Previsão da percentagem da despesa total direcionada para alimentação num agregado familiar de Espanha (por mês)

GRUPO 4:

- Gonçalo Mealha 123391
- Jingyu Huang 123432
- Ricardo Valério 112255

Packages e Importação do Dataset

In [ ]:
```r
# packages necessários
set.seed(123)
 #install.packages("corrplot")
library(corrplot) # Matrizes de correlação
 #install.packages("car")
library(car)
 #install.packages("dplyr")
library(dplyr)
 #install.packages("lmtest")
library(lmtest)  # Para testes de hipótese sobre os pressupostos dos resíduos
 #install.packages("tseries")
library(tseries)  # Teste de Jarque-Bera
 #install.packages("caTools") # Separar dataset em teste e treino
library(caTools)
 #install.packages("MASS")
library(MASS)

# Bibliotecas necessárias para a construção dos histogramas e barplots
 #install.packages("ggplot2")
library(ggplot2)
 #install.packages("gridExtra")
library(gridExtra)
 #install.packages("grid")
library(grid)
```

In [2]:
```r
# Importar a base de dados do ficheiro CSV
budget_food_data <- read.csv(
                        file = "BudgetFood_Grupo4.csv",
                        sep = ";",
                        stringsAsFactors = TRUE
                    )[ , -1] # ignorar a 1ª coluna (índice de observações)

# Estrutura do dataframe
str(budget_food_data)
```

```
'data.frame':   2501 obs. of  6 variables:
 $ wfood : num  0.298 0.243 0.653 0.575 0.267 ...
 $ totexp: int  928015 1402566 140676 950352 1226469 643915 498404 511335 982737 437
213 ...
 $ age   : int  30 56 44 43 48 56 45 63 44 68 ...
 $ size  : int  6 4 1 6 2 3 5 4 4 3 ...
 $ town  : int  2 2 2 2 2 2 2 2 2 2 ...
 $ sex   : Factor w/ 2 levels "man","woman": 1 1 1 1 1 1 1 1 1 1 ...
```

Descrição das Variáveis

- wfood: percentagem da despesa total que uma família gasta para alimentação por mês
- totexp: despesas totais do agregado familiar (moeda em peseta Espanhola, transformar em euro, sabendo que 1000 pesetas = 6€)
- age: idade da pessoa de referência do agregado familiar
- size: número de elementos do agregado familiar
- town: tamanho da cidade onde a família tem o domicilio (5 categórias: 1 – cidade pequena,...,5 - cidade grande)
- sex: sexo da pessoa de referência do agregado familiar (man, woman)

Tratamento de dados - Limpezas e Transformações

Nesta etapa, procedemos a limpeza e a transformação dos dados

In [3]:
```
# Verificar a existência dos valores nulos
colSums(is.na(budget_food_data))
```

wfood: 0 **totexp:** 0 **age:** 0 **size:** 0 **town:** 0 **sex:** 0

In [4]:
```
# Verificar a existência dos dados duplicados
duplicate <- sum(duplicated(budget_food_data))

# Mostrar os duplicados
budget_food_data[duplicated(budget_food_data) | duplicated(budget_food_data,
fromLast = TRUE), ]

# Para remover os duplicados
budget_food_data <- budget_food_data %>% distinct()
```

A data.frame: 12 × 6

	wfood	totexp	age	size	town	sex
	<dbl>	<int>	<int>	<int>	<int>	<fct>
1609	0.09017669	4384814	52	4	4	man
1610	0.14984846	3278272	45	6	4	man
1611	0.19597145	904295	41	2	4	woman
1612	0.12740736	2051310	50	4	4	man
1613	0.06305279	2769362	33	4	4	man
1614	0.21822980	1132311	45	5	4	man
1632	0.09017669	4384814	52	4	4	man
1633	0.14984846	3278272	45	6	4	man
1634	0.19597145	904295	41	2	4	woman
1635	0.12740736	2051310	50	4	4	man
1636	0.06305279	2769362	33	4	4	man
1637	0.21822980	1132311	45	5	4	man

In [5]:
```
dim(budget_food_data)  #Foram eliminadas 6 observações
```

2495 · 6

In [6]:
```
# Transformar a coluna "sex" para 0 e 1 (woman = 0, man = 1)
budget_food_data$sex <- ifelse(budget_food_data$sex == "woman", 0, 1)

# Visualizar as primeiras linhas do dataframe para verificar a transformação
head(budget_food_data)
```

A data.frame: 6 × 6

	wfood	totexp	age	size	town	sex
	<dbl>	<int>	<int>	<int>	<int>	<dbl>
1	0.2978185	928015	30	6	2	1
2	0.2433597	1402566	56	4	2	1
3	0.6531605	140676	44	1	2	1
4	0.5747975	950352	43	6	2	1
5	0.2668115	1226469	48	2	2	1
6	0.2880567	643915	56	3	2	1

In [7]:
```
# Transformar a coluna "totexp" (em Peseta Espanhola) para Euros(€)
# sabendo que 1000 pesetas = 6 euros
```

```
budget_food_data$totexp <- budget_food_data$totexp * 6 / 1000

# Visualizar as primeiras linhas do dataframe para verificar a transformação
head(budget_food_data)
```

A data.frame: 6 × 6

	wfood	totexp	age	size	town	sex
	<dbl>	<dbl>	<int>	<int>	<int>	<dbl>
1	0.2978185	5568.090	30	6	2	1
2	0.2433597	8415.396	56	4	2	1
3	0.6531605	844.056	44	1	2	1
4	0.5747975	5702.112	43	6	2	1
5	0.2668115	7358.814	48	2	2	1
6	0.2880567	3863.490	56	3	2	1

In [8]:
```
# Casos em que não existe nenhum gasto na alimentação
wfood0 <- which(budget_food_data$wfood == 0)
budget_food_data[wfood0,]

# Para remover
budget_food_data <- budget_food_data[-wfood0, ]
```

A data.frame: 5 × 6

	wfood	totexp	age	size	town	sex
	<dbl>	<dbl>	<int>	<int>	<int>	<dbl>
947	0	454.560	72	1	2	0
998	0	798.996	78	1	2	1
1455	0	3256.440	26	3	2	1
1907	0	1806.144	55	2	4	0
2012	0	1355.604	49	1	4	1

Análises Descritivas

Nesta fase, fizemos uma breve análise descrtiva das variáveis através de gráficos e histogramas

In [9]:
```
summary(budget_food_data)
```

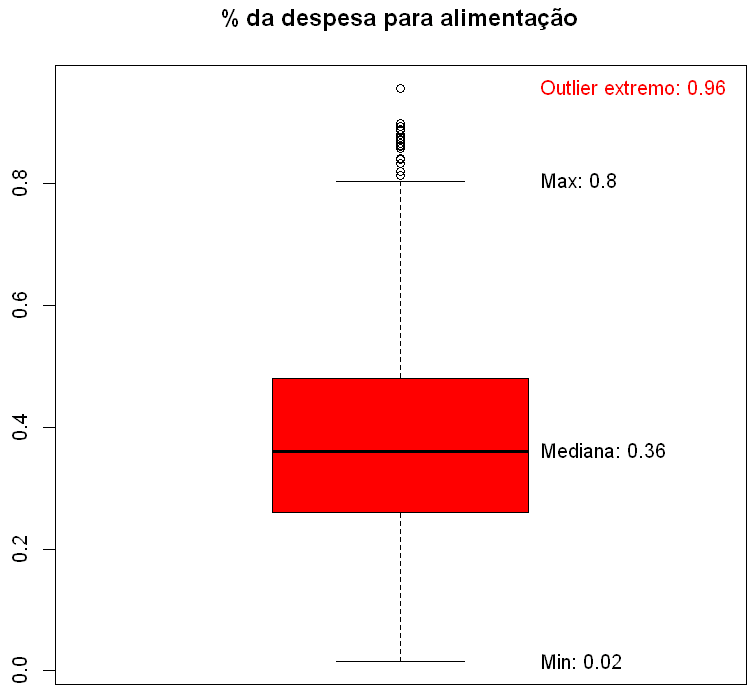
```
     wfood             totexp              age              size       
 Min.   :0.01504   Min.   :  121.7   Min.   :18.00   Min.   : 1.000  
 1st Qu.:0.26063   1st Qu.: 2638.9   1st Qu.:39.00   1st Qu.: 2.000  
 Median :0.36083   Median : 4372.4   Median :51.00   Median : 4.000  
 Mean   :0.37822   Mean   : 5167.8   Mean   :51.28   Mean   : 3.706  
 3rd Qu.:0.47952   3rd Qu.: 6556.4   3rd Qu.:62.00   3rd Qu.: 5.000  
 Max.   :0.95696   Max.   :68385.3   Max.   :97.00   Max.   :17.000  
      town            sex        
 Min.   :1.000   Min.   :0.0000  
 1st Qu.:2.000   1st Qu.:1.0000  
 Median :3.000   Median :1.0000  
 Mean   :2.755   Mean   :0.8695  
 3rd Qu.:4.000   3rd Qu.:1.0000  
 Max.   :4.000   Max.   :1.0000  
```

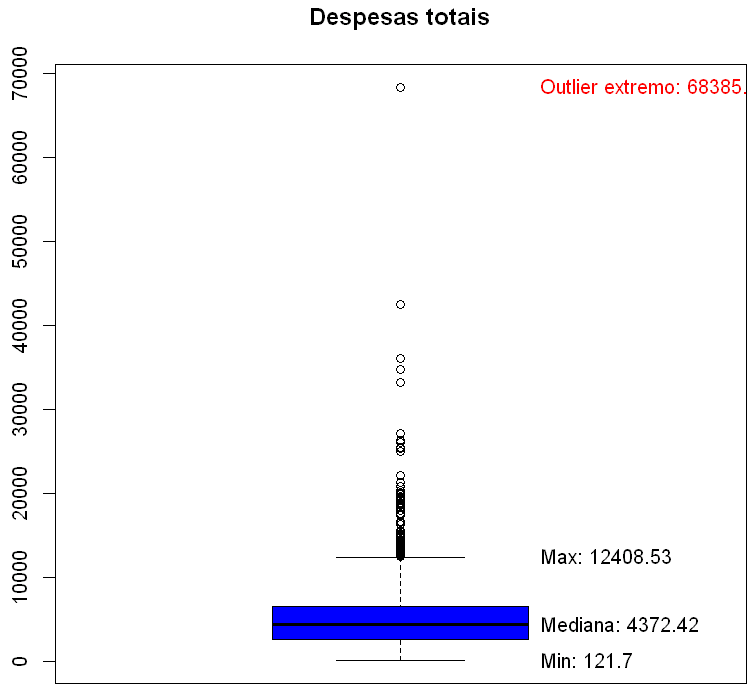
Deteção dos outliers das variáveis numéricas através dos boxplots

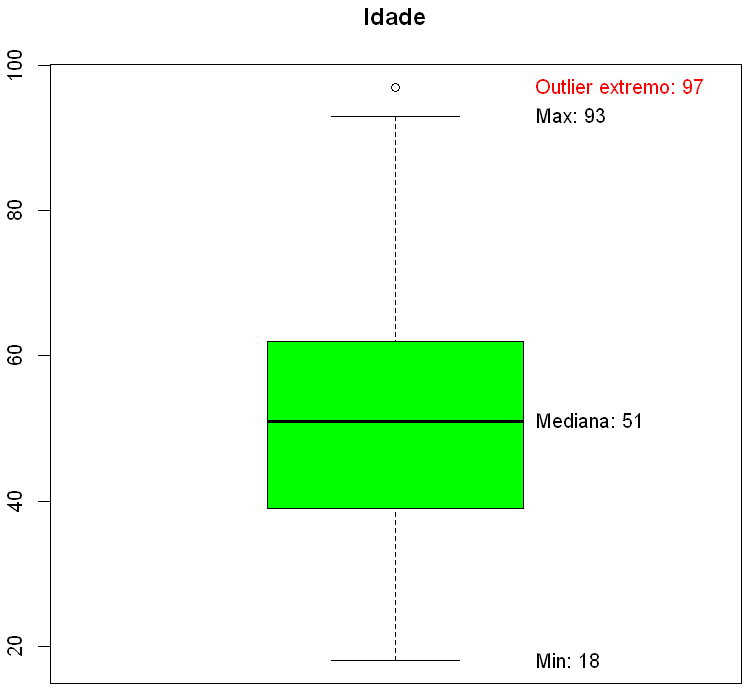
In [10]:

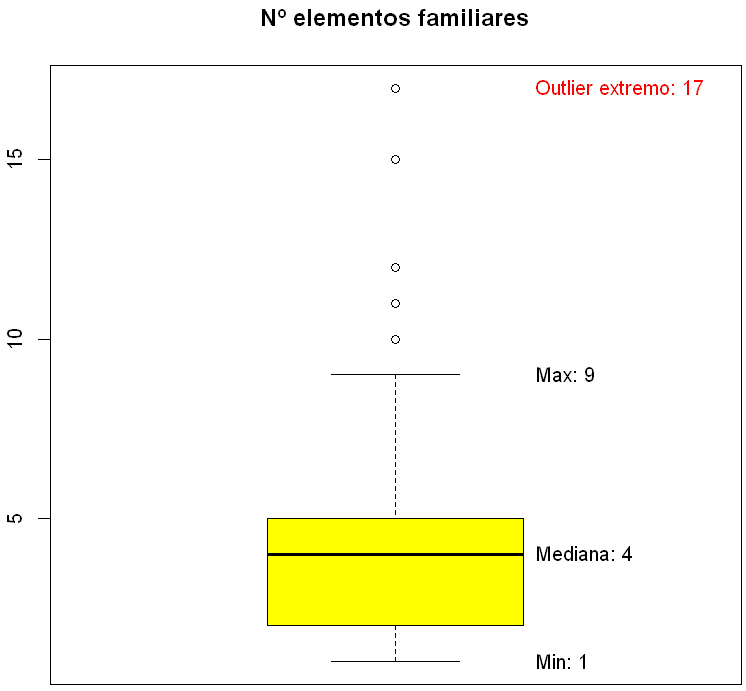
```r
# Outliers das variáveis através do boxplot
variaveis <- list(wfood = budget_food_data$wfood, totexp =
budget_food_data$totexp, age = budget_food_data$age, size = budget_food_data$size,
town = budget_food_data$town)
titulos <- c("% da despesa para alimentação", "Despesas totais", "Idade", "Nº
elementos familiares", "Tamanho da cidade")

# Definir cores para os boxplots
cores <- c("red", "blue", "green", "yellow", "purple")

# Criar os boxplots num ciclo
for (i in 1:length(variaveis)) {
  # Cria o boxplot com cor
  box <- boxplot(variaveis[[i]], main = titulos[i], col = cores[i])

  # Valores estatísticos do boxplot
  min_val <- box$stats[1]  # Mínimo
  q1_val <- box$stats[2]    # Q1
  median_val <- box$stats[3] # Q2
  q3_val <- box$stats[4]    # Q3
  max_val <- box$stats[5]  # Máximo
  outliers <- box$out  # Outliers

  # Identificar o outlier mais extremo (se houver)
  if (length(outliers) > 0) {
    extreme_outlier <- outliers[which.max(abs(outliers - median_val))]  # O mais
distante da mediana
    text(1.2, extreme_outlier, labels = paste("Outlier extremo:",
round(extreme_outlier, 2)), col = "red", pos = 4)
  }

  # Adicionar os valores no gráfico
  text(1.2, min_val, labels = paste("Min:", round(min_val, 2)), pos = 4, col =
"black")
  text(1.2, max_val, labels = paste("Max:", round(max_val, 2)), pos = 4, col =
"black")
  text(1.2, median_val, labels = paste("Mediana:", round(median_val, 2)), pos = 4,
```

