Devoir A2022

Marsel Bakashov

08 Dec 2022

library(dplyr)  
library(knitr)  
library(ggplot2)  
library(gtsummary)  
  
opts\_chunk$set(  
 fig.width = 6,  
 fig.height = 4,  
 fig.align = 'center'  
)  
  
reset\_gtsummary\_theme()

# Fonction pour calculer le coefficient de variation.  
cv <- function(x) {  
 return(sd(x) / mean(x))  
}  
  
# Fonction pour calculer l'écart interquartile.  
iqr <- function(x) {  
 return(quantile(x, .75) - quantile(x, .25))  
}  
  
# Fonction pour calculer l'erreur type  
se <- function(x) {  
 return(sd(x)/sqrt(length(x)))  
}

source('charger.R')  
mondata<-charger(2147174)  
mondata

## mpg displacement weight origin  
## 22 15.0 390.0 3850 1  
## 81 23.0 120.0 2506 0  
## 77 32.8 78.0 1985 0  
## 96 34.4 98.0 2045 1  
## 56 10.0 307.0 4376 1  
## 208 11.0 400.0 4997 1  
## 138 20.0 114.0 2582 0  
## 40 28.0 97.0 2288 0  
## 154 16.0 250.0 3278 1  
## 60 13.0 302.0 3870 1  
## 227 13.0 350.0 4055 1  
## 5 16.0 250.0 3897 1  
## 165 23.9 119.0 2405 0  
## 101 23.0 97.0 2254 0  
## 44 25.0 97.5 2126 1  
## 218 16.0 318.0 4498 1  
## 131 18.0 250.0 3021 1  
## 93 31.0 76.0 1649 0  
## 250 17.0 260.0 4060 1  
## 102 17.5 305.0 3880 1  
## 122 23.0 122.0 2220 1  
## 76 23.0 115.0 2694 0  
## 181 27.0 112.0 2640 1  
## 28 21.0 200.0 2587 1  
## 129 30.0 111.0 2155 1  
## 219 26.0 97.0 1835 0  
## 162 13.0 400.0 5140 1  
## 201 30.5 98.0 2051 1  
## 184 15.0 318.0 4135 1  
## 172 12.0 383.0 4955 1  
## 222 20.0 156.0 2807 0  
## 238 22.0 108.0 2379 0  
## 215 32.0 91.0 1965 0  
## 151 16.0 400.0 4278 1  
## 99 24.2 146.0 2930 0  
## 226 30.7 145.0 3160 0  
## 71 29.0 68.0 1867 0  
## 197 31.0 119.0 2720 1  
## 240 11.0 429.0 4633 1  
## 179 32.0 85.0 1990 0  
## 205 24.5 98.0 2164 1  
## 180 24.0 121.0 2660 0  
## 190 13.0 351.0 4363 1  
## 8 24.3 151.0 3003 1  
## 147 28.4 151.0 2670 1  
## 171 13.0 318.0 3755 1  
## 183 30.0 98.0 2155 1  
## 21 37.0 85.0 1975 0  
## 52 21.0 199.0 2648 1  
## 91 14.0 340.0 3609 1  
## 74 33.7 107.0 2210 0  
## 58 23.5 173.0 2725 1  
## 123 21.5 80.0 2720 0  
## 216 25.8 156.0 2620 1  
## 115 29.0 90.0 1937 0  
## 42 25.0 140.0 2572 1  
## 189 30.0 79.0 2074 0  
## 112 19.0 121.0 2868 0  
## 46 29.5 98.0 2135 0  
## 217 19.0 70.0 2330 0  
## 18 26.0 97.0 2300 0  
## 89 22.0 225.0 3233 1  
## 62 17.0 250.0 3329 1  
## 43 15.0 383.0 3563 1  
## 107 29.8 134.0 2711 0  
## 11 21.6 121.0 2795 0  
## 175 18.5 360.0 3940 1  
## 207 29.0 135.0 2525 1  
## 98 15.0 400.0 3761 1  
## 186 15.5 351.0 4054 1  
## 195 19.2 305.0 3425 1  
## 223 24.0 140.0 2865 1  
## 6 17.0 163.0 3140 0  
## 103 18.0 199.0 2774 1  
## 82 44.0 97.0 2130 0  
## 166 13.0 302.0 3169 1  
## 229 26.6 350.0 3725 1  
## 168 29.0 97.0 2171 0  
## 26 20.0 262.0 3221 1  
## 234 31.0 79.0 2000 0  
## 157 36.1 98.0 1800 1  
## 211 13.0 440.0 4735 1  
## 23 26.0 91.0 1955 1  
## 132 25.0 90.0 2223 0  
## 57 19.8 200.0 2990 1  
## 248 26.0 79.0 1963 0  
## 110 26.6 151.0 2635 1  
## 113 13.0 350.0 4699 1  
## 199 36.0 120.0 2160 0  
## 33 18.0 70.0 2124 0  
## 247 19.0 232.0 3211 1  
## 203 28.0 151.0 2678 1  
## 19 32.0 83.0 2003 0  
## 7 18.5 250.0 3525 1  
## 87 14.0 318.0 4077 1  
## 92 21.5 231.0 3245 1  
## 119 20.0 232.0 2914 1  
## 79 30.9 105.0 2230 1  
## 27 21.0 231.0 3039 1  
## 61 19.0 120.0 3270 0  
## 124 14.0 318.0 4457 1  
## 246 34.2 105.0 2200 1  
## 209 37.2 86.0 2019 0  
## 140 34.5 105.0 2150 1  
## 210 25.0 104.0 2375 0  
## 63 27.5 134.0 2560 0  
## 118 35.0 72.0 1613 0  
## 144 25.4 183.0 3530 0  
## 24 34.3 97.0 2188 0  
## 148 22.5 232.0 3085 1  
## 164 26.0 98.0 2265 0  
## 200 28.0 120.0 2625 1  
## 167 46.6 86.0 2110 0  
## 97 22.4 231.0 3415 1  
## 64 26.0 97.0 2265 0  
## 161 18.0 225.0 3785 1  
## 220 36.1 91.0 1800 0  
## 241 15.5 318.0 4140 1  
## 85 15.0 318.0 3777 1  
## 134 18.0 318.0 3436 1  
## 114 34.0 108.0 2245 0  
## 182 37.0 119.0 2434 0  
## 88 23.0 140.0 2592 1  
## 236 16.0 318.0 4190 1  
## 225 26.0 122.0 2451 1  
## 232 10.0 360.0 4615 1  
## 116 25.5 122.0 2300 1  
## 221 19.0 250.0 3282 1  
## 185 23.9 260.0 3420 1  
## 34 44.3 90.0 2085 0  
## 121 34.0 112.0 2395 1  
## 125 19.1 225.0 3381 1  
## 14 17.5 250.0 3520 1  
## 108 32.7 168.0 2910 0  
## 243 19.0 156.0 2930 0  
## 30 26.0 121.0 2234 0  
## 94 30.5 97.0 2190 0  
## 49 41.5 98.0 2144 0  
## 239 20.3 131.0 2830 0  
## 245 16.0 302.0 4141 1  
## 75 36.0 107.0 2205 0  
## 242 35.1 81.0 1760 0  
## 150 20.0 130.0 3150 0  
## 173 25.0 140.0 2542 1  
## 230 18.0 232.0 2945 1  
## 86 24.0 120.0 2489 0  
## 212 20.8 200.0 3070 1  
## 51 16.5 168.0 3820 0  
## 137 14.0 302.0 4638 1  
## 249 27.0 97.0 1834 0  
## 117 12.0 455.0 4951 1  
## 146 22.0 122.0 2395 1  
## 106 24.0 107.0 2430 0  
## 13 22.3 140.0 2890 1  
## 111 29.0 90.0 1937 0  
## 95 27.4 121.0 2670 1  
## 244 27.0 97.0 2100 0  
## 1 28.0 97.0 2155 0  
## 47 29.0 97.0 1940 0  
## 20 38.1 89.0 1968 0  
## 4 23.0 198.0 2904 1  
## 145 20.2 200.0 3060 1  
## 213 21.0 140.0 2401 1  
## 53 31.0 91.0 1970 0  
## 188 14.0 304.0 4257 1  
## 41 15.5 400.0 4325 1  
## 70 12.0 350.0 4499 1  
## 12 21.1 134.0 2515 0  
## 158 20.2 302.0 3570 1  
## 68 22.0 146.0 2815 0  
## 25 27.0 101.0 2202 0  
## 83 36.0 135.0 2370 1  
## 156 16.0 400.0 4220 1  
## 159 28.0 98.0 2164 1  
## 155 43.4 90.0 2335 0  
## 149 18.0 258.0 2962 1  
## 177 17.6 302.0 3725 1  
## 231 18.0 232.0 2789 1  
## 135 31.3 120.0 2542 0  
## 73 34.1 91.0 1985 0  
## 105 31.0 71.0 1773 0  
## 50 18.0 307.0 3504 1  
## 59 22.0 198.0 2833 1  
## 72 24.0 90.0 2108 0  
## 109 20.0 225.0 3651 1  
## 54 18.0 121.0 2933 0  
## 237 20.2 200.0 2965 1  
## 133 33.8 97.0 2145 0  
## 15 13.0 307.0 4098 1  
## 194 30.0 88.0 2065 0  
## 206 31.6 120.0 2635 0  
## 192 14.0 302.0 4042 1  
## 139 39.4 85.0 2070 0  
## 2 11.0 318.0 4382 1  
## 67 18.0 232.0 3288 1  
## 36 33.5 98.0 2075 1  
## 224 19.0 250.0 3302 1  
## 100 9.0 304.0 4732 1  
## 204 12.0 350.0 4456 1  
## 65 15.0 250.0 3432 1  
## 37 24.0 134.0 2702 0  
## 3 37.7 89.0 2050 0  
## 191 16.0 250.0 3781 1  
## 160 13.0 360.0 3821 1  
## 31 17.5 318.0 4080 1  
## 35 31.5 89.0 1990 0  
## 104 16.9 350.0 4360 1  
## 187 24.0 113.0 2372 0  
## 16 32.4 107.0 2290 0  
## 130 29.0 98.0 2219 0  
## 69 25.0 110.0 2672 0  
## 235 15.5 350.0 4165 1  
## 9 13.0 350.0 4274 1  
## 176 28.1 141.0 3230 0  
## 141 28.0 107.0 2464 0  
## 233 23.0 120.0 2957 0  
## 90 44.6 91.0 1850 0  
## 126 34.1 86.0 1975 0  
## 143 23.7 70.0 2420 0  
## 48 28.0 90.0 2125 1

