipótesis se puede aplicar al resto de variables, indicándonos que Model_year no debería tener relevancia para este problema de regresión.

No podemos afirmar la hipótesis anterior y por consiguiente esta tampoco.

- H.9: Horse_power podría depender de las variables Displacement y Weight

Lo hemos comentado anteriormente

Conclusiones Como conclusiones podemos decir que tenemos un dataset altamente correlacionado, distribuído de forma no normal pero con la información bien representada. Existen relaciones fuertes entre las variables de entrada y de las de salida para la regresión que probablemente nos ayuden a solucionar con facilidad el problema.

Aunque no hemos descubierto los tipos de distribución que siguen nuestras variables, por si quisiéramos transformarlas a una normal, podemos sin ninguna duda aplicar una estandarización de los datos (puesto que sabemos que no afecta negativamente al problema de regresión) siempre y cuando lo tengamos en cuenta a la hora de analizar los resultados.

Se nos pide elegir 5 regresores para la regresión y contamos exactamente con ese número, por lo que no podemos descartar ninguna variable. Aún así, hemos visto que tenemos algunas variables más interesentas que otras. Varibles correladas con la salida nos aumentan las posibilidades de obtener un buen regresor, pero debemos evitar usar variables correladas entre sí para evitar la multicolinealidad. Sería conveniente evitarla para aumentar la interpretabilidad del modelo, pero la potencia en sí de este no cambia. (https://statisticsbyjim.com/regression/multicollinearity-in-regression-analysis

/#:~:text=Multicollinearity%20occurs%20when%20independent%20variables,model%20and%20interpret%20the%20results.)
(referenciar esta frase en el apartado de regresión)