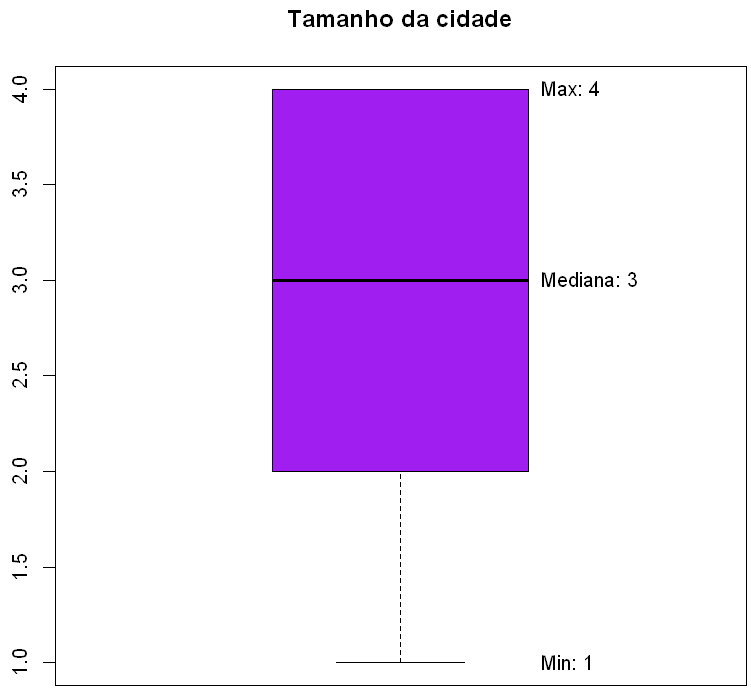
```
col = "black")
}
```


% da despesa para alimentação
Outlier extremo: 0.96
Max: 0.8
Mediana: 0.36
Min: 0.02
0.0
0.2
0.4
0.6
0.8


Despesas totais
Outlier extremo: 68385.
Max: 12408.53
Mediana: 4372.42
Min: 121.7
0
10000
20000
30000
40000
50000
60000
70000


Idade
100
80
60
40
20
Outlier extremo: 97
Max: 93
Mediana: 51
Min: 18


Nº elementos familiares
15
10
5
Outlier extremo: 17
Max: 9
Mediana: 4
Min: 1



Observações:

- A variável "town" não possui outliers;
- As restantes variáveis possuem apenas outliers superiores;
- A variável "totexp" apresenta um número elevado de outliers, existem os quais bastante distindos do máximo.

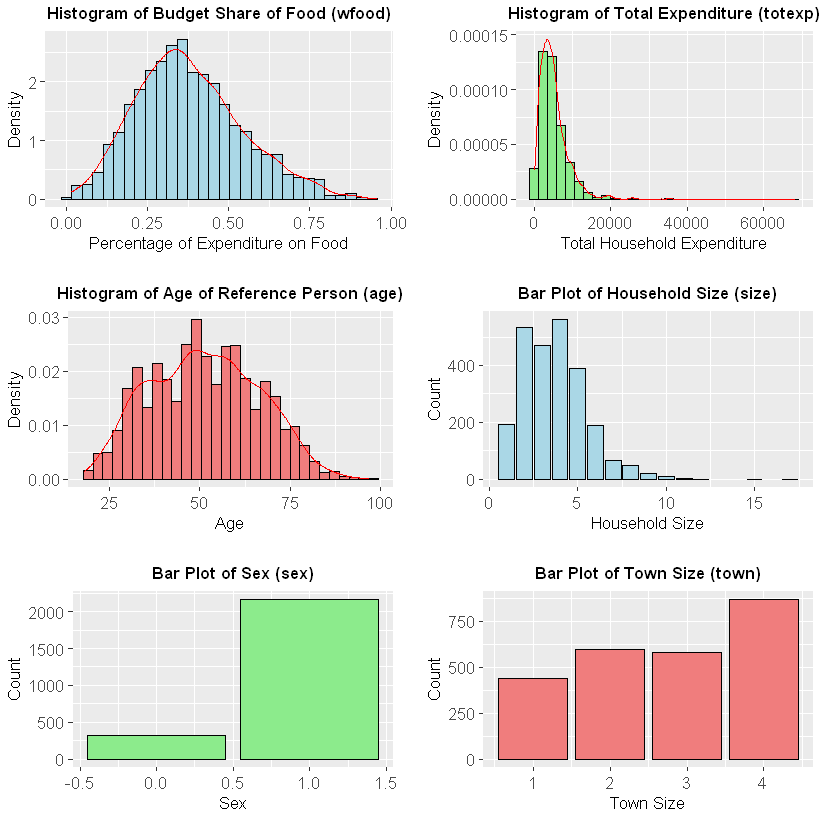
Histogramas da distribuição das variáveis

Esta secção tem como objetivo analisar e compreender a distribuição dos nossos dados

In [11]:
```
# Definir um tema para melhorar a legibilidade
custom_theme <- theme(
    plot.title = element_text(size = 10, face = "bold", hjust = 0.5),
    axis.title = element_text(size = 10),
    axis.text = element_text(size = 10),
    plot.margin = margin(10, 10, 10, 10)
)

# Histograma da variável wfood
hist_wfood <- ggplot(budget_food_data, aes(x = wfood)) +
    geom_histogram(aes(y = ..density..), bins = 30, fill = "lightblue", color =
"black") +
    geom_density(color = "red") + # Adicionar uma curva de densidade em cima do
histograma
    ggtitle("Histogram of Budget Share of Food (wfood)") +
    xlab("Percentage of Expenditure on Food") +
```

```
    ylab("Density") +
    custom_theme

# Histograma da variável totexp
hist_totexp <- ggplot(budget_food_data, aes(x = totexp)) +
    geom_histogram(aes(y = ..density..), bins = 30, fill = "lightgreen", color =
"black") +
    geom_density(color = "red") + # Adicionar uma curva de densidade em cima do
histograma
    ggtitle("Histogram of Total Expenditure (totexp)") +
    xlab("Total Household Expenditure") +
    ylab("Density") +
    custom_theme

# Histograma da  variável age
hist_age <- ggplot(budget_food_data, aes(x = age)) +
    geom_histogram(aes(y = ..density..), bins = 30, fill = "lightcoral", color =
"black") +
    geom_density(color = "red") + # Adicionar uma curva de densidade em cima do
histograma
    ggtitle("Histogram of Age of Reference Person (age)") +
    xlab("Age") +
    ylab("Density") +
    custom_theme

# Barplot da variável size
bar_size <- ggplot(budget_food_data, aes(x = size)) +
    geom_bar(fill = "lightblue", color = "black") +
    ggtitle("Bar Plot of Household Size (size)") +
    xlab("Household Size") +
    ylab("Count") +
    custom_theme

# Barplot da variável sex
bar_age <- ggplot(budget_food_data, aes(x = sex)) +
    geom_bar(fill = "lightgreen", color = "black") +
    ggtitle("Bar Plot of Sex (sex)") +
    xlab("Sex") +
    ylab("Count") +
    custom_theme

# Barplot da variável town
bar_town <- ggplot(budget_food_data, aes(x = town)) +
    geom_bar(fill = "lightcoral", color = "black") +
    ggtitle("Bar Plot of Town Size (town)") +
    xlab("Town Size") +
    ylab("Count") +
    custom_theme

# Organizar gráficos numa matriz
grid.arrange(hist_wfood, hist_totexp,
    hist_age, bar_size,
    bar_age, bar_town,
    ncol = 2, nrow = 3
)
```

```
Warning message:
"The dot-dot notation (`..density..`) was deprecated in ggplot2 3.4.0.
ℹ Please use `after_stat(density)` instead."
```



In [12]:
```
# Tabela de frequência absoluta da variável size
table(budget_food_data$size)
```

```
  1   2   3   4   5   6   7   8   9  10  11  12  15  17
193 534 469 562 391 188  66  49  21  11   3   1   1   1
```

Identificação dos outliers (superiores) das variáveis *wfood*; *totexp*; *age* e *size*

A expressão do cálculo dos outliers superiores é dada por: Outliers = {x ∈ variável | x > Q3 + 1.5 × (Q3−Q1)}

In [13]:
```
# Para podermos referir variáveis diretamente
attach(budget_food_data)
```

In [14]:
```
# Outliers da variável "wfood"
out_wfood <- which(wfood>quantile(wfood,prob=0.75)+
                   1.5*(quantile(wfood,prob=0.75)- quantile(wfood,prob=0.25)))

print(paste("Existem", length(out_wfood), "Outliers para a variáviel 'wfood'"))

# Visualizar observações dos outliers em questão
budget_food_data[out_wfood, ]
```

```
[1] "Existem 18 Outliers para a variáviel 'wfood'"
```

A data.frame: 18 × 6

	wfood	totexp	age	size	town	sex
	<dbl>	<dbl>	<int>	<int>	<int>	<dbl>
148	0.8588747	3617.040	78	2	1	1
150	0.8711138	397.560	68	1	1	0
155	0.8761410	652.032	75	1	1	0
209	0.8989888	4494.036	67	2	2	1
578	0.8900860	1920.192	60	2	2	1
579	0.8333538	2542.488	35	5	2	1
605	0.9569605	606.420	97	3	3	0
685	0.8735967	1218.576	77	2	2	1
1225	0.8607099	446.952	52	2	4	1
1336	0.8637669	4049.148	64	2	1	1
1533	0.8659533	787.248	71	1	3	0
1768	0.8935447	1059.036	59	7	2	1
1785	0.8404431	1370.220	67	2	3	1
1840	0.8144171	5279.064	68	2	2	0
1853	0.8201207	775.320	70	2	2	1
2030	0.8876501	2979.228	52	4	4	1
2172	0.8423464	3059.454	80	2	1	1
2420	0.8811572	1916.280	84	2	1	1

In [15]:
```
# Outliers da variável "totexp"
out_totexp <- which(totexp>quantile(totexp,prob=0.75)+
                    1.5*(quantile(totexp,prob=0.75)- quantile(totexp,prob=0.25)))

print(paste("Existem", length(out_totexp), "Outliers para a variáviel 'totexp'"))

# Visualizar observações dos outliers em questão
budget_food_data[out_totexp, ]
```

```
[1] "Existem 104 Outliers para a variáviel 'totexp'"
```

A data.frame: 104 × 6

	wfood	totexp	age	size	town	sex
	<dbl>	<dbl>	<int>	<int>	<int>	<dbl>
165	0.19357869	15203.62	39	4	3	1
170	0.37307878	13727.61	48	7	2	1
187	0.32698148	13428.21	51	6	3	1
197	0.22126745	12769.49	27	9	2	1
230	0.20603583	22160.26	52	2	2	0
253	0.24254123	19451.34	46	6	3	0
262	0.14491115	19932.88	52	8	3	1
264	0.13751562	17531.27	30	3	3	1
285	0.32611576	13427.50	51	7	3	1
292	0.21130570	17609.14	39	3	3	1
303	0.14877625	19454.88	48	8	4	1
311	0.26621841	15091.46	42	4	4	1
318	0.21593571	12570.41	60	7	4	1
320	0.24745630	14277.62	57	6	4	1
333	0.13019256	19176.40	50	8	4	1
346	0.18581968	12898.44	49	5	4	1
364	0.24514853	14795.11	42	4	4	1
395	0.13259323	13821.88	65	2	4	1
397	0.21216358	14267.41	59	7	4	1
398	0.04753049	36044.06	34	5	4	1
401	0.17020395	15068.04	65	3	4	1
422	0.10033332	13032.48	38	3	1	1
494	0.17514834	19008.76	46	5	3	1
510	0.15463602	16700.02	65	6	1	1
515	0.13614976	13144.58	61	3	2	1
525	0.23122705	15584.68	49	8	3	1
529	0.48671714	12926.36	43	5	3	1
571	0.06620027	21307.34	43	6	1	1
609	0.25140626	12921.49	38	6	3	1

	wfood	totexp	age	size	town	sex
	<dbl>	<dbl>	<int>	<int>	<int>	<dbl>
616	0.18554892	16562.75	64	9	2	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮
1922	0.16607831	12613.13	68	6	4	1
1940	0.15907759	13264.32	45	4	4	1
1948	0.18705476	13266.96	48	4	4	1
1970	0.17976744	12787.72	36	6	4	1
1997	0.12465494	12667.22	27	4	4	1
2002	0.11976124	15586.81	53	6	4	1
2007	0.03566139	12878.47	67	2	4	1
2011	0.14459614	15248.71	47	7	4	1
2016	0.06363736	17375.45	41	4	4	1
2019	0.19102621	19970.16	50	7	4	1
2047	0.08925939	19563.92	37	3	4	0
2093	0.14338929	18610.43	44	5	2	1
2094	0.25896071	19933.68	46	6	2	1
2159	0.36706705	13920.14	56	5	3	1
2165	0.36956588	19212.23	44	7	2	1
2190	0.17623382	18027.73	59	4	1	1
2192	0.02027410	21329.29	65	2	1	1
2212	0.07581148	25520.04	50	5	3	1
2215	0.22758873	14029.73	56	5	3	1
2234	0.23009087	20905.24	61	3	2	1
2255	0.11027955	14756.97	59	4	1	1
2275	0.07125111	18378.16	48	5	3	1
2326	0.38548279	25510.68	48	10	4	1
2360	0.01991481	68385.28	30	3	4	1
2379	0.23808457	12775.66	56	4	4	1
2381	0.08693240	16355.05	58	3	4	0
2463	0.15501486	19939.92	48	5	2	1

	wfood	totexp	age	size	town	sex
	<dbl>	<dbl>	<int>	<int>	<int>	<dbl>
2464	0.24102992	12702.39	57	5	2	1
2483	0.12710192	12754.74	68	3	2	1
2484	0.27962862	14851.96	45	5	2	1