#Phase 1 : Analyse statistique descriptive et inference

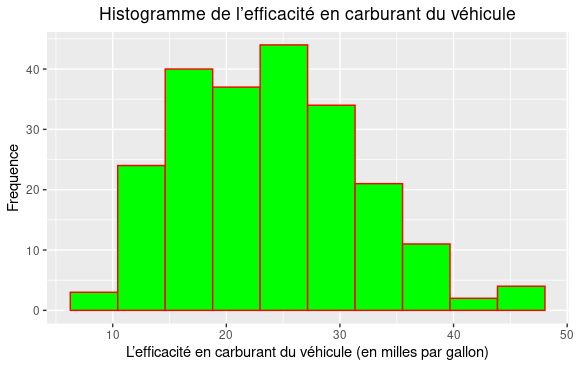
1. (2 points) Examinez les liens entre les variables quantitatives de l’étude. Pour cela, produisez une matrice des corrélations pour l’ensemble des trois variables quantitatives et commentez brièvement

mcor <- cor(mondata[,1:3])  
round(mcor,2)

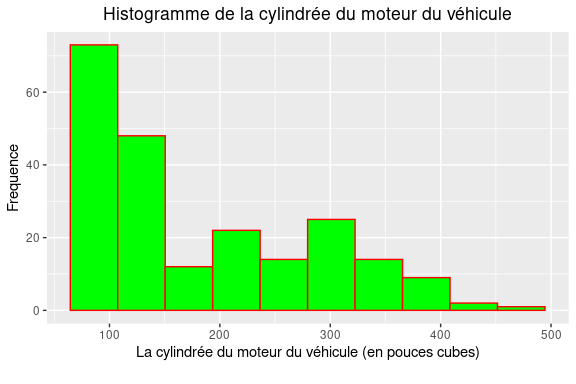
## mpg displacement weight  
## mpg 1.00 -0.79 -0.82  
## displacement -0.79 1.00 0.93  
## weight -0.82 0.93 1.00

1. (8 points) Pour chacune des trois variables Y (l’efficacité en carburant), X1 (la cylindrée) et X2 (le poids), produisez les graphiques et les tableaux demandés ci-dessous et interprétez brièvement les résultats dans chaque cas : • un histogramme et un diagramme de Tukey (ou «Box Plot») ;

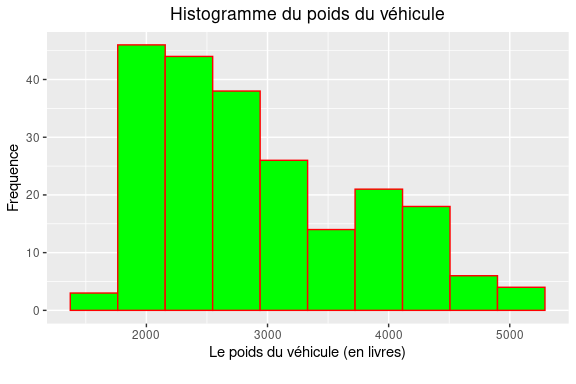
mondata%>%  
 ggplot(aes(x=mpg)) +   
 geom\_histogram(  
 col = "red",  
 fill = "green",  
 bins = 10  
 ) +  
 labs(x='L’efficacité en carburant du véhicule (en milles par gallon)  
', y ='Frequence') +  
 ggtitle('Histogramme de l’efficacité en carburant du véhicule') +  
 theme(plot.title = element\_text(hjust = 0.5))



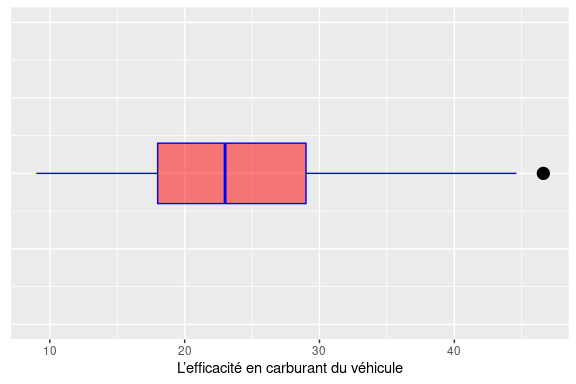
mondata%>%  
 ggplot(aes(x=displacement)) +   
 geom\_histogram(  
 col = "red",  
 fill = "green",  
 bins = 10  
 ) +  
 labs(x='La cylindrée du moteur du véhicule (en pouces cubes)  
', y ='Frequence') +  
 ggtitle('Histogramme de la cylindrée du moteur du véhicule') +  
 theme(plot.title = element\_text(hjust = 0.5))



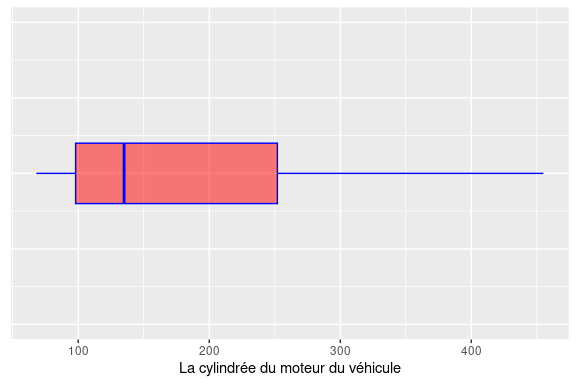
mondata%>%  
 ggplot(aes(x=weight)) +   
 geom\_histogram(  
 col = "red",  
 fill = "green",  
 bins = 10  
 ) +  
 labs(x='Le poids du véhicule (en livres)  
', y ='Frequence') +  
 ggtitle('Histogramme du poids du véhicule') +  
 theme(plot.title = element\_text(hjust = 0.5))



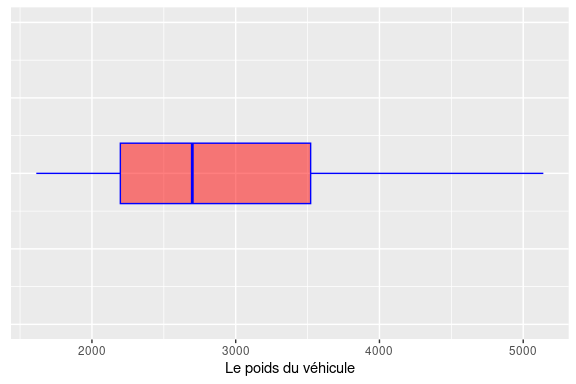
mondata%>%  
 ggplot(aes(x=mpg)) +  
 geom\_boxplot(  
 color = 'blue',  
 fill = 'red',  
 alpha = 0.5,  
 width = 0.2,  
 outlier.color = 'black',  
 outlier.fill = 'black',  
 outlier.alpha = 1,  
 outlier.size = 4  
 ) +   
ylim(-0.5, 0.5) +   
labs(x='L’efficacité en carburant du véhicule') +  
theme(  
 axis.ticks.y = element\_blank(),  
 axis.text.y = element\_blank()  
)



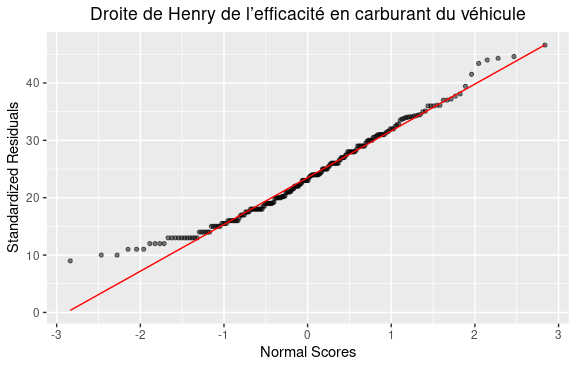
mondata%>%  
 ggplot(aes(x=displacement)) +  
 geom\_boxplot(  
 color = 'blue',  
 fill = 'red',  
 alpha = 0.5,  
 width = 0.2,  
 outlier.color = 'black',  
 outlier.fill = 'black',  
 outlier.alpha = 1,  
 outlier.size = 4  
 ) +   
ylim(-0.5, 0.5) +   
labs(x='La cylindrée du moteur du véhicule') +  
theme(  
 axis.ticks.y = element\_blank(),  
 axis.text.y = element\_blank()  
)