In [16]:
```
# Outliers da variável "age"
out_age <- which(age>quantile(age,prob=0.75)+
                 1.5*(quantile(age,prob=0.75)- quantile(age,prob=0.25)))

print(paste("Existem", length(out_age), "Outliers para a variáviel 'age'"))

# Visualizar observações dos outliers em questão
budget_food_data[out_age, ]
```

```
[1] "Existem 1 Outliers para a variáviel 'age'"
```

A data.frame: 1 × 6

	wfood	totexp	age	size	town	sex
	<dbl>	<dbl>	<int>	<int>	<int>	<dbl>
605	0.9569605	606.42	97	3	3	0

In [17]:
```
# Outliers da variável "size"
out_size <- which(size>quantile(size,prob=0.75)+
                 1.5*(quantile(size,prob=0.75)- quantile(size,prob=0.25)))

print(paste("Existem", length(out_size), "Outliers para a variáviel 'size'"))

# Visualizar observações dos outliers em questão
budget_food_data[out_size, ]
```

```
[1] "Existem 17 Outliers para a variáviel 'size'"
```

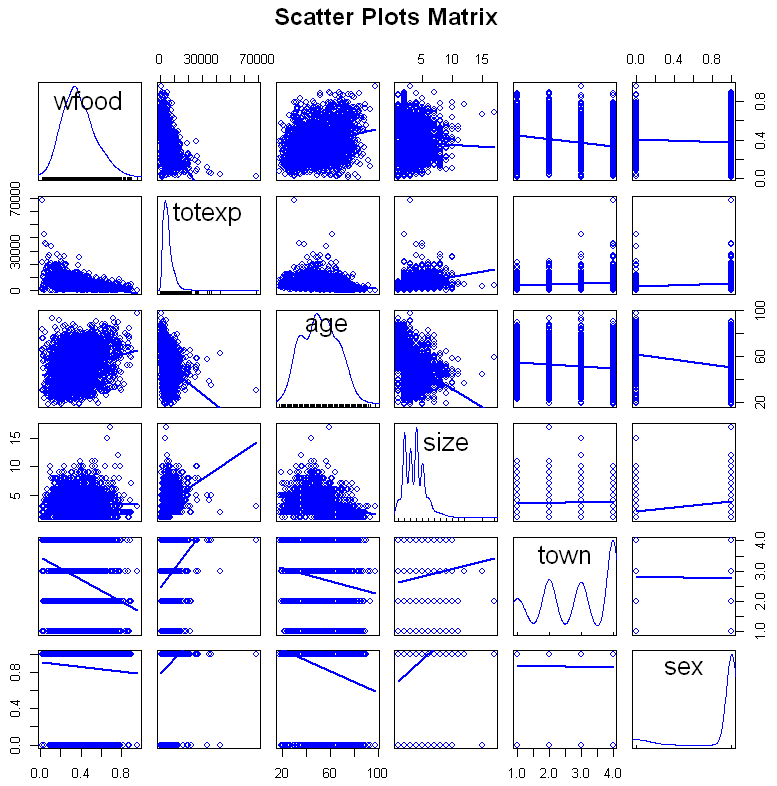
A data.frame: 17 × 6

	wfood	totexp	age	size	town	sex
	<dbl>	<dbl>	<int>	<int>	<int>	<dbl>
281	0.5085926	5171.448	36	10	3	1
512	0.3938285	4966.446	44	11	1	1
624	0.6909641	4129.812	59	17	2	1
748	0.7898063	1400.394	32	10	4	0
842	0.4035172	6661.128	37	10	4	1
887	0.1004179	13357.056	44	10	4	1
952	0.7397707	1958.196	47	10	2	1
954	0.5114084	5095.998	47	11	2	1
1004	0.6380957	2243.814	42	10	2	1
1111	0.5397637	6277.992	57	11	1	1
1365	0.5452784	4384.080	44	10	1	1
1529	0.3338047	11431.116	47	10	3	1
1590	0.2314929	26086.152	58	10	4	1
1913	0.4236173	6515.202	47	10	4	1
2072	0.6628213	3341.136	44	15	3	0
2080	0.7719328	4833.996	36	12	3	1
2326	0.3854828	25510.680	48	10	4	1

Correlações

Nesta fase, foram feitas as matrizes de dispersão e de correlação com a finalidade de perceber que tipo de relação existe entre as variáveis.

In [18]:
```r
# Matriz de dispersão

scatterplotMatrix(
      budget_food_data,
      smooth = FALSE,
      main = "Scatter Plots Matrix"
)
```



Observações:

- Relação mais evidente ocorre entre totexp e wfood, parece uma relação não linear;
- Existe uma leve tendência positiva entre age e wfood;
- Valores dispersos no size que pode surgir uma relação não linear com wfood.

In [19]:

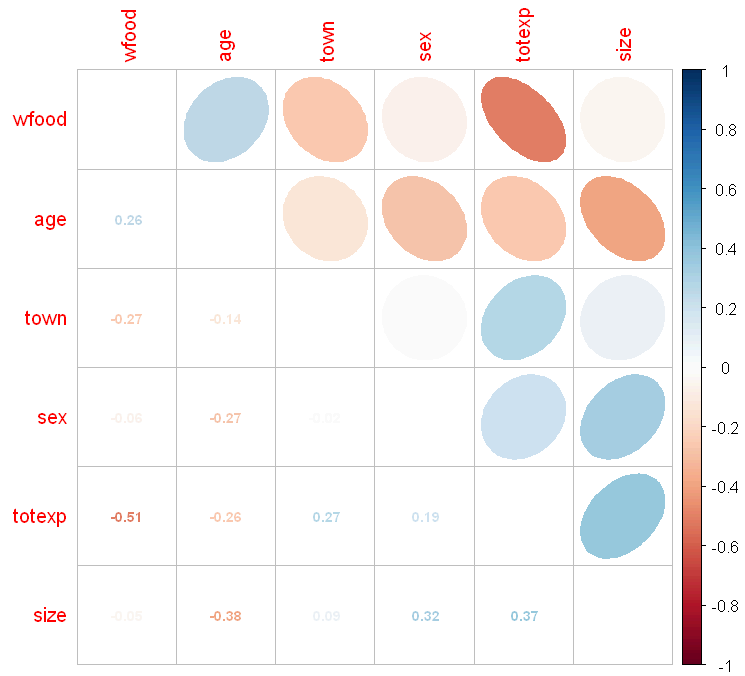
```r
# ====== Matriz de correlação, sex e town com Pearson e Spearman ======

# Criar a matriz completa com todas as variáveis
cor_matrix <- cor(budget_food_data[, c("wfood", "totexp", "age", "size")], method
= "pearson")

# Correlação para a dummy (sex)
cor_sex <- sapply(budget_food_data[, c("wfood", "totexp", "age", "size")],
function(var) {
  cor(var, budget_food_data$sex, method = "pearson")
})

# Correlação de Spearman para a variável ordinal (town)
cor_town <- sapply(budget_food_data[, c("wfood", "totexp", "age", "size")],
function(var) {
  cor(var, budget_food_data$town, method = "spearman")
})

# Adicionar colunas para 'town' e 'sex'
cor_matrix <- cbind(cor_matrix, "town" = cor_town, "sex" = cor_sex)

# Criar uma nova linha para 'town' e 'sex' e adicionar corretamente
new_row_town <- c(cor_town, 1, cor(budget_food_data$sex, budget_food_data$town,
```

```
method = "spearman"))
new_row_sex <- c(cor_sex, cor(budget_food_data$sex, budget_food_data$town, method
= "spearman"), 1)

cor_matrix <- rbind(cor_matrix, "town" = new_row_town, "sex" = new_row_sex)

# Matriz final
cor_matrix
```

A matrix: 6 × 6 of type dbl

	wfood	totexp	age	size	town	sex
wfood	1.00000000	-0.5082536	0.2573123	-0.04663193	-0.26754086	-0.06302258
totexp	-0.50825357	1.0000000	-0.2623574	0.37126596	0.27174155	0.19201889
age	0.25731229	-0.2623574	1.0000000	-0.38470869	-0.13609020	-0.27012098
size	-0.04663193	0.3712660	-0.3847087	1.00000000	0.08750351	0.32155099
town	-0.26754086	0.2717415	-0.1360902	0.08750351	1.00000000	-0.01993470
sex	-0.06302258	0.1920189	-0.2701210	0.32155099	-0.01993470	1.00000000

In [20]:
```
# visualizar as correlações num plot
par(oma = c(2, 2, 2, 2))
corrplot.mixed(
    cor_matrix,
    order = "hclust",
    tl.pos = "lt",
    number.cex = 0.7,
    upper = "ellipse"
)
```



Observações:

- A relação entre as variáveis independentes são fracas, o que é importante para a construção do modelo posteriormente.
- A relação mais forte ocorre entre wfood e totexp -> -0.51, uma correlação moderada nagetiva -> À medida que totexp aumenta, a proporção direcionada à alimentação diminui.

Modelos

Nesta secção, testámos várias hipóteses para encontrar o modelo ótimo que melhor prevê a nossa variável target - *wfood*

Modelo1

No primeiro modelo, incluiram-se todas as variáveis da base de dados sem transformação nenhuma.

In [21]:
```
modelo1 <- lm(wfood ~. , data = budget_food_data)
summary(modelo1)

# Para verificar a multicolinearidade entre as variáveis dependentes
vif(modelo1)
```

```r
# Métrica para averiguar a qualidade de um modelo
AIC(modelo1)
```

```
Call:
lm(formula = wfood ~ ., data = budget_food_data)

Residuals:
     Min       1Q   Median       3Q      Max
-0.47722 -0.08708 -0.01473  0.07892  1.05604

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  3.573e-01  1.774e-02  20.139   <2e-16 ***
totexp      -2.102e-05  7.463e-07 -28.164   <2e-16 ***
age          2.108e-03  1.994e-04  10.568   <2e-16 ***
size         2.053e-02  1.733e-03  11.846   <2e-16 ***
town        -2.163e-02  2.487e-03  -8.696   <2e-16 ***
sex          5.765e-03  8.561e-03   0.673    0.501
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1336 on 2484 degrees of freedom
Multiple R-squared:  0.3375,    Adjusted R-squared:  0.3362
F-statistic: 253.1 on 5 and 2484 DF,  p-value: < 2.2e-16
```

totexp: 1.23566664537053 **age:** 1.24158525901316 **size:** 1.35707711371421 **town:** 1.0681419846294 **sex:** 1.15973929753808

-2948.77499913199

Observações:

- A variável sex tem um p-value de 0.501, pelo que não é significativa a 10%.
- O R^2 é 0.3375.
- O AIC é aproximadamente -2948.77.

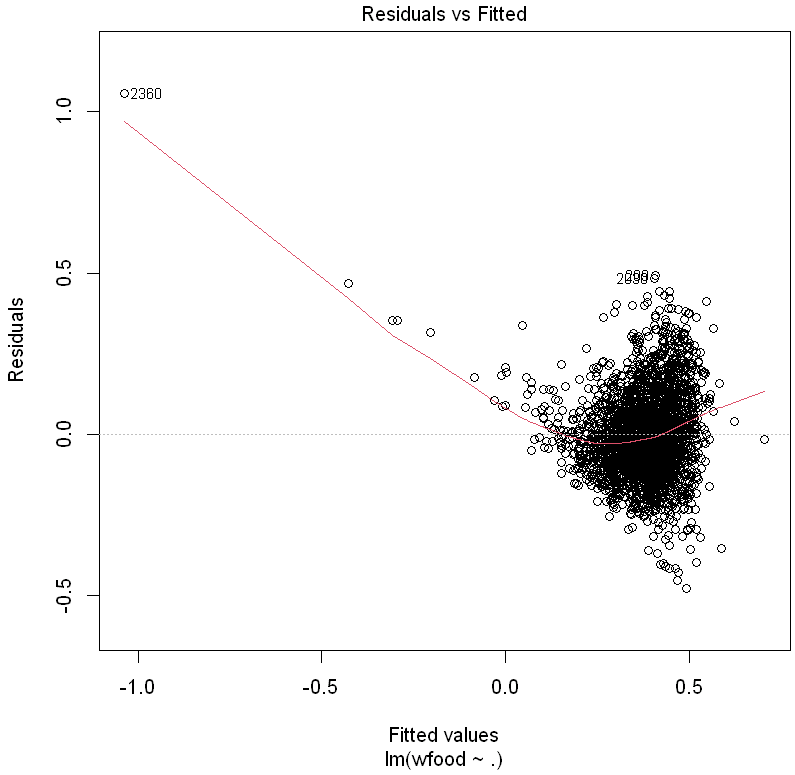
Verificação dos pressupostos dos resíduos do modelo1