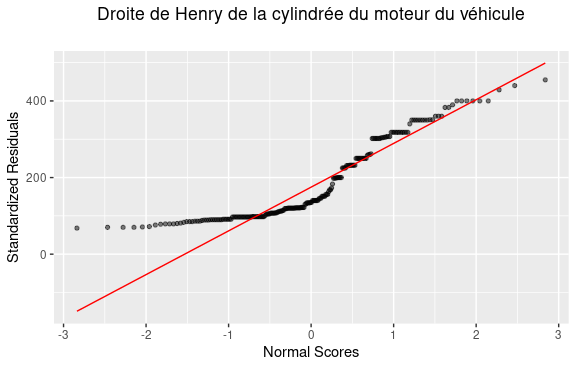
mondata%>%  
 ggplot(aes(x=weight)) +  
 geom\_boxplot(  
 color = 'blue',  
 fill = 'red',  
 alpha = 0.5,  
 width = 0.2,  
 outlier.color = 'black',  
 outlier.fill = 'black',  
 outlier.alpha = 1,  
 outlier.size = 4  
 ) +   
ylim(-0.5, 0.5) +   
labs(x='Le poids du véhicule') +  
theme(  
 axis.ticks.y = element\_blank(),  
 axis.text.y = element\_blank()  
)

 • une droite de Henry (ou «Normal Probability Plot») et un test de normalité (Shapiro-Wilk) ;

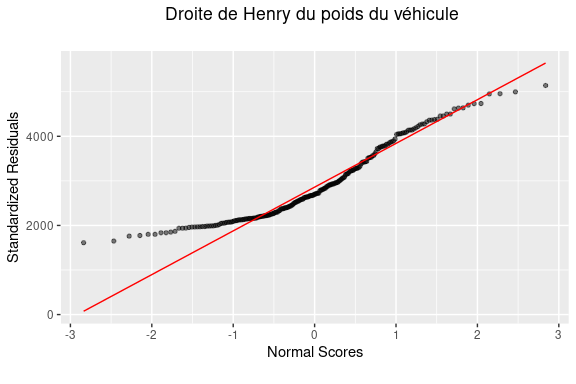
mondata%>%  
 ggplot(aes(sample=mpg)) +  
 stat\_qq(  
 alpha = 0.5,  
 size = 1.2  
 ) +  
 stat\_qq\_line(  
 color = 'red'  
 )+  
 labs(x='Normal Scores  
', y ='Standardized Residuals') +  
 ggtitle('Droite de Henry de l’efficacité en carburant du véhicule') +  
 theme(plot.title = element\_text(hjust = 0.5))



mondata%>%  
 ggplot(aes(sample=displacement)) +  
 stat\_qq(  
 alpha = 0.5,  
 size = 1.2  
 ) +  
 stat\_qq\_line(  
 color = 'red'  
 )+  
 labs(x='Normal Scores  
', y ='Standardized Residuals') +  
 ggtitle('Droite de Henry de la cylindrée du moteur du véhicule  
') +  
 theme(plot.title = element\_text(hjust = 0.5))



mondata%>%  
 ggplot(aes(sample=weight)) +  
 stat\_qq(  
 alpha = 0.5,  
 size = 1.2  
 ) +  
 stat\_qq\_line(  
 color = 'red'  
 )+  
 labs(x='Normal Scores  
', y ='Standardized Residuals') +  
 ggtitle('Droite de Henry du poids du véhicule   
') +  
 theme(plot.title = element\_text(hjust = 0.5))



shapiro.test(mondata$mpg)

##   
## Shapiro-Wilk normality test  
##   
## data: mondata$mpg  
## W = 0.97628, p-value = 0.0009237

shapiro.test(mondata$displacement)

##   
## Shapiro-Wilk normality test  
##   
## data: mondata$displacement  
## W = 0.86054, p-value = 2.808e-13

shapiro.test(mondata$weight)

##   
## Shapiro-Wilk normality test  
##   
## data: mondata$weight  
## W = 0.92796, p-value = 6.651e-09

• un tableau de statistiques descriptives comprenant : moyenne, quartiles, écart type, erreur type, intervalle de confiance pour la moyenne ;

mondata%>%  
 tbl\_summary(  
 include = mpg,  
 label = mpg ~ "mpg",  
 type = mpg ~ "continuous2",  
 statistic = mpg ~ c(  
 "{mean}",  
 "{sd}",  
 "{se}",  
 "{min}",  
 "{max}",  
 "{p25}",  
 "{median}",  
 "{p75}"  
 )  
 )-> mpg.stats.table  
  
mpg.stats.table %>%  
 add\_stat\_label(  
 label = mpg ~ c(  
 "Moyenne",  
 "Écart-type",  
 "Erreur type",  
 "Minimum",  
 "Maximum",  
 "Premier quartile",  
 "Médiane",  
 "Troisième quartile"  
 )  
 ) -> mpg.stats.table  
  
mpg.stats.table %>%  
 modify\_header(  
 label ~ "\*\*Statistiques descriptives\*\*",  
 all\_stat\_cols() ~ ""  
 ) -> mpg.stats.table  
  
mpg.stats.table %>%  
 bold\_labels() %>%   
 italicize\_levels() -> mpg.stats.table  
  
mpg.stats.table %>%  
 add\_ci()

## Table printed with `knitr::kable()`, not {gt}. Learn why at  
## https://www.danieldsjoberg.com/gtsummary/articles/rmarkdown.html  
## To suppress this message, include `message = FALSE` in the code chunk header.

| **Statistiques descriptives** |  | **95% CI** |
| --- | --- | --- |
| **mpg** |  | 23, 25 |
| *Moyenne* | 24 |  |
| *Écart-type* | 8 |  |
| *Erreur type* | 1 |  |
| *Minimum* | 9 |  |
| *Maximum* | 47 |  |
| *Premier quartile* | 18 |  |
| *Médiane* | 23 |  |
| *Troisième quartile* | 29 |  |

mondata%>%  
 tbl\_summary(  
 include = displacement,  
 label = displacement ~ "displacement",  
 type = displacement ~ "continuous2",  
 statistic = displacement ~ c(  
 "{mean}",  
 "{sd}",  
 "{se}",  
 "{min}",  
 "{max}",  
 "{p25}",  
 "{median}",  
 "{p75}"  
 )  
 )-> displacement.stats.table  
  
displacement.stats.table %>%  
 add\_stat\_label(  
 label = displacement ~ c(  
 "Moyenne",  
 "Écart-type",  
 "Erreur type",  
 "Minimum",  
 "Maximum",  
 "Premier quartile",  
 "Médiane",  
 "Troisième quartile"  
 )  
 ) -> displacement.stats.table  
  
displacement.stats.table %>%  
 modify\_header(  
 label ~ "\*\*Statistiques descriptives\*\*",  
 all\_stat\_cols() ~ ""  
 ) -> displacement.stats.table  
  
displacement.stats.table %>%  
 bold\_labels() %>%   
 italicize\_levels() -> displacement.stats.table  
  
displacement.stats.table %>%  
 add\_ci()