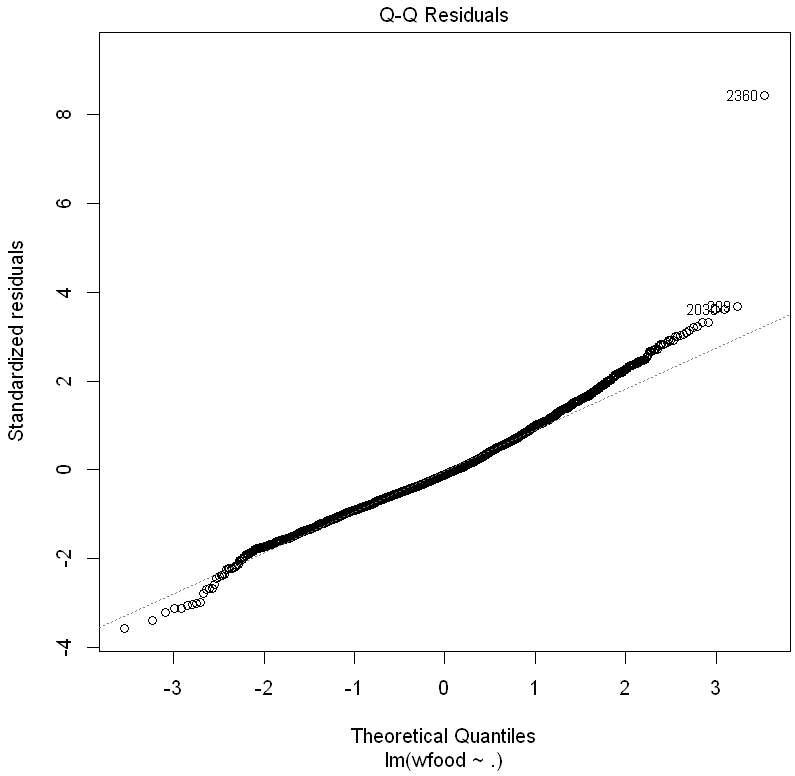
In [22]:
```r
# Verificacão dos pressupostos dos residuos

mean(modelo1$residuals) # média nula
bptest(modelo1) # variância constante
bgtest(modelo1) # independênia entre resíduos
jarque.bera.test(modelo1$residuals) # normalidade

plot(modelo1)
```

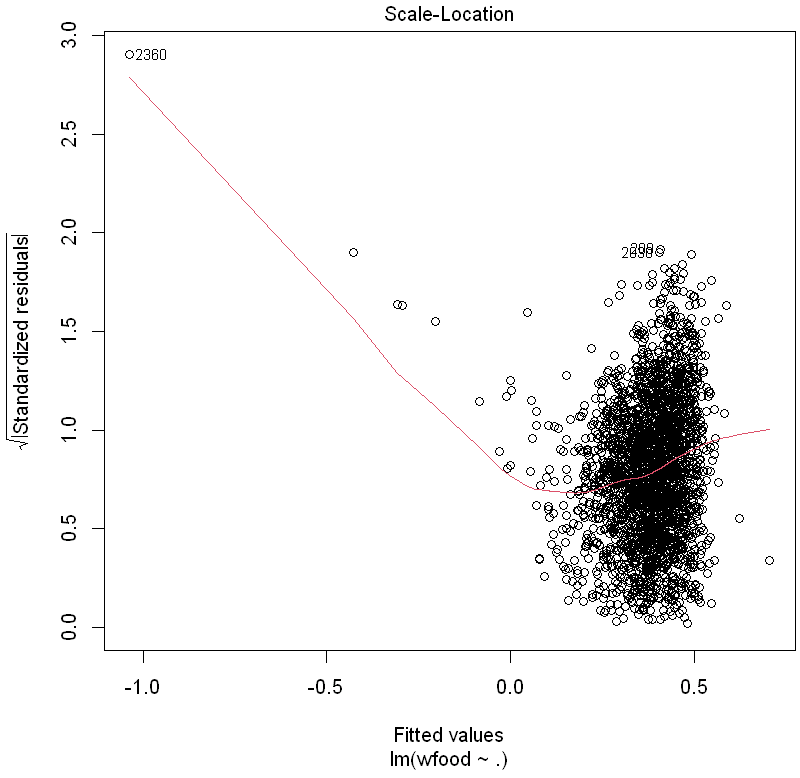
1.36402267022799e-17

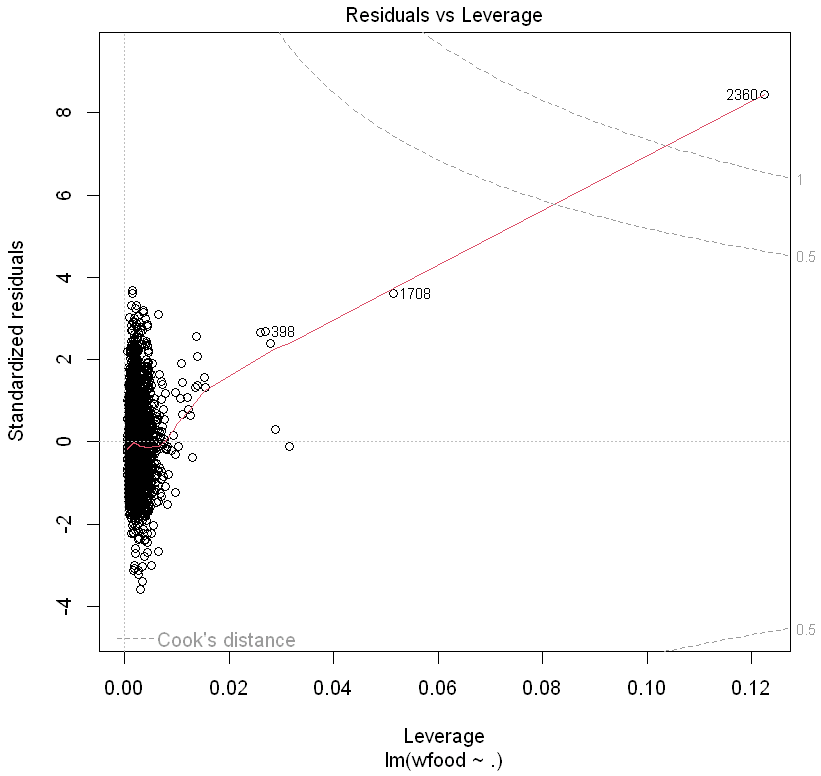
```
        studentized Breusch-Pagan test

data:  modelo1
BP = 174.83, df = 5, p-value < 2.2e-16
```

```
        Breusch-Godfrey test for serial correlation of order up to 1

data:  modelo1
LM test = 30.341, df = 1, p-value = 3.624e-08
        Jarque Bera Test

data:  modelo1$residuals
X-squared = 536.06, df = 2, p-value < 2.2e-16
```




Q-Q Residuals
Standardized residuals
Theoretical Quantiles
lm(wfood ~ .)
2360


Scale-Location
√|Standardized residuals|
Fitted values
lm(wfood ~ .)
2360



Observações:

- A média dos reísiduos é estatisticamente nula;
- Com o p-value < 2.2e-16 no teste Breusch-Pagan, reijata-se a H0 de que os resíduos têm variância constante, logo, existe problema de heterocedasticidade -> **Pressuposto de homocedasticidade NÃO verificado**;
- Com o p-value = 3.642e-8 no teste Breusch-Godfrey, reijata-se a H0 de que os resíduos são independentes, logo, existe correlação -> **Pressuposto da ndependência NÃO verificado**;
- Apesar do pressuposto da normalidade dos resíduos não está verificado neste caso, o Teorema do Limite Central garante que, com o aumento do tamanho da amostra, a distribuição dos resíduos tende a se aproximar de uma distribuição normal.

Modelo2

Neste modelo, aplicamos a transformação logarítmica à variável "*totexp*" devido à sua grande amplitude e convertimos a variável "*town*" em dummy, adotando town4 como categoria de referência, pois é a mais frequente na base de dados.

In [23]:
```
# Tabela de frequência absoluta da variável town
table(budget_food_data$town)
```

```
  1   2   3   4 
441 599 580 870 
```

In [24]:
```r
# Aplicar town4 como categoria de referência
budget_food_data$town <- relevel(as.factor(budget_food_data$town), ref = "4")


modelo2 <- lm(wfood ~ log(totexp) + age + size + sex + as.factor(town) , data =
budget_food_data)
summary(modelo2)
AIC(modelo2)
vif(modelo2)
```

```
Call:
lm(formula = wfood ~ log(totexp) + age + size + sex + as.factor(town),
    data = budget_food_data)

Residuals:
     Min       1Q   Median       3Q      Max
-0.66323 -0.07936 -0.01154  0.07364  0.55323

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)       1.338809   0.038283  34.971  < 2e-16 ***
log(totexp)      -0.140530   0.004268 -32.926  < 2e-16 ***
age               0.001126   0.000196   5.744 1.03e-08 ***
size              0.025226   0.001693  14.897  < 2e-16 ***
sex               0.040710   0.008345   4.879 1.14e-06 ***
as.factor(town)1  0.052920   0.007756   6.823 1.12e-11 ***
as.factor(town)2  0.022281   0.006993   3.186  0.00146 **
as.factor(town)3  0.017993   0.006892   2.611  0.00909 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1281 on 2482 degrees of freedom
Multiple R-squared:  0.3917,    Adjusted R-squared:   0.39
F-statistic: 228.4 on 7 and 2482 DF,  p-value: < 2.2e-16
```

```
-3157.28140153064
```

A matrix: 5 × 3 of type dbl

	GVIF	Df	GVIF^(1/(2*Df))
log(totexp)	1.547588	1	1.244021
age	1.305031	1	1.142380
size	1.410293	1	1.187558
sex	1.199078	1	1.095024
as.factor(town)	1.107369	3	1.017143

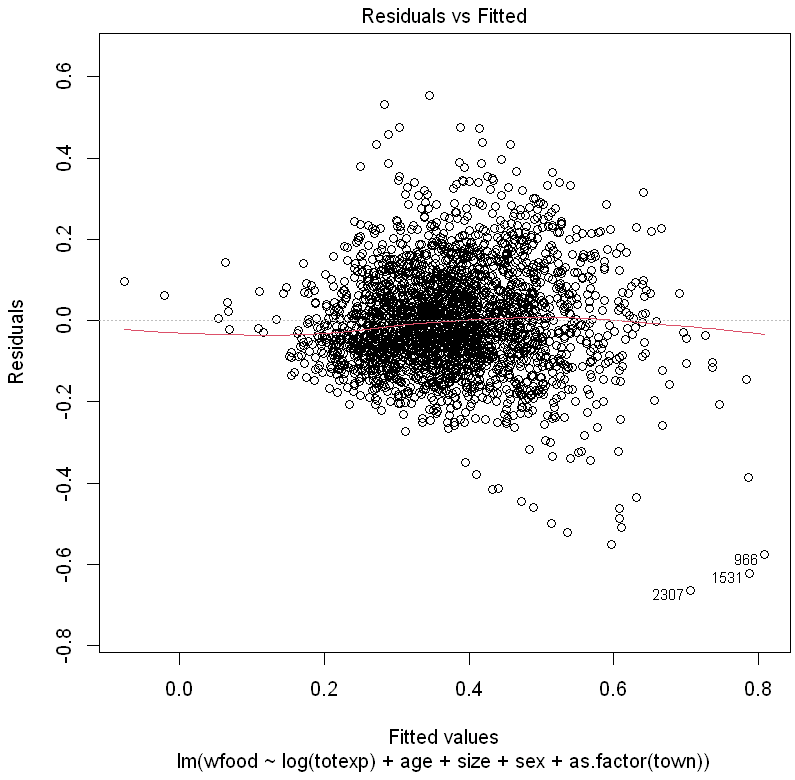
Observações:

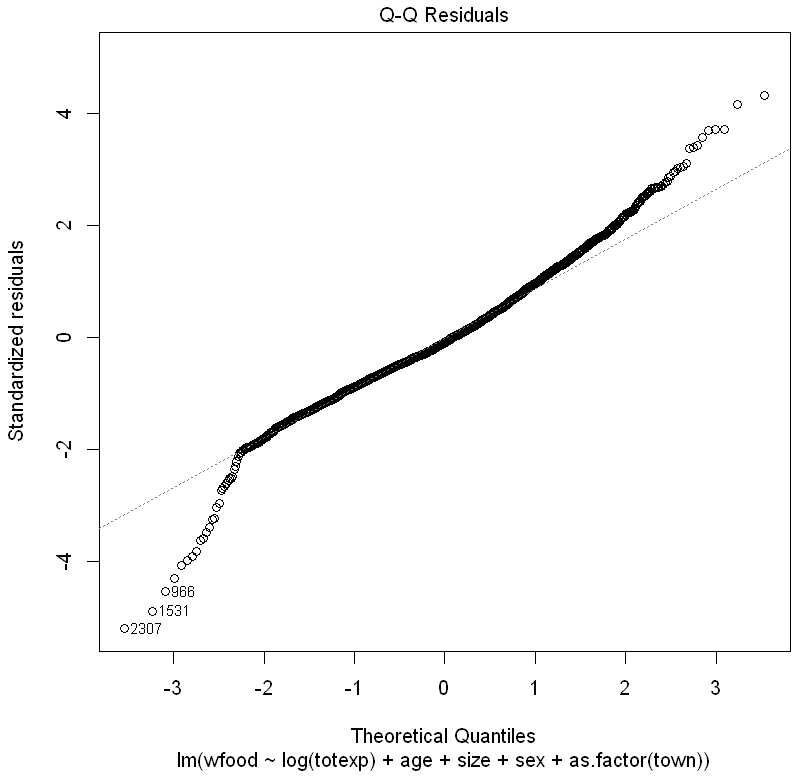
- Todas as variáveis são relevantes;
- O R^2 é 0.3917, melhorou em relação ao modelo1;
- O AIC é aproximadamente -3157.281, melhorou em relação ao modelo1