## Table printed with `knitr::kable()`, not {gt}. Learn why at  
## https://www.danieldsjoberg.com/gtsummary/articles/rmarkdown.html  
## To suppress this message, include `message = FALSE` in the code chunk header.

| **Statistiques descriptives** |  | **95% CI** |
| --- | --- | --- |
| **displacement** |  | 171, 198 |
| *Moyenne* | 184 |  |
| *Écart-type* | 102 |  |
| *Erreur type* | 7 |  |
| *Minimum* | 68 |  |
| *Maximum* | 455 |  |
| *Premier quartile* | 98 |  |
| *Médiane* | 135 |  |
| *Troisième quartile* | 252 |  |

mondata%>%  
 tbl\_summary(  
 include = weight,  
 label = weight ~ "weight",  
 type = weight ~ "continuous2",  
 statistic = weight ~ c(  
 "{mean}",  
 "{sd}",  
 "{se}",  
 "{min}",  
 "{max}",  
 "{p25}",  
 "{median}",  
 "{p75}"  
 )  
 )-> weight.stats.table  
  
weight.stats.table %>%  
 add\_stat\_label(  
 label = weight ~ c(  
 "Moyenne",  
 "Écart-type",  
 "Erreur type",  
 "Minimum",  
 "Maximum",  
 "Premier quartile",  
 "Médiane",  
 "Troisième quartile"  
 )  
 ) -> weight.stats.table  
  
weight.stats.table %>%  
 modify\_header(  
 label ~ "\*\*Statistiques descriptives\*\*",  
 all\_stat\_cols() ~ ""  
 ) -> weight.stats.table  
  
weight.stats.table %>%  
 bold\_labels() %>%   
 italicize\_levels() -> weight.stats.table  
  
weight.stats.table %>%  
 add\_ci()

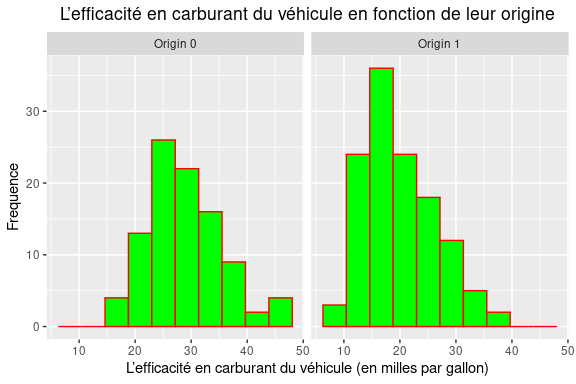
## Table printed with `knitr::kable()`, not {gt}. Learn why at  
## https://www.danieldsjoberg.com/gtsummary/articles/rmarkdown.html  
## To suppress this message, include `message = FALSE` in the code chunk header.

| **Statistiques descriptives** |  | **95% CI** |
| --- | --- | --- |
| **weight** |  | 2,806, 3,032 |
| *Moyenne* | 2,919 |  |
| *Écart-type* | 851 |  |
| *Erreur type* | 57 |  |
| *Minimum* | 1,613 |  |
| *Maximum* | 5,140 |  |
| *Premier quartile* | 2,198 |  |
| *Médiane* | 2,698 |  |
| *Troisième quartile* | 3,521 |  |

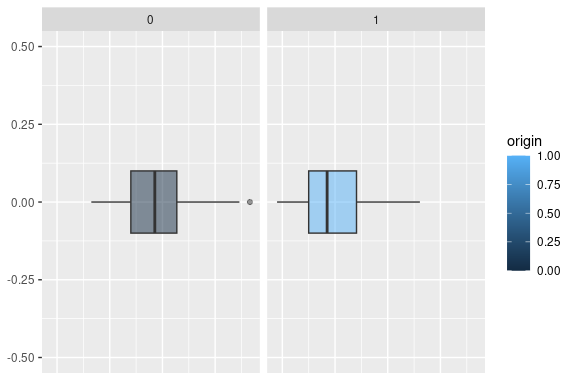
1. (8 points) Afin de vérifier si l’efficacité en carburant d’un véhicule dépend de l’origine de celui-ci, on peut considérer deux groupes de données selon la variable origin et effectuer une comparaison des deux groupes en termes de moyenne,symétrie et variabilité. Pour ce faire, effectuez les analyses suivantes et donnez une brève conclusion :

• deux histogrammes juxtaposés, et deux diagrammes de Tukey (ou «Box Plot») juxtaposés ;

origin.labeller <- function(n) {  
 return(paste("Origin", n))  
}  
  
mondata%>%  
 filter(origin %in% c(0,1))%>%  
 ggplot(aes(x=mpg)) +   
 geom\_histogram(  
 col = "red",  
 fill = "green",  
 bins = 10  
 ) +  
 facet\_wrap(  
 ~factor(origin),  
 labeller = as\_labeller(origin.labeller)  
 )+  
 labs(x='L’efficacité en carburant du véhicule (en milles par gallon)', y ='Frequence') +  
 ggtitle('L’efficacité en carburant du véhicule en fonction de leur origine') +  
 theme(plot.title = element\_text(hjust = 0.5))



mondata%>%  
 ggplot(aes(x=mpg, fill=origin)) +  
 geom\_boxplot(  
 alpha = 0.5,  
 width = 0.2  
 ) +  
 facet\_wrap(  
 ~origin, nrow=1  
 )+  
 ylim(-0.5, 0.5) +   
 labs(x='L’efficacité en carburant du véhicule') +  
 theme(  
 axis.title.x = element\_blank(),  
 axis.text.x = element\_blank(),  
 axis.ticks.x = element\_blank()  
)

 • un tableau des statistiques descriptives par groupe : moyenne, quartiles, variance, écart type, intervalle de confiance pour la moyenne ;

mondata%>%  
 tbl\_summary(  
 include = mpg,  
 by = origin,  
 label = mpg ~ "mpg",  
 type = mpg ~ "continuous2",  
 statistic = mpg ~ c(  
 "{mean}",  
 "{var}",  
 "{sd}",  
 "{min}",  
 "{max}",  
 "{p25}",  
 "{median}",  
 "{p75}"  
 )  
 )-> mpg.stats.table  
  
mpg.stats.table %>%  
 add\_stat\_label(  
 label = mpg ~ c(  
 "Moyenne",  
 "Variance",  
 "Écart-type",  
 "Minimum",  
 "Maximum",  
 "Premier quartile",  
 "Médiane",  
 "Troisième quartile"  
 )  
 ) -> mpg.stats.table  
  
mpg.stats.table %>%  
 modify\_header(label ~ "\*\*Statistiques descriptives\*\*") %>%  
 modify\_header(all\_stat\_cols() ~ "\*{level}\*"  
)-> mpg.stats.table  
  
mpg.stats.table %>%  
 bold\_labels() %>%   
 italicize\_levels() -> mpg.stats.table  
  
mpg.stats.table %>%  
 add\_ci()