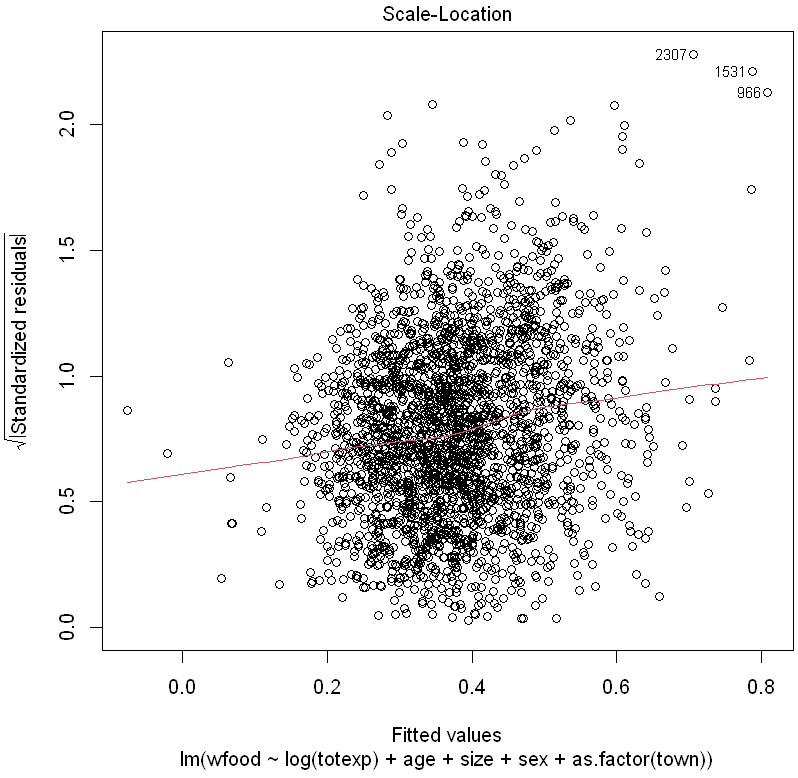
In [25]:
```r
# Verificacão dos pressupostos dos residuos

mean(modelo2$residuals) #média nula
bptest(modelo2) # variância constante
bgtest(modelo2) # independência entre os resíduos
jarque.bera.test(modelo2$residuals) # normalidade

plot(modelo2)
```

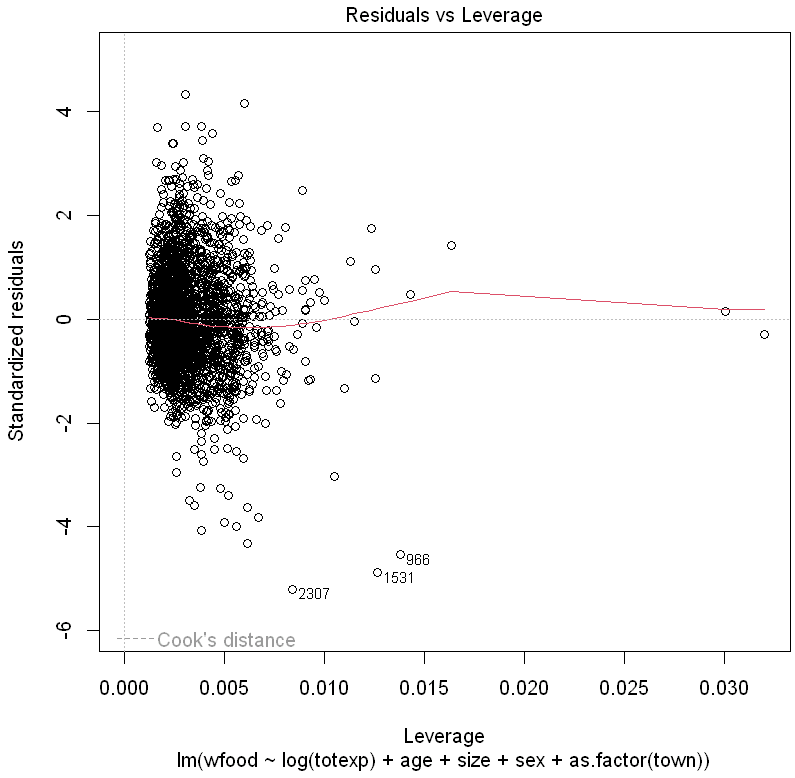
```
6.05241439417758e-19
```

```
        studentized Breusch-Pagan test

data:  modelo2
BP = 164.31, df = 7, p-value < 2.2e-16
        Breusch-Godfrey test for serial correlation of order up to 1

data:  modelo2
LM test = 24.339, df = 1, p-value = 8.079e-07
        Jarque Bera Test

data:  modelo2$residuals
X-squared = 307.25, df = 2, p-value < 2.2e-16
```




Q-Q Residuals
Standardized residuals
Theoretical Quantiles
lm(wfood ~ log(totexp) + age + size + sex + as.factor(town))
966
1531
2307


Scale-Location
√|Standardized residuals|
Fitted values
lm(wfood ~ log(totexp) + age + size + sex + as.factor(town))
2307
1531
966



Observações:

- A média dos reísiduos é estatisticamente nula;
- Com o p-value < 2.2e-16 no teste Breusch-Pagan, rejeita-se a H0 de que os resíduos têm variância constante, logo, existe problema de heterocedasticidade -> **Pressuposto da variância constante NÃO verificado**;
- Com o p-value = 8.079e-07 no teste Breusch-Godfrey, rejeita-se a H0 de que os resíduos são independentes, logo, existe correlação -> **Pressuposto da independência NÃO verificado**;
- Apesar do pressuposto da normalidade dos resíduos não está verificado neste caso, o Teorema do Limite Central garante que, com o aumento do tamanho da amostra, a distribuição dos resíduos tende a se aproximar de uma distribuição normal.

Modelo3

Neste modelo, incluimos 2 interações: totexp:size e totexp:as.factor(town). Devido à forte multicolinearidade, retirámos as variáveis as.factor(town) e size.

In [26]:
```
modelo3 <- lm(wfood ~ log(totexp)  + sex  + age + log(totexp):size
              + log(totexp):as.factor(town) , data = budget_food_data)
summary(modelo3)
AIC(modelo3)
vif(modelo3)
```

```
Call:
lm(formula = wfood ~ log(totexp) + sex + age + log(totexp):size + 
    log(totexp):as.factor(town), data = budget_food_data)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.66658 -0.08003 -0.01176  0.07426  0.55035 

Coefficients:
                              Estimate Std. Error t value Pr(>|t|)    
(Intercept)                  1.4331659  0.0386752  37.056  < 2e-16 ***
log(totexp)                 -0.1512046  0.0044615 -33.891  < 2e-16 ***
sex                          0.0444174  0.0083419   5.325 1.10e-07 ***
age                          0.0010341  0.0001956   5.287 1.35e-07 ***
log(totexp):size             0.0028255  0.0001988  14.214  < 2e-16 ***
log(totexp):as.factor(town)1 0.0064044  0.0009468   6.764 1.66e-11 ***
log(totexp):as.factor(town)2 0.0030856  0.0008425   3.662 0.000255 ***
log(totexp):as.factor(town)3 0.0022162  0.0008164   2.715 0.006681 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1286 on 2482 degrees of freedom
Multiple R-squared:  0.3873,    Adjusted R-squared:  0.3856 
F-statistic: 224.2 on 7 and 2482 DF,  p-value: < 2.2e-16
```

-3139.25711359043

```
there are higher-order terms (interactions) in this model
consider setting type = 'predictor'; see ?vif
```

A matrix: 5 × 3 of type dbl

	GVIF	Df	GVIF^(1/(2*Df))
log(totexp)	1.678834	1	1.295698
sex	1.189655	1	1.090713
age	1.290138	1	1.135842
log(totexp):size	1.601916	1	1.265668
log(totexp):as.factor(town)	1.061264	3	1.009959

Observações:

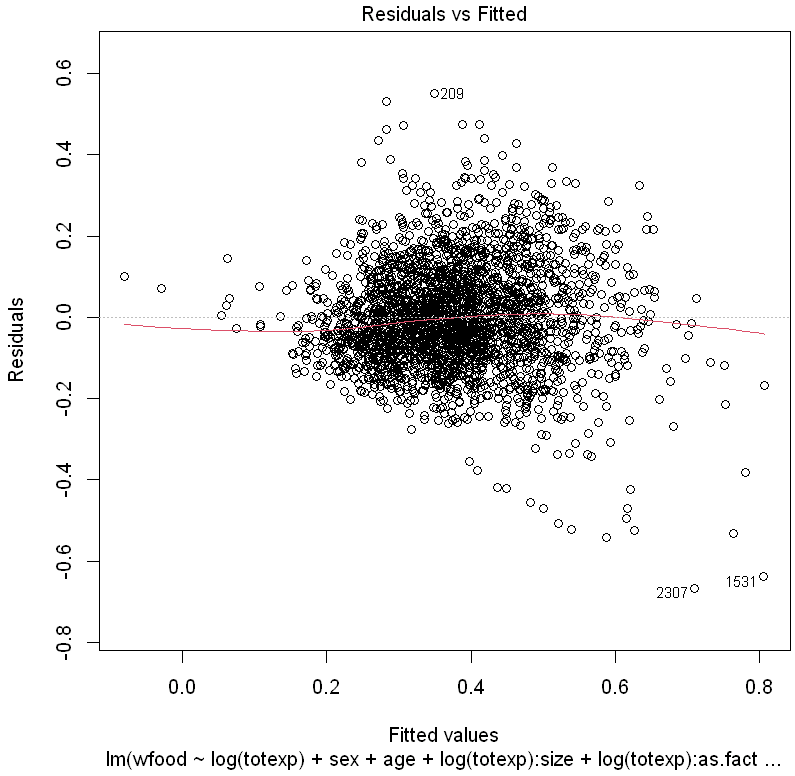
- Todas as variáveis são relevantes;
- O R^2 é 0.3873, piorou em relação ao modelo2;
- O AIC é aproximadamente -3139.257, também piorou em relação ao modelo2.

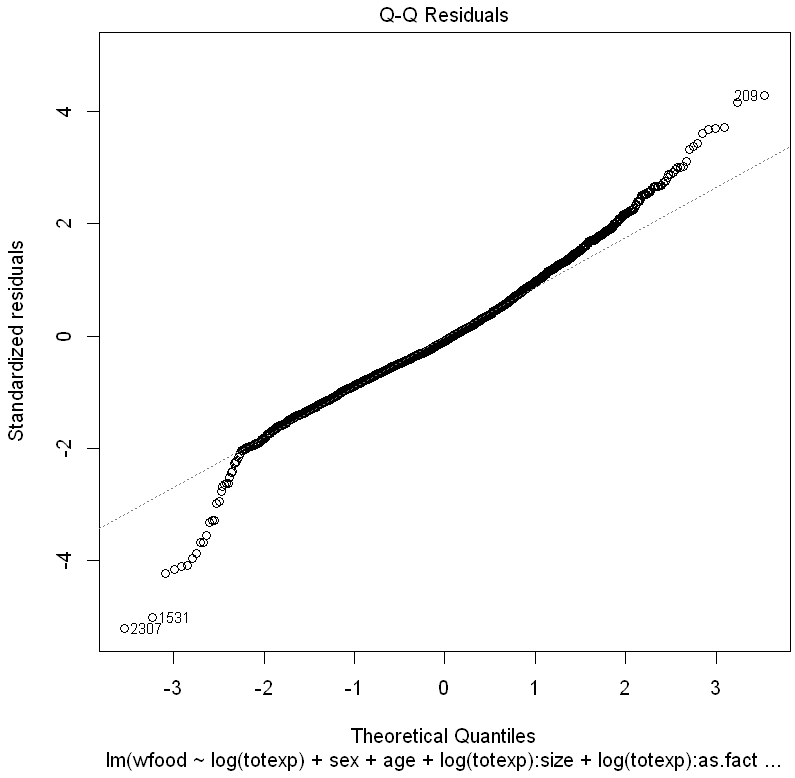
In [27]:
```r
# Verificação dos pressupostos dos resíduos

mean(modelo3$residuals) # média nula
bptest(modelo3) # variância constante
bgtest(modelo3) # independência entre os resíduos
jarque.bera.test(modelo3$residuals) # normalidade
```

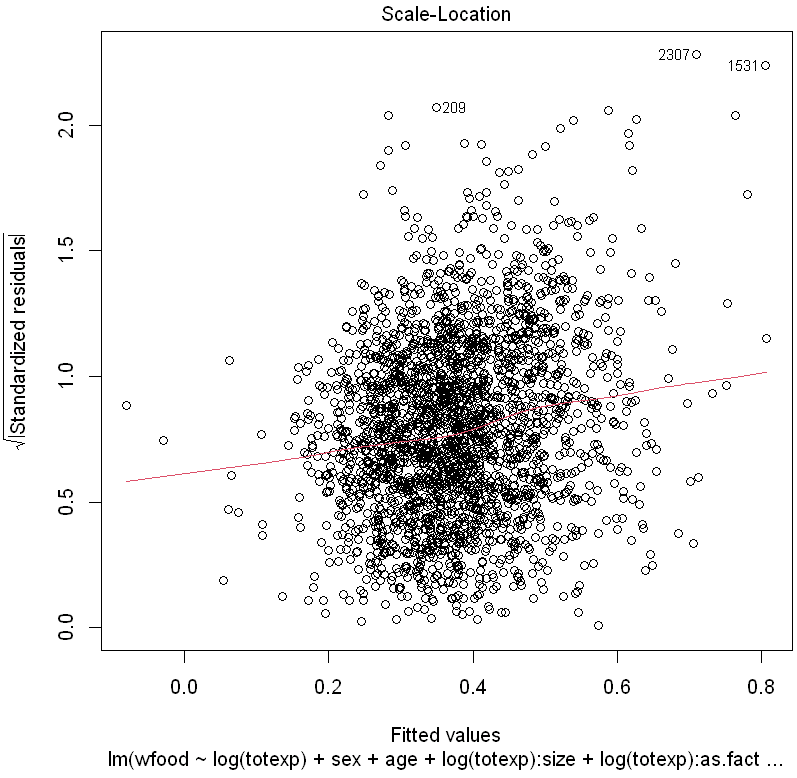
```
plot(modelo3)
```

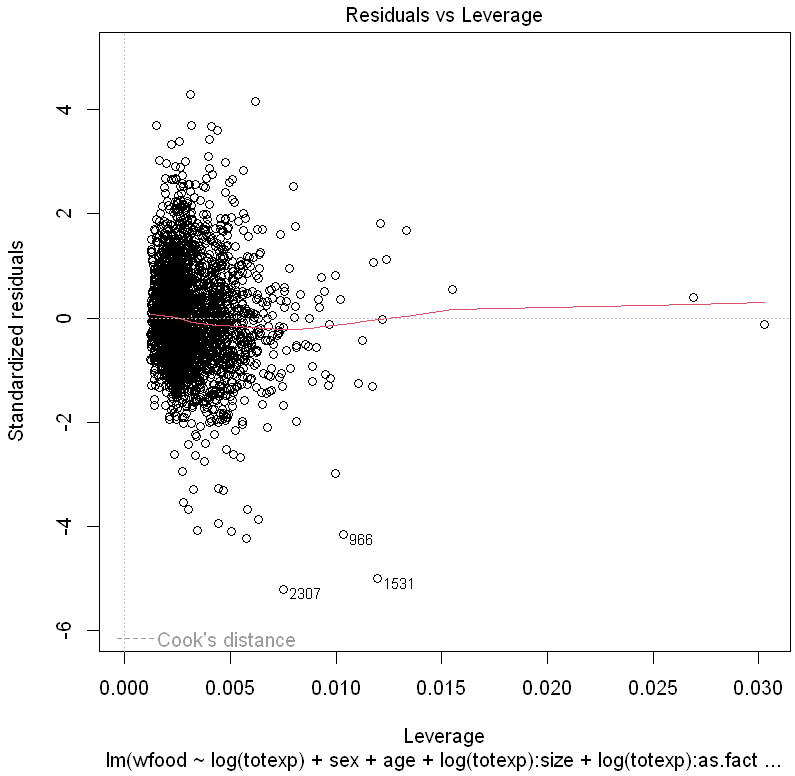
```
-8.8121579526674e-18
```

```
        studentized Breusch-Pagan test

data:  modelo3
BP = 165.37, df = 7, p-value < 2.2e-16
        Breusch-Godfrey test for serial correlation of order up to 1

data:  modelo3
LM test = 23.977, df = 1, p-value = 9.747e-07
        Jarque Bera Test

data:  modelo3$residuals
X-squared = 298.78, df = 2, p-value < 2.2e-16
```