## Table printed with `knitr::kable()`, not {gt}. Learn why at  
## https://www.danieldsjoberg.com/gtsummary/articles/rmarkdown.html  
## To suppress this message, include `message = FALSE` in the code chunk header.

| **Statistiques descriptives** | *0* | **95% CI** | *1* | **95% CI** |
| --- | --- | --- | --- | --- |
| **mpg** |  | 27, 30 |  | 19, 21 |
| *Moyenne* | 29 |  | 20 |  |
| *Variance* | 46 |  | 41 |  |
| *Écart-type* | 7 |  | 6 |  |
| *Minimum* | 16 |  | 9 |  |
| *Maximum* | 47 |  | 36 |  |
| *Premier quartile* | 24 |  | 15 |  |
| *Médiane* | 29 |  | 18 |  |
| *Troisième quartile* | 33 |  | 24 |  |

• un test d’hypothèses sur l’égalité des variances des deux groupes ;

var.test(mpg ~ origin, data=mondata, alternative = "two.sided")

##   
## F test to compare two variances  
##   
## data: mpg by origin  
## F = 1.1153, num df = 95, denom df = 123, p-value = 0.5667  
## alternative hypothesis: true ratio of variances is not equal to 1  
## 95 percent confidence interval:  
## 0.7657915 1.6395772  
## sample estimates:  
## ratio of variances   
## 1.11526

• un test d’hypothèses sur l’égalité des moyennes des deux groupes.

t.test(mpg ~ origin, data=mondata, alternative = "two.sided")

##   
## Welch Two Sample t-test  
##   
## data: mpg by origin  
## t = 9.7929, df = 198.71, p-value < 2.2e-16  
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0  
## 95 percent confidence interval:  
## 7.02920 10.57389  
## sample estimates:  
## mean in group 0 mean in group 1   
## 28.73542 19.93387

#Phase 2 : Recherche d’un modèle On s’intéresse dans cette phase à la détermination d’un modèle permettant d’expliquer la performance d’un véhicule en fonction des différentes variables de l’étude. Pour ce faire, on envisage des modèles de régression en considérant l’efficacité en carburant comme variable dépendante, Y . d) (15 points) On considère les huit modèles suivants : Pour chacun des huit modèles ci-dessus :

• (5 points) Effectuez l’ajustement (i.e. obtenir le tableau des coefficients de régression, le tableau d’analyse de la variance).

Y <- mondata$mpg  
X1 <- mondata$displacement  
X2 <- mondata$weight  
X3 <- mondata$origin  
  
linModel1 <- lm(Y ~ X1)  
summary(linModel1)

##   
## Call:  
## lm(formula = Y ~ X1)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -12.6978 -3.0598 -0.6526 2.3310 16.8728   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 34.944473 0.674347 51.82 <2e-16 \*\*\*  
## X1 -0.060666 0.003202 -18.94 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 4.853 on 218 degrees of freedom  
## Multiple R-squared: 0.6221, Adjusted R-squared: 0.6204   
## F-statistic: 358.9 on 1 and 218 DF, p-value: < 2.2e-16

linModel1$coefficients

## (Intercept) X1   
## 34.94447336 -0.06066641

anova(linModel1)

## Analysis of Variance Table  
##   
## Response: Y  
## Df Sum Sq Mean Sq F value Pr(>F)   
## X1 1 8453.2 8453.2 358.86 < 2.2e-16 \*\*\*  
## Residuals 218 5135.1 23.6   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

linModel2 <- lm(Y ~ X1^2)  
summary(linModel2)

##   
## Call:  
## lm(formula = Y ~ X1^2)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -12.6978 -3.0598 -0.6526 2.3310 16.8728   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 34.944473 0.674347 51.82 <2e-16 \*\*\*  
## X1 -0.060666 0.003202 -18.94 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 4.853 on 218 degrees of freedom  
## Multiple R-squared: 0.6221, Adjusted R-squared: 0.6204   
## F-statistic: 358.9 on 1 and 218 DF, p-value: < 2.2e-16

linModel2$coefficients

## (Intercept) X1   
## 34.94447336 -0.06066641

anova(linModel2)

## Analysis of Variance Table  
##   
## Response: Y  
## Df Sum Sq Mean Sq F value Pr(>F)   
## X1 1 8453.2 8453.2 358.86 < 2.2e-16 \*\*\*  
## Residuals 218 5135.1 23.6   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

linModel3 <- lm(log(Y) ~log(X1) )  
summary(linModel3)

##   
## Call:  
## lm(formula = log(Y) ~ log(X1))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.66556 -0.12026 0.00221 0.13383 0.59515   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 5.85295 0.11992 48.81 <2e-16 \*\*\*  
## log(X1) -0.54067 0.02352 -22.98 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.1861 on 218 degrees of freedom  
## Multiple R-squared: 0.7079, Adjusted R-squared: 0.7065   
## F-statistic: 528.2 on 1 and 218 DF, p-value: < 2.2e-16

linModel3$coefficients

## (Intercept) log(X1)   
## 5.8529463 -0.5406658

anova(linModel3)

## Analysis of Variance Table  
##   
## Response: log(Y)  
## Df Sum Sq Mean Sq F value Pr(>F)   
## log(X1) 1 18.2947 18.2947 528.23 < 2.2e-16 \*\*\*  
## Residuals 218 7.5502 0.0346   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

linModel4 <- lm(log(Y) ~X1)  
summary(linModel4)

##   
## Call:  
## lm(formula = log(Y) ~ X1)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.57650 -0.11262 -0.01162 0.12735 0.63696   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 3.6313215 0.0258848 140.29 <2e-16 \*\*\*  
## X1 -0.0028211 0.0001229 -22.95 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.1863 on 218 degrees of freedom  
## Multiple R-squared: 0.7073, Adjusted R-squared: 0.7059   
## F-statistic: 526.7 on 1 and 218 DF, p-value: < 2.2e-16

linModel4$coefficients

## (Intercept) X1   
## 3.631321505 -0.002821058

anova(linModel4)

## Analysis of Variance Table  
##   
## Response: log(Y)  
## Df Sum Sq Mean Sq F value Pr(>F)   
## X1 1 18.2788 18.2788 526.66 < 2.2e-16 \*\*\*  
## Residuals 218 7.5661 0.0347   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

linModel5 <- lm(Y ~ X2)  
summary(linModel5)

##   
## Call:  
## lm(formula = Y ~ X2)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -11.8378 -2.9317 -0.4811 2.3942 16.6554   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 46.0413411 1.0796593 42.64 <2e-16 \*\*\*  
## X2 -0.0076288 0.0003552 -21.48 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 4.472 on 218 degrees of freedom  
## Multiple R-squared: 0.6791, Adjusted R-squared: 0.6776   
## F-statistic: 461.3 on 1 and 218 DF, p-value: < 2.2e-16

linModel5$coefficients

## (Intercept) X2   
## 46.041341096 -0.007628798

anova(linModel5)

## Analysis of Variance Table  
##   
## Response: Y  
## Df Sum Sq Mean Sq F value Pr(>F)   
## X2 1 9227.7 9227.7 461.33 < 2.2e-16 \*\*\*  
## Residuals 218 4360.5 20.0   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

linModel6 <- lm(Y ~ X2^2)  
summary(linModel6)

##   
## Call:  
## lm(formula = Y ~ X2^2)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -11.8378 -2.9317 -0.4811 2.3942 16.6554   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 46.0413411 1.0796593 42.64 <2e-16 \*\*\*  
## X2 -0.0076288 0.0003552 -21.48 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 4.472 on 218 degrees of freedom  
## Multiple R-squared: 0.6791, Adjusted R-squared: 0.6776   
## F-statistic: 461.3 on 1 and 218 DF, p-value: < 2.2e-16

linModel6$coefficients

## (Intercept) X2   
## 46.041341096 -0.007628798

anova(linModel6)

## Analysis of Variance Table  
##   
## Response: Y  
## Df Sum Sq Mean Sq F value Pr(>F)   
## X2 1 9227.7 9227.7 461.33 < 2.2e-16 \*\*\*  
## Residuals 218 4360.5 20.0   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

linModel7 <- lm(log(mondata$mpg) ~ log(mondata$weight) )  
summary(linModel7)

##   
## Call:  
## lm(formula = log(mondata$mpg) ~ log(mondata$weight))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.51720 -0.10534 -0.00395 0.09967 0.47163   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 11.56674 0.32020 36.12 <2e-16 \*\*\*  
## log(mondata$weight) -1.06502 0.04031 -26.42 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.168 on 218 degrees of freedom  
## Multiple R-squared: 0.762, Adjusted R-squared: 0.7609   
## F-statistic: 698.1 on 1 and 218 DF, p-value: < 2.2e-16

linModel7$coefficients

## (Intercept) log(mondata$weight)   
## 11.566735 -1.065018

anova(linModel7)

## Analysis of Variance Table  
##   
## Response: log(mondata$mpg)  
## Df Sum Sq Mean Sq F value Pr(>F)   
## log(mondata$weight) 1 19.6947 19.6947 698.1 < 2.2e-16 \*\*\*  
## Residuals 218 6.1502 0.0282   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

linModel8 <- lm(log(Y) ~ X2)  
summary(linModel8)

##   
## Call:  
## lm(formula = log(Y) ~ X2)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.50237 -0.10633 -0.00992 0.09648 0.45388   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 4.143e+00 4.021e-02 103.05 <2e-16 \*\*\*  
## X2 -3.533e-04 1.323e-05 -26.71 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.1666 on 218 degrees of freedom  
## Multiple R-squared: 0.766, Adjusted R-squared: 0.7649   
## F-statistic: 713.6 on 1 and 218 DF, p-value: < 2.2e-16

linModel8$coefficients

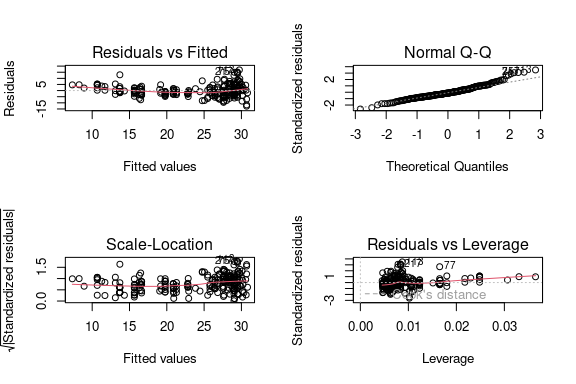
## (Intercept) X2   
## 4.1432714372 -0.0003533544

anova(linModel8)

## Analysis of Variance Table  
##   
## Response: log(Y)  
## Df Sum Sq Mean Sq F value Pr(>F)   
## X2 1 19.7972 19.7972 713.63 < 2.2e-16 \*\*\*  
## Residuals 218 6.0477 0.0277   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

• (5 points) Tester la signification du modèle et effectuez une analyse des résidus (normalité, homoscédasticité, points atypiques, etc.)

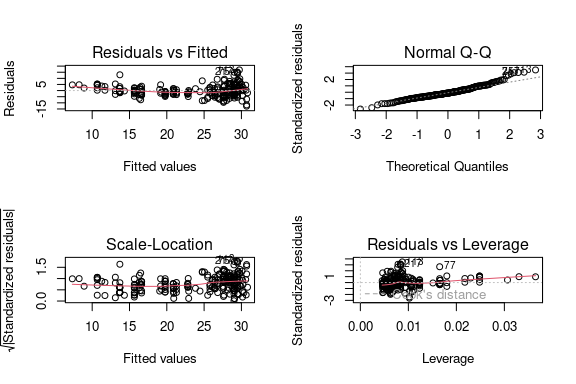
par(mfrow = c(2,2))  
plot(linModel1)



shapiro.test(rstudent(linModel1))

##   
## Shapiro-Wilk normality test  
##   
## data: rstudent(linModel1)  
## W = 0.96332, p-value = 1.878e-05

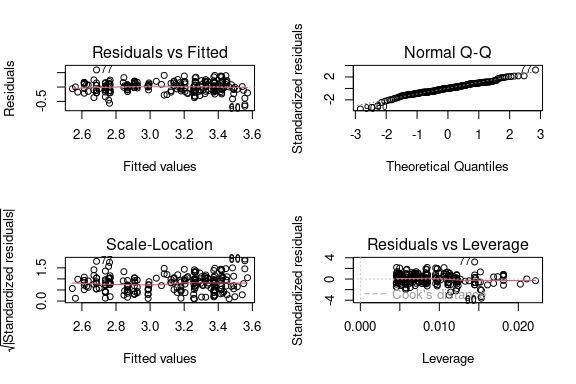
par(mfrow = c(2,2))  
plot(linModel2)



shapiro.test(rstudent(linModel2))