Q-Q Residuals
Standardized residuals
Theoretical Quantiles
lm(wfood ~ log(totexp) + sex + age + log(totexp):size + log(totexp):as.fact ...
209
1531
2307


Scale-Location
√|Standardized residuals|
Fitted values
lm(wfood ~ log(totexp) + sex + age + log(totexp):size + log(totexp):as.fact ...
2307
1531
209



Observações:

- A média dos reísiduos é estatisticamente nula;
- Com o p-value < 2.2e-16 no teste Breusch-Pagan, rejeita-se a H0 de que os resíduos têm variância constante, logo, existe problema de heterocedasticidade -> **Pressuposto da variância constante NÃO verificado**;
- Com o p-value = 4.13e-06 no teste Breusch-Godfrey, rejeita-se a H0 de que os resíduos são independentes, logo, existe correlação -> **Pressuposto da independência NÃO verificado**;
- Apesar do pressuposto da normalidade dos resíduos não está verificado neste caso, o Teorema do Limite Central garante que, com o aumento do tamanho da amostra, a distribuição dos resíduos tende a se aproximar de uma distribuição normal.

Modelo4

Para lidar com problemas de heterocedasticidade, foi decidido desenvolver um modelo de regrassão ponderado (WSL), onde os pesos foram estimados por um modelo de regressão robusta.

In [28]:
```
# Modelo robusto que calcula os pesos
modelo4_auxiliar <- rlm(wfood ~ log(totexp)  + sex  + age  + log(totexp):size
                        + log(totexp):as.factor(town) , data = budget_food_data)

# Extração dos pesos
```

```r
weights <- modelo4_auxiliar$w

# Modelo WLS
modelo4 <- lm(wfood ~ log(totexp)  + sex  + age  + log(totexp):size
              + log(totexp):as.factor(town) , data = budget_food_data, weights =
weights )

summary(modelo4)
AIC(modelo4)
vif(modelo4)
```

```
Call:
lm(formula = wfood ~ log(totexp) + sex + age + log(totexp):size +
    log(totexp):as.factor(town), data = budget_food_data, weights = weights)

Weighted Residuals:
     Min       1Q   Median       3Q      Max
-0.32345 -0.07535 -0.00628  0.07800  0.29152

Coefficients:
                              Estimate Std. Error t value Pr(>|t|)
(Intercept)                  1.5390938  0.0339638  45.316  < 2e-16 ***
log(totexp)                 -0.1639672  0.0039169 -41.862  < 2e-16 ***
sex                          0.0447809  0.0073011   6.133 9.98e-10 ***
age                          0.0008700  0.0001693   5.138 3.00e-07 ***
log(totexp):size             0.0030405  0.0001717  17.704  < 2e-16 ***
log(totexp):as.factor(town)1 0.0055246  0.0008233   6.711 2.39e-11 ***
log(totexp):as.factor(town)2 0.0027957  0.0007279   3.841 0.000126 ***
log(totexp):as.factor(town)3 0.0021618  0.0006999   3.089 0.002033 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1081 on 2482 degrees of freedom
Multiple R-squared:  0.4793,    Adjusted R-squared:  0.4778
F-statistic: 326.3 on 7 and 2482 DF,  p-value: < 2.2e-16
-3821.54339263018
there are higher-order terms (interactions) in this model
consider setting type = 'predictor'; see ?vif
```

A matrix: 5 × 3 of type dbl

	GVIF	Df	GVIF^(1/(2*Df))
log(totexp)	1.665493	1	1.290540
sex	1.180143	1	1.086344
age	1.271656	1	1.127677
log(totexp):size	1.583288	1	1.258288
log(totexp):as.factor(town)	1.062244	3	1.010115

Observações:

- Todas as variáveis são relevantes;

- O R^2 é 0.4793, melhorou consideravelmente em relação aos modelos anteriores;
- O AIC é aproximadamente -3821.543, melhorou em relação aos modelos anteriores

In [29]:
```
# Verificação dos pressupostos dos resíduos

mean(modelo4$residuals) # média nula
bptest(modelo4) # variância constante
bgtest(modelo4) # independência entre resíduos
jarque.bera.test(modelo4$residuals) #normalidade
```

0.00324848452120172

```
        studentized Breusch-Pagan test

data:  modelo4
BP = 47.459, df = 7, p-value = 4.542e-08
        Breusch-Godfrey test for serial correlation of order up to 1

data:  modelo4
LM test = 23.977, df = 1, p-value = 9.747e-07
        Jarque Bera Test

data:  modelo4$residuals
X-squared = 394.45, df = 2, p-value < 2.2e-16
```

Observações:

- A média dos reísiduos é estatisticamente nula;
- Com o p-value = 4.542e-08 no teste Breusch-Pagan, rejeita-se a H0 de que os resíduos têm variância constante, logo, existe problema de heterocedasticidade -> **Pressuposto da variância constante NÃO verificado**;
- Com o p-value = 9.747e-07 no teste Breusch-Godfrey, rejeita-se a H0 de que os resíduos são independentes, logo, existe correlação -> **Pressuposto da independência NÃO verificado**;
- Apesar do pressuposto da normalidade dos resíduos não está verificado neste caso, o Teorema do Limite Central garante que, com o aumento do tamanho da amostra, a distribuição dos resíduos tende a se aproximar de uma distribuição normal.

Modelo5

Neste modelo, a variável dependente passou a ser o valor absoluto da despesa total direcionada para alimentação - wfood*totexp - dada a grande amplitude que totex apresenta, logritmizámos a nossa variável dependente. Utilizámos pesos de 1/sqrt(totexp) com o efeito de reduzir a influência de observações com despesas totais muito elevadas e dão maior peso às de menor despesa.

In [30]:
```
modelo5<- lm(log(wfood*totexp) ~ log(totexp) + I(totexp^2)  +  sex + age +
               log(totexp):size + log(totexp):as.factor(town),
```

```
            data = budget_food_data, weights = 1/sqrt(totexp))

summary(modelo5)
AIC(modelo5)
vif(modelo5)
```

```
Call:
lm(formula = log(wfood * totexp) ~ log(totexp) + I(totexp^2) +
    sex + age + log(totexp):size + log(totexp):as.factor(town),
    data = budget_food_data, weights = 1/sqrt(totexp))

Weighted Residuals:
     Min       1Q   Median       3Q      Max
-0.62257 -0.02437  0.00404  0.03261  0.25232

Coefficients:
                              Estimate Std. Error t value Pr(>|t|)
(Intercept)                  7.492e-01  1.330e-01   5.633 1.97e-08 ***
log(totexp)                  7.116e-01  1.558e-02  45.662  < 2e-16 ***
I(totexp^2)                 -1.293e-09  1.421e-10  -9.100  < 2e-16 ***
sex                          7.919e-02  2.592e-02   3.055  0.00227 **
age                          3.812e-03  6.831e-04   5.581 2.65e-08 ***
log(totexp):size             9.515e-03  7.359e-04  12.930  < 2e-16 ***
log(totexp):as.factor(town)1 2.102e-02  3.265e-03   6.436 1.46e-10 ***
log(totexp):as.factor(town)2 8.505e-03  2.985e-03   2.850  0.00441 **
log(totexp):as.factor(town)3 6.509e-03  3.003e-03   2.168  0.03027 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.05738 on 2481 degrees of freedom
Multiple R-squared:  0.6847,    Adjusted R-squared:  0.6837
F-statistic: 673.6 on 8 and 2481 DF,  p-value: < 2.2e-16
```

3175.3385218522

```
there are higher-order terms (interactions) in this model
consider setting type = 'predictor'; see ?vif
```

A matrix: 6 × 3 of type dbl

	GVIF	Df	GVIF^(1/(2*Df))
log(totexp)	2.215201	1	1.488355
I(totexp^2)	1.283793	1	1.133046
sex	1.295263	1	1.138096
age	1.505720	1	1.227078
log(totexp):size	1.798954	1	1.341251
log(totexp):as.factor(town)	1.066869	3	1.010846

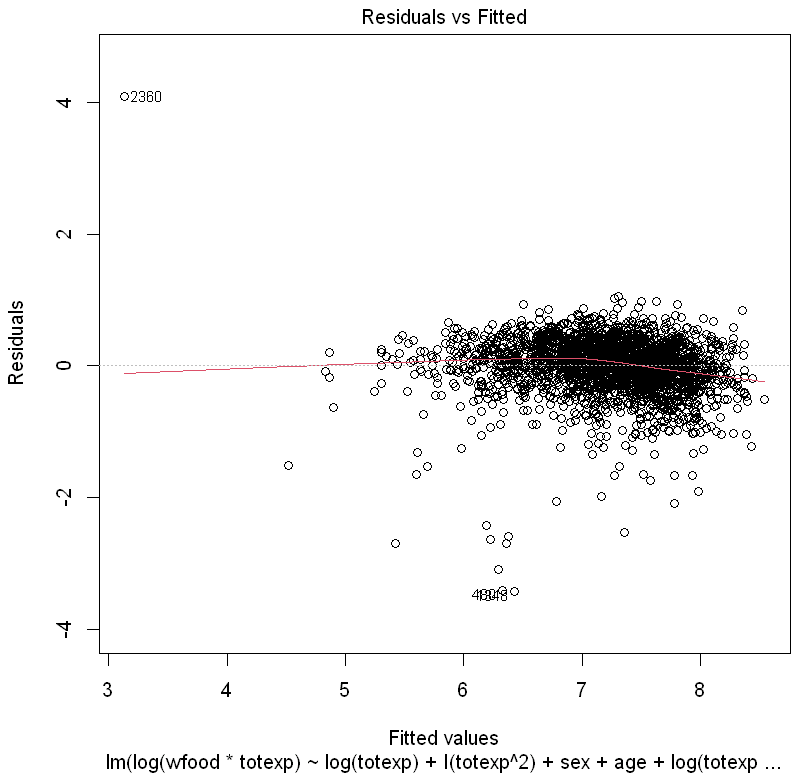
Observações:

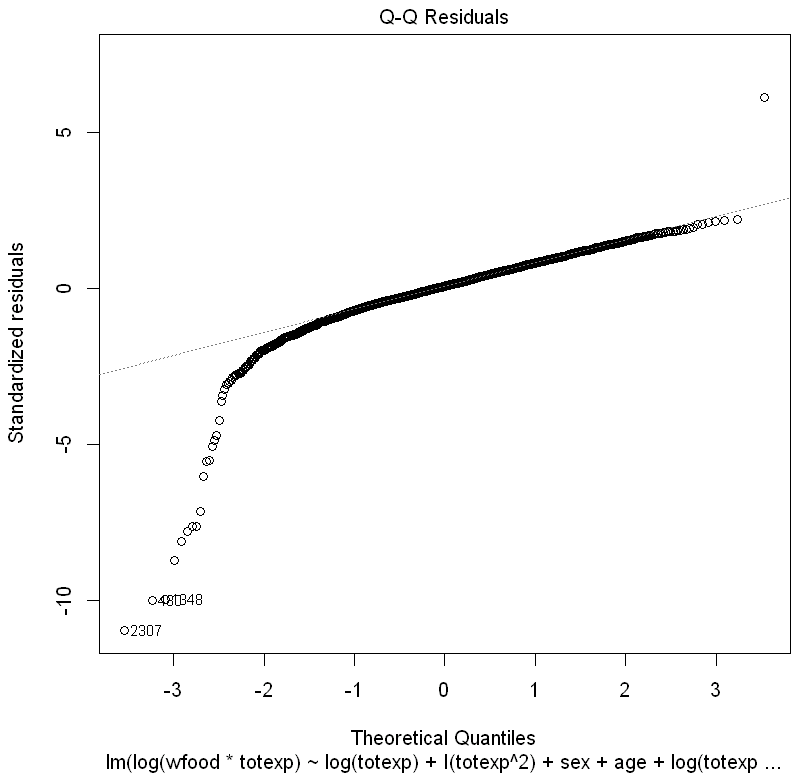
- Quase todas as variáveis são relevantes;
- O R^2 é 0.6847, melhorou significativamente em relação aos modelos anteriores;

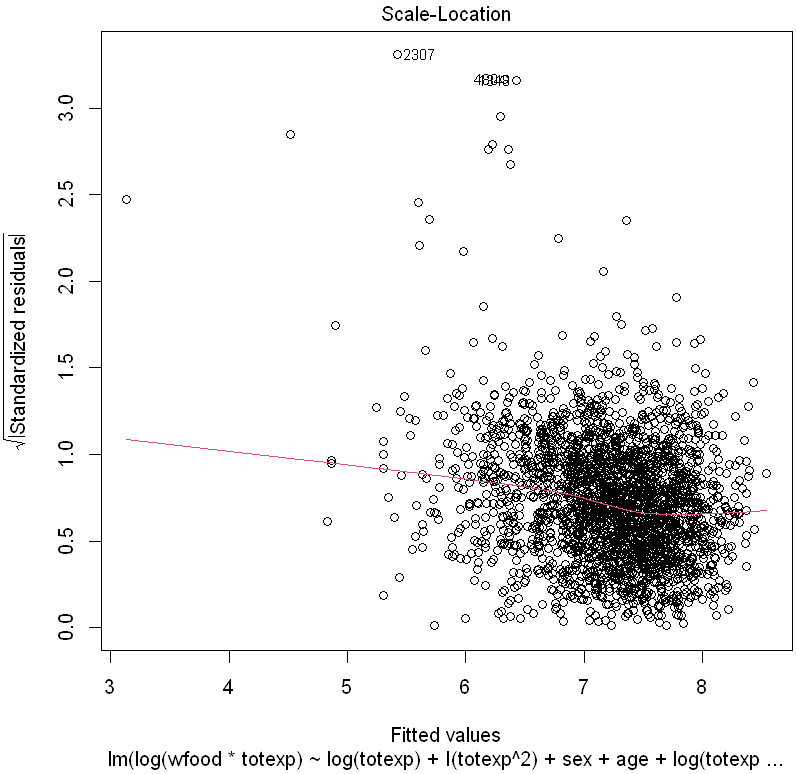
- O AIC é aproximadamente 3175.339, primeiro modelo com AIC positivo.