##   
## Shapiro-Wilk normality test  
##   
## data: rstudent(linModel2)  
## W = 0.96332, p-value = 1.878e-05

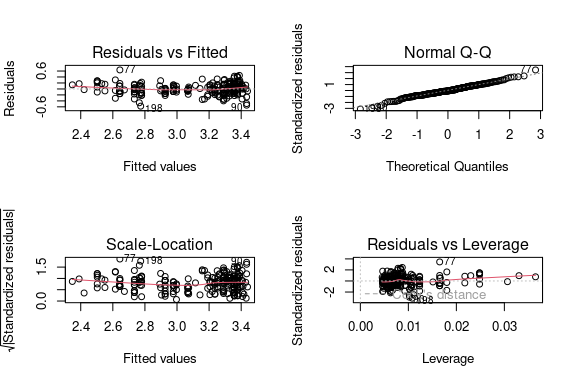
par(mfrow = c(2,2))  
plot(linModel3)



shapiro.test(rstudent(linModel3))

##   
## Shapiro-Wilk normality test  
##   
## data: rstudent(linModel3)  
## W = 0.98211, p-value = 0.006909

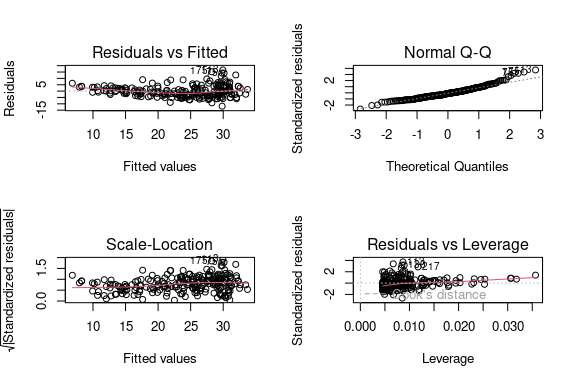
par(mfrow = c(2,2))  
plot(linModel4)



shapiro.test(rstudent(linModel4))

##   
## Shapiro-Wilk normality test  
##   
## data: rstudent(linModel4)  
## W = 0.99109, p-value = 0.1977

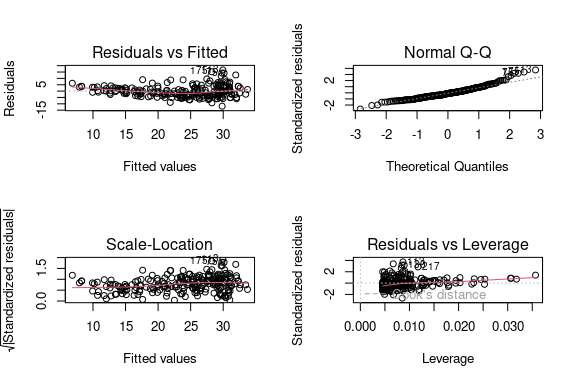
par(mfrow = c(2,2))  
plot(linModel5)



shapiro.test(rstudent(linModel5))

##   
## Shapiro-Wilk normality test  
##   
## data: rstudent(linModel5)  
## W = 0.9526, p-value = 1.213e-06

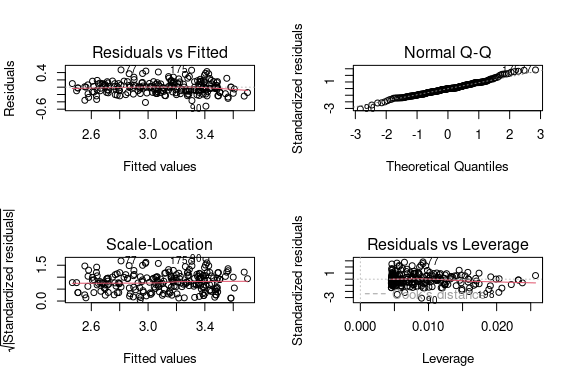
par(mfrow = c(2,2))  
plot(linModel6)



shapiro.test(rstudent(linModel6))

##   
## Shapiro-Wilk normality test  
##   
## data: rstudent(linModel6)  
## W = 0.9526, p-value = 1.213e-06

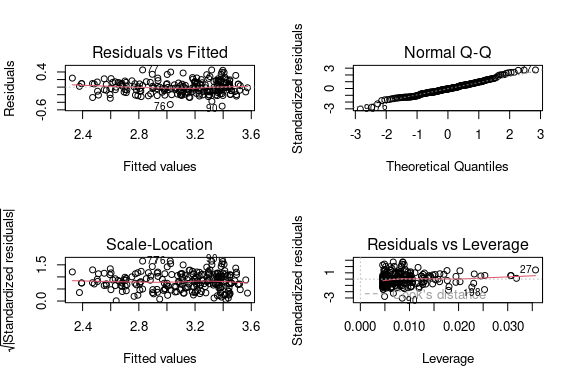
par(mfrow = c(2,2))  
plot(linModel7)



shapiro.test(residuals(linModel7))

##   
## Shapiro-Wilk normality test  
##   
## data: residuals(linModel7)  
## W = 0.98727, p-value = 0.04696

par(mfrow = c(2,2))  
plot(linModel8)



shapiro.test(residuals(linModel8))

##   
## Shapiro-Wilk normality test  
##   
## data: residuals(linModel8)  
## W = 0.9877, p-value = 0.05519

• (3 points) Donner un intervalle de confiance pour chacun des paramètres β0 et β1 des modèles 1 et 5.

confint(linModel1)

## 2.5 % 97.5 %  
## (Intercept) 33.61539919 36.27354753  
## X1 -0.06697816 -0.05435467

confint(linModel5)

## 2.5 % 97.5 %  
## (Intercept) 43.913434591 48.169247600  
## X2 -0.008328828 -0.006928768

• (2 points) En conclusion : effectuez une comparaison

1. (5 points) Sur la base du meilleur modèle que vous avez obtenu en d), calculez un intervalle de prévision pour l’efficacité en carburant d’un véhicule ayant les caractéristiques suivantes : X1 = 190; X2 = 2500. Commentez brièvement votre résultat. Remarque. Notez que le modèle que vous avez obtenu en d) n’utilise pas nécessairement toutes les valeurs ci-dessus

predict(linModel8, newdata = data.frame(X1 = 190, X2 = 2500), interval = "predict", level = 0.95)

## fit lwr upr  
## 1 3.259885 2.930689 3.589082