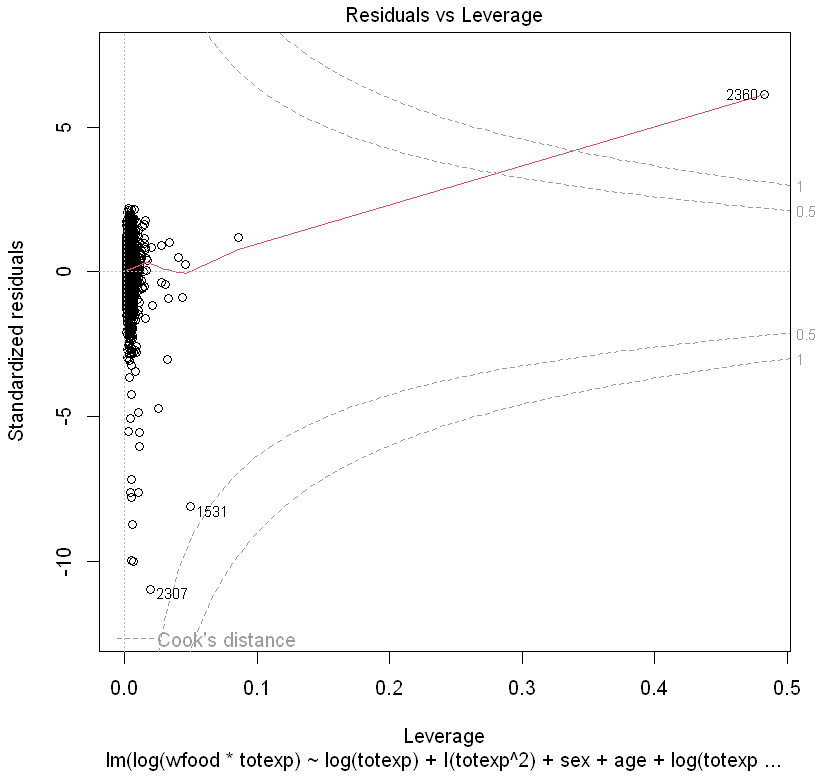
In [31]:
```
# Verificação dos pressupostos dos resíduos

mean(modelo5$residuals) # média nula
bptest(modelo5) # variância constante
bgtest(modelo5) # independência entre resíduos
jarque.bera.test(modelo5$residuals) # normalidade

plot(modelo5)
```

```
-0.00884861781860733
        studentized Breusch-Pagan test

data:  modelo5
BP = 5.8783, df = 8, p-value = 0.6609
        Breusch-Godfrey test for serial correlation of order up to 1

data:  modelo5
LM test = 12.382, df = 1, p-value = 0.0004335
        Jarque Bera Test

data:  modelo5$residuals
X-squared = 14648, df = 2, p-value < 2.2e-16
```




Q-Q Residuals
Standardized residuals
5
0
-5
-10
2307
-3
-2
-1
0
1
2
3
Theoretical Quantiles
lm(log(wfood * totexp) ~ log(totexp) + I(totexp^2) + sex + age + log(totexp ...


Scale-Location
√|Standardized residuals|
2307
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3
4
5
6
7
8
Fitted values
lm(log(wfood * totexp) ~ log(totexp) + I(totexp^2) + sex + age + log(totexp ...



Observações:

- A média dos reísiduos é estatisticamente nula;
- Com o p-value = 0.6609 no teste Breusch-Pagan, não se rejeita a H0 de que os resíduos têm variância constante, logo verificou homocedasticidade -> **Pressuposto da variância constante verificado**;
- Com o p-value = 0.0004335 no teste Breusch-Godfrey, rejeita-se a H0 de que os resíduos são independentes, logo, existe correlação -> **Pressuposto da independência NÃO verificado**;
- Apesar do pressuposto da normalidade dos resíduos não está verificado neste caso, o Teorema do Limite Central garante que, com o aumento do tamanho da amostra, a distribuição dos resíduos tende a se aproximar de uma distribuição normal.

Passaram a ser verificados três pressupostos.

Modelo6

Redefinimos a variável dependente como a despesa alimentar per capita, calculada através da razão (wfood/size).

In [32]:
```
modelo6 <- lm(((wfood) / size) ~  totexp + age + sex+ as.factor(town) + size:age +
(totexp:size)
              + (totexp:as.factor(town)) ,
```

```
              data = budget_food_data, weights = 1/sqrt(totexp))
summary(modelo6)
AIC(modelo6)
vif(modelo6)
```

```
Call:
lm(formula = ((wfood)/size) ~ totexp + age + sex + as.factor(town) +
    size:age + (totexp:size) + (totexp:as.factor(town)), data = budget_food_data,
    weights = 1/sqrt(totexp))

Weighted Residuals:
      Min        1Q    Median        3Q       Max
-0.077261 -0.004410 -0.000504  0.003086  0.094738

Coefficients:
                           Estimate Std. Error t value Pr(>|t|)
(Intercept)               2.040e-01  1.033e-02  19.742  < 2e-16 ***
totexp                   -3.009e-05  1.540e-06 -19.537  < 2e-16 ***
age                       4.189e-03  1.354e-04  30.942  < 2e-16 ***
sex                      -5.944e-02  5.333e-03 -11.145  < 2e-16 ***
as.factor(town)1          4.144e-02  8.344e-03   4.966 7.28e-07 ***
as.factor(town)2          3.118e-03  7.727e-03   0.404   0.6866
as.factor(town)3         -4.995e-03  8.267e-03  -0.604   0.5458
age:size                 -1.034e-03  3.285e-05 -31.483  < 2e-16 ***
totexp:size               4.876e-06  2.745e-07  17.765  < 2e-16 ***
totexp:as.factor(town)1 -4.582e-06  1.855e-06  -2.469   0.0136 *
totexp:as.factor(town)2  9.430e-07  1.580e-06   0.597   0.5507
totexp:as.factor(town)3  1.722e-06  1.443e-06   1.193   0.2329
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.01153 on 2478 degrees of freedom
Multiple R-squared:  0.649,     Adjusted R-squared:  0.6474
F-statistic: 416.5 on 11 and 2478 DF,  p-value: < 2.2e-16
```

-4814.68200247368

```
there are higher-order terms (interactions) in this model
consider setting type = 'predictor'; see ?vif
```

A matrix: 7 × 3 of type dbl

	GVIF	Df	GVIF^(1/(2*Df))
totexp	7.674393	1	2.770269
age	1.465690	1	1.210657
sex	1.359196	1	1.165845
as.factor(town)	19.787845	3	1.644623
age:size	2.400454	1	1.549340
totexp:size	8.177348	1	2.859606
totexp:as.factor(town)	24.930797	3	1.709186

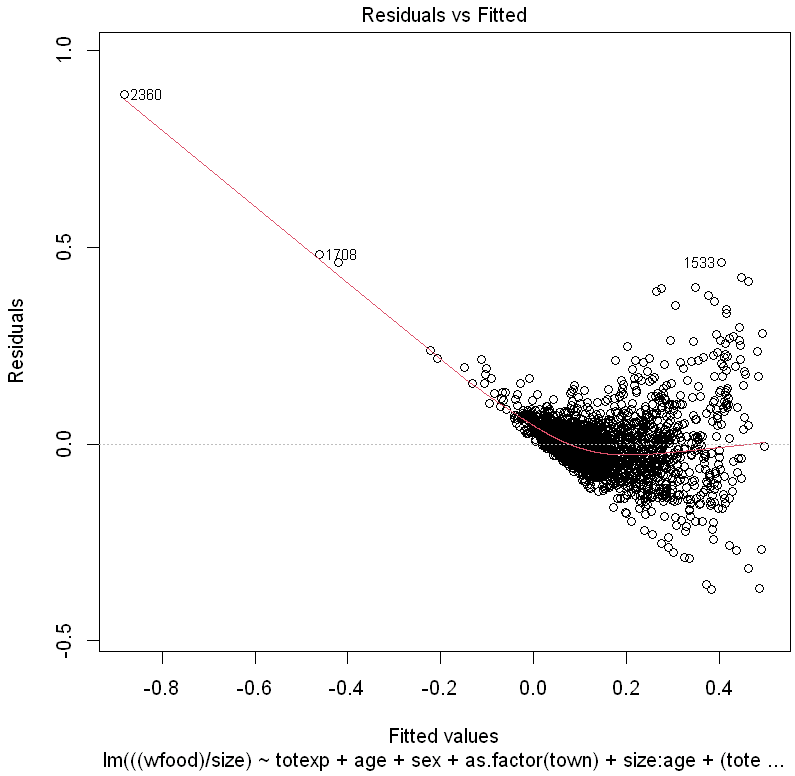
Observações:

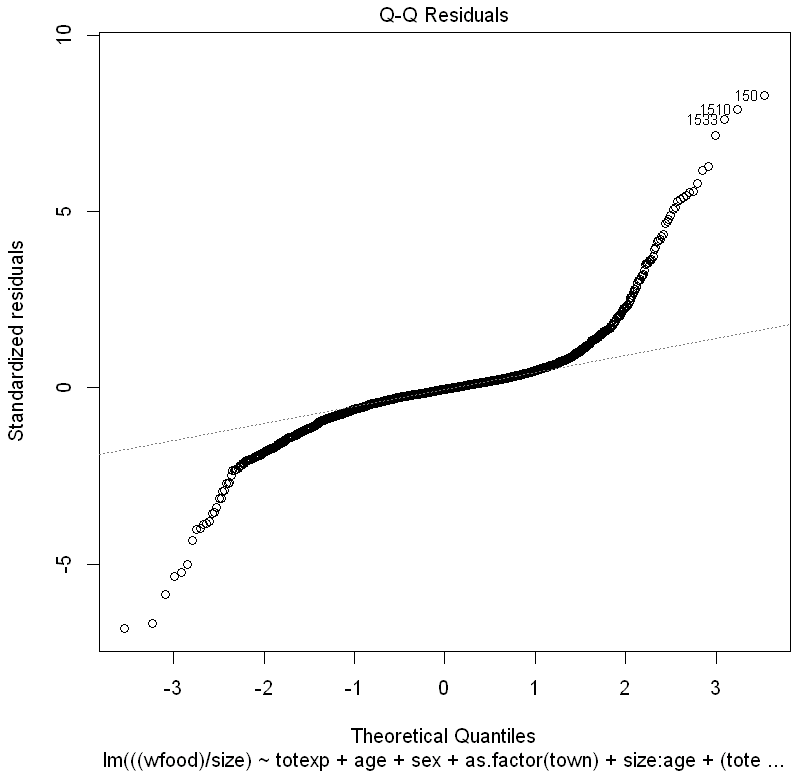
- A maioria das variáveis é relevante;
- O R^2 é 0.649, diminui em relação ao modelo5;
- O AIC é aproximadamente -4814.159, modelo com melhor AIC.

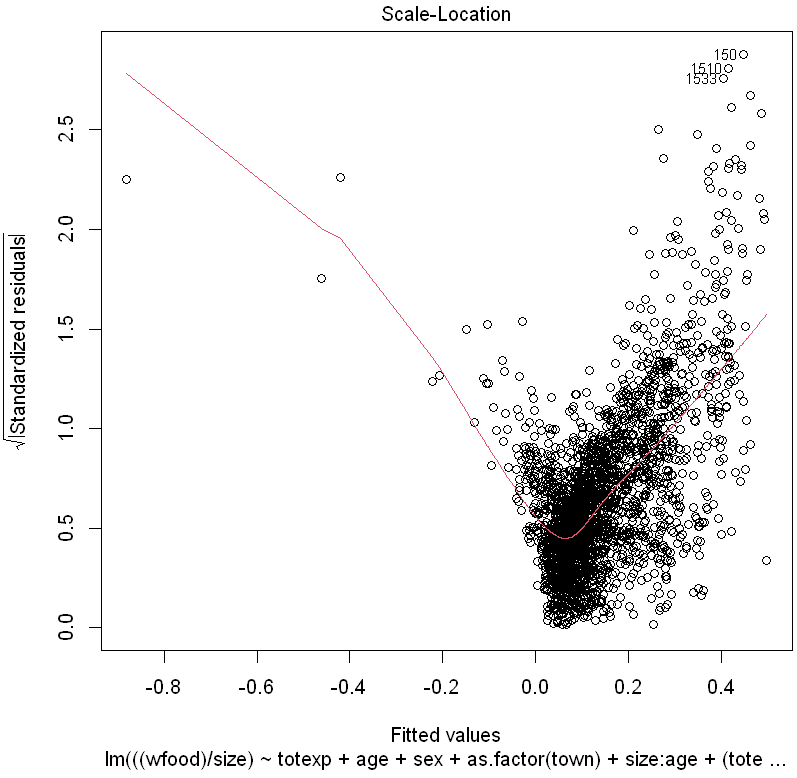
In [33]:
```
# Verificação dos pressupostos dos resíduos
mean(modelo6$residuals) # média nula
bptest(modelo6) # variância constante
bgtest(modelo6) # independência entre resíduos
jarque.bera.test(modelo6$residuals) # normalidade

plot(modelo6)
```

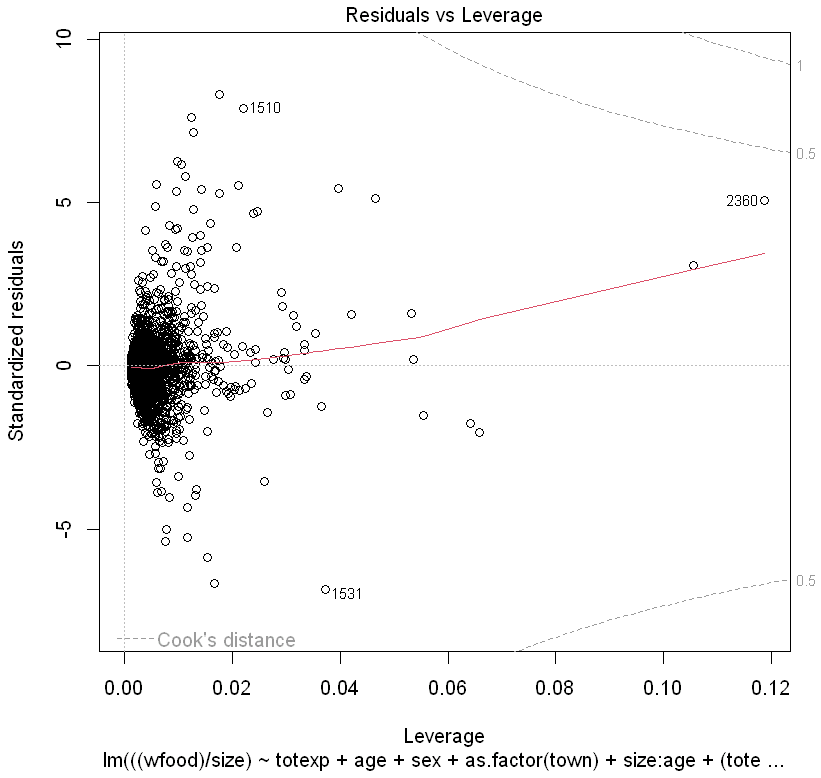
```
-0.00176987046039865
```

```
        studentized Breusch-Pagan test

data:  modelo6
BP = 11.556, df = 11, p-value = 0.3979
        Breusch-Godfrey test for serial correlation of order up to 1

data:  modelo6
LM test = 3.0672, df = 1, p-value = 0.07989
        Jarque Bera Test

data:  modelo6$residuals
X-squared = 20704, df = 2, p-value < 2.2e-16
```




Q-Q Residuals
Standardized residuals
Theoretical Quantiles
lm(((wfood)/size) ~ totexp + age + sex + as.factor(town) + size:age + (tote ...
150
1510
1533


Scale-Location
√|Standardized residuals|
Fitted values
lm(((wfood)/size) ~ totexp + age + sex + as.factor(town) + size:age + (tote ...
150
1510
1533



Observações:

- A média dos reísiduos é estatisticamente nula;
- Com o p-value = 0.3979 no teste Breusch-Pagan, não se rejeita a H0 de que os resíduos têm variância constante, logo verificou homocedasticidade -> **Pressuposto da variância constante verificado**;
- Com o p-value = 0.07989 no teste Breusch-Godfrey, não se rejeita a H0 de que os resíduos são independentes, logo **Pressuposto da independência verificado**;
- Apesar do pressuposto da normalidade dos resíduos não está verificado neste caso, o Teorema do Limite Central garante que, com o aumento do tamanho da amostra, a distribuição dos resíduos tende a se aproximar de uma distribuição normal.

Único modelo que se verificou os três pressupostos

Resumo dos modelos numa tabela

In [34]:
```
# Lista de modelos
modelos <- list(modelo1, modelo2, modelo3, modelo4, modelo5, modelo6)
nomes <- c("Modelo 1", "Modelo 2", "Modelo 3", "Modelo 4", "Modelo 5", "Modelo 6")

# Criar a tabela de resultados
resultados <- data.frame(
  Modelo = nomes,
```

```
  R2 = sapply(modelos, function(m) round(summary(m)$r.squared, 4)),  # Arredondar
R2 para 3 casas decimais
  AIC = sapply(modelos, AIC),
  Média_Nula = sapply(modelos, function(m) ifelse(abs(mean(residuals(m))) < 1e-2,
"SIM", "NÃO")),
  # Testes de pressupostos: SIM se p-valor > 0.05, NÃO se p-valor < 0.05
  Homocedasticidade = sapply(modelos, function(m) ifelse(bptest(m)$p.value > 0.05,
"SIM", "NÃO")),
  Independência = sapply(modelos, function(m) ifelse(bgtest(m)$p.value > 0.05,
"SIM", "NÃO")),
  Normalidade = "SIM" )# Sempre "SIM", com o TLC a normalidade é sempre verificada

resultados
```

A data.frame: 6 × 7

Modelo	R2	AIC	Média_Nula	Homocedasticidade	Independência	Normalidade
<chr>	<dbl>	<dbl>	<chr>	<chr>	<chr>	<chr>
Modelo 1	0.3375	-2948.775	SIM	NÃO	NÃO	SIM
Modelo 2	0.3917	-3157.281	SIM	NÃO	NÃO	SIM
Modelo 3	0.3873	-3139.257	SIM	NÃO	NÃO	SIM
Modelo 4	0.4793	-3821.543	SIM	NÃO	NÃO	SIM
Modelo 5	0.6847	3175.339	SIM	SIM	NÃO	SIM
Modelo 6	0.6490	-4814.682	SIM	SIM	SIM	SIM

Previsão dos modelos no conjunto de teste

In [35]:
```
set.seed(123)

# Dividir a base de dados em treino(80%) e teste(20%)
index = sample.split(budget_food_data, SplitRatio = 0.8)
train = budget_food_data[index,] # Conjunto de treino
test = budget_food_data[!index,] # Conjunto de teste
```

In [36]:
```
# Modelos aplicados no conjunto de treino

train1 <- lm(wfood ~. , data = train)

train2 <- lm(wfood ~ log(totexp) + age + size + sex + as.factor(town) , data =
train)

train3 <- lm(wfood ~ log(totexp)  + sex  + age + log(totexp):size
```

```
              + log(totexp):as.factor(town) , data = train)

# Modelo robusto que calcula os pesos
train4_auxiliar <- rlm(wfood ~ log(totexp)  + sex  + age  + log(totexp):size
                       + log(totexp):as.factor(town) , data = train)

# Extração dos pesos
weights <- train4_auxiliar$w

# Modelo WLS
train4 <- lm(wfood ~ log(totexp)  + sex  + age  + log(totexp):size
              + log(totexp):as.factor(town) , data = train, weights = weights )

train5<- lm(log(wfood*totexp) ~ log(totexp) + I(totexp^2)  +  sex + age +
               log(totexp):size + log(totexp):as.factor(town),
             data = train, weights = 1/sqrt(totexp))

train6 <- lm(((wfood) / size) ~  totexp + age + sex+ as.factor(town) + size:age +
(totexp:size)
              + (totexp:as.factor(town)) ,
              data = train, weights = 1/sqrt(totexp))
```

In [37]:

```
n_obs <- length(test$wfood) # Número de observações no conjunto de teste
actualn_obs <- test$wfood #  Valores reais dos wfood
```

In [38]:

```
## ============= modelo1 ===========
pred_wfood1_t = predict(train1, newdata = test)

# Cálculo do MAPE
MAPE1t = (1/n_obs) * sum(abs((actualn_obs - pred_wfood1_t) / actualn_obs)) * 100

# Cálculo do RMSE
RMSE1t = sqrt(sum((pred_wfood1_t - actualn_obs)^2) / n_obs)


## ================ modelo2 ===============
pred_wfood2_t = predict(train2, newdata = test)

# Cálculo do MAPE
MAPE2t = (1/n_obs) * sum(abs((actualn_obs - pred_wfood2_t) / actualn_obs)) * 100

# Cálculo do RMSE
RMSE2t = sqrt(sum((pred_wfood2_t - actualn_obs)^2) / n_obs)


## ================ modelo3 ==============
pred_wfood3_t = predict(train3, newdata = test)

# Cálculo do MAPE
MAPE3t = (1/n_obs) * sum(abs((actualn_obs - pred_wfood3_t) / actualn_obs)) * 100

# Cálculo do RMSE
RMSE3t = sqrt(sum((pred_wfood3_t - actualn_obs)^2) / n_obs)
```

```
## ========== modelo4 ============
pred_wfood4_t = predict(train4, newdata = test)

# Cálculo do MAPE
MAPE4t = (1/n_obs) * sum(abs((actualn_obs - pred_wfood4_t) / actualn_obs)) * 100

# Cálculo do RMSE
RMSE4t = sqrt(sum((pred_wfood4_t - actualn_obs)^2) / n_obs)


## ========== modelo5 ============
pred_log_wfood_totexp5_train = predict(train5,newdata = test)
pred_wfood_totexp5_t = exp(pred_log_wfood_totexp5_train)

pred_wfood5_t = pred_wfood_totexp5_t / test$totexp

# Cálculo do MAPE
MAPE5t = (1/n_obs) * sum(abs((actualn_obs - pred_wfood5_t) / actualn_obs)) * 100

# Cálculo do RMSE
RMSE5t = sqrt(sum((pred_wfood5_t - actualn_obs)^2) / n_obs)


## ============ modelo6 ==============
pred_wfood_size6_t = predict(train6,newdata = test)

pred_wfood6_t = pred_wfood_size6_t * test$size

# Cálculo do MAPE
MAPE6t = (1/n_obs) * sum(abs((actualn_obs - pred_wfood6_t) / actualn_obs)) * 100

# Cálculo do RMSE
RMSE6t = sqrt(sum((pred_wfood6_t - actualn_obs)^2) / n_obs)
```

Representação dos resultados dos modelos no teste

In [39]:

```
# Listar os modelos
modelos <- c("Modelo 1", "Modelo 2", "Modelo 3", "Modelo 4", "Modelo 5", "Modelo
6")

# MAPE de cada modelo
MAPEt <- c(MAPE1t, MAPE2t, MAPE3t, MAPE4t, MAPE5t, MAPE6t)

# RMSE de cada modelo
RMSEt <- c(RMSE1t, RMSE2t, RMSE3t, RMSE4t, RMSE5t, RMSE6t)

# Tabela de resultados
tab_eval_t <- cbind(modelos,"MAPE Out-Sample(%)" = round(MAPEt,3), "RMSE" =
round(RMSEt,3))
tab_eval_t
```

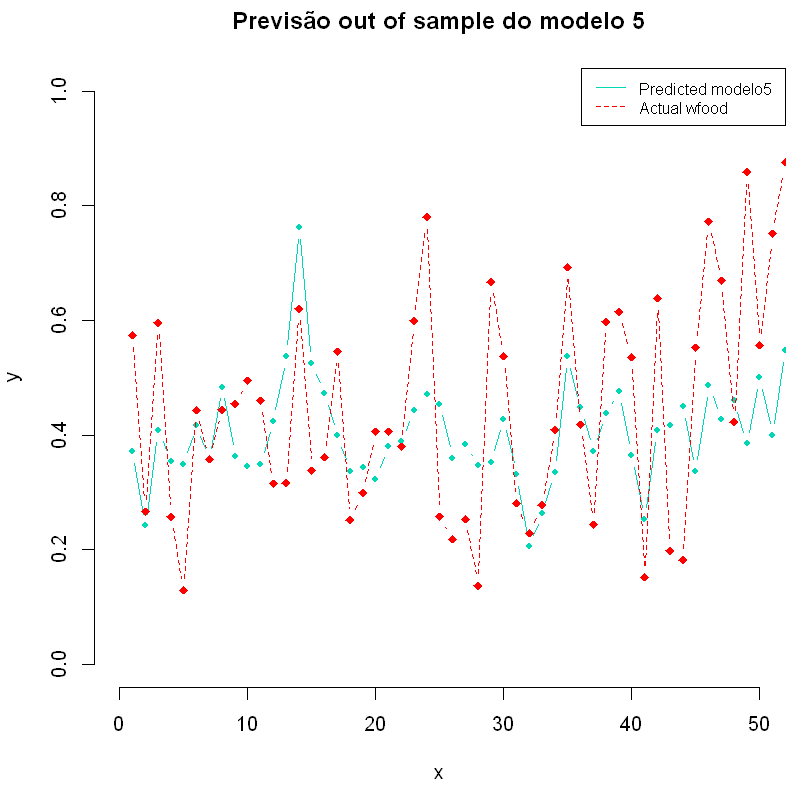
A matrix: 6 × 3 of type chr

modelos	MAPE Out-Sample(%)	RMSE
Modelo 1	40.119	0.132
Modelo 2	38.161	0.13
Modelo 3	38.386	0.13
Modelo 4	37.571	0.13
Modelo 5	36.547	0.135
Modelo 6	54.412	0.246

Previsão do modelo 5 sobre o conjunto de teste (50 observações)

Através da comparação dos valores de MAPE (Mean Absolute Percentage Error) e RMSE (Root Mean Square Error) de cada modelo, o Modelo 5 foi selecionado como o mais adequado para prosseguir com as previsões, por apresentar os menores erros e, consequentemente, maior precisão.

In [40]:
```
# Previsões do modelo5 para wfood - verde
plot(pred_wfood5_t, type = "b", frame = FALSE, pch = 20, col = "#00d9b5", xlab =
"x", ylab = "y",
     xlim=c(0,50), ylim=c(0,1), main="Previsão out of sample do modelo 5")

# Valores reais de wfood - vermelha
lines(actualn_obs, pch = 18, col = "red", type = "b", lty = 2, xlim=c(0,50),
ylim=c(0,1))

# Adicionar legendas às linhas do gráfico
legend("topright", legend=c("Predicted modelo5", "Actual wfood"), col=c("#00d9b5",
"red"), lty = 1:2, cex=0.8)
```


Previsão out of sample do modelo 5
Predicted modelo5
Actual wfood
y
x
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
50