Tipologia i cicle de vida de les dades - Pràctica 2

Autor: Isabel Barrera Benavent i Maria Font Sánchez

Desembre 2021

# Descripció del dataset. Perquè és important i quina pregunta/problema pretén respondre?

El dataset que hem decidit fer servir per aquesta pràctica és el següent:

<https://www.kaggle.com/imakash3011/customer-personality-analysis>

És un conjunt de dades obtingut de la web Kaggle i construït per 29 atributs, un dels quals és l’identificador únic del registre, 25 atributs numèrics enters i 3 atributs de tipus text.

Es tracta d’una enquesta feta a 2.240 clients d’una empresa per a poder valorar la seva personalitat a l’hora de comprar els seus productes i amb la finalitat de comprendre i analitzar els gustos i necessitats que tenen segons el tipus de clients que siguin.

D’aquesta manera l’empresa pot prendre decisions respecte als productes que ven o fer campanyes de màrqueting més específiques segons el tipus de clients i producte.

Els atributs que conté el dataset són els següents:

* **ID**: Identificador únic del client.
* **Year\_Birth**\*: Any de naixement del client.
* **Education**: Nivell d’educació del client.
* **Marital\_Status**: Estat marital del client.
* **Income**: Ingrés familiar del client.
* **Kidhome**: Número de fills amb qui viu el client.
* **Teenhome**: Número d’adolescents amb qui viu el client.
* **Dt\_Customer**: Data en que s’inscriu com client de l’empresa.
* **Recency**: Número de dies que han passat des de l’última compra del client.
* **MntWines**: Quantitat gastada en vi durant els últims dos anys per el client.
* **MntFruits**: Quantitat gastada en vi durant els últims dos anys per el client.
* **MntMeatProducts**: Quantitat gastada en carn durant els últims dos anys per el client.
* **MntFishProducts**: Quantitat gastada en peix durant els últims dos anys per el client.
* **MntSweetProducts**: Quantitat gastada en dolços durant els últims dos anys per el client.
* **MntGoldProds**: Quantitat gastada en or durant els últims dos anys per el client.
* **NumDealsPurchases**: Número de compres realitzades amb descomptes per el client.
* **NumWebPurchases**: Número de compres realitzades des de la web de l’empresa per el client.
* **NumCatalogPurchases**: Número de compres realitzades des del catàleg de l’empresa per el client.
* **NumStorePurchases**: Número de compres realitzades directament en una tenda de l’empresa per el client.
* **NumWebVisitsMonth**: Número de visites realitzades durant l’últim mes a la web de l’empresa del client.
* **AcceptedCmp3**: 1 si el client va acceptar l’oferta en la tercera campanya, 0 en cas contrari
* **AcceptedCmp4**: 1 si el client va acceptar l’oferta en la quarta campanya, 0 en cas contrari.
* **AcceptedCmp5**: 1 si el client va acceptar l’oferta en la cinquena campanya, 0 en cas contrari.
* **AcceptedCmp1**: 1 si el client va acceptar l’oferta en la primera campanya, 0 en cas contrari.
* **AcceptedCmp2**: 1 si el client va acceptar l’oferta en la segons campanya, 0 en cas contrari.
* **Complain**: 1 si hi ha hagut alguna queixa del client en els últims dos anys. 0 en cas contrari.
* **Z\_CostContact**: Cost Contacte. Té valor 3 per defecte.
* **Z\_Revenue**: Ingressos del client: Té valor 11 per defecte.
* **Response**: 1 si el client va acceptar l’oferta en la última campanya, 0 en cas contrari.

A partir del conjunt de dades d’aquest dataset, es pretén estudiar com influeix el perfil del client amb la quantitat gastada en les compres de productes d’aliments. Per això, s’aplicaran mètodes de regressió per a veure la relació entre les diferents variables, i mètodes de classificació per a crear el perfil del client. Aquest estudi és important per a les empreses de ventes de productes (tan online com físiques) per a poder identificar el tipus de client que compra cada producte i crear així campanyes de màrqueting específiques.

# Integració i selecció de les dades d’interès a analitzar

Per tal de poder realitzar la selecció de les dades d’interès a analitzar, el primer pas és fer la lectura del fitxer CSV anomenat ‘marketing\_campaign.csv’:

data\_clients<-read.csv("../data/marketing\_campaign.csv", header=T, sep="\t")

D’aquestes 29 variables, ens interessa quedar-nos amb les següents:

* ID: identificador únic
* **Perfil del client**:
* Year\_Birth
* Education
* Marital\_status
* Income
* Kidhome
* Teenhome
* Dt\_Customer
* **Gasto per productes alimentaris**:
* MntFruits
* MntMeatProducts
* MntFishProducts
* MntSweetProducts
* NumWebPurchases
* NumCatalogPurchases
* NumStorePurchases

La resta de variables, degut a que no es faran servir, s’eliminen del conjunt de dades:

data\_clients <- data\_clients[, c('ID', 'Year\_Birth', 'Education', 'Marital\_Status', 'Income', 'Kidhome', 'Teenhome',   
 'Dt\_Customer', 'MntFruits', 'MntMeatProducts', 'MntFishProducts', 'MntSweetProducts',  
 'NumWebPurchases', 'NumCatalogPurchases', 'NumStorePurchases')]

# Neteja de dades

Abans de començar amb la neteja de dades, cal mirar el tipus de dades que R ha assignat a cada columna, i també amb la funció ‘summary’ es pot veure els valors màxims, mínims i les mitjanes de les variables numèriques.

summary(data\_clients)

## ID Year\_Birth Education Marital\_Status   
## Min. : 0 Min. :1893 Length:2240 Length:2240   
## 1st Qu.: 2828 1st Qu.:1959 Class :character Class :character   
## Median : 5458 Median :1970 Mode :character Mode :character   
## Mean : 5592 Mean :1969   
## 3rd Qu.: 8428 3rd Qu.:1977   
## Max. :11191 Max. :1996   
##   
## Income Kidhome Teenhome Dt\_Customer   
## Min. : 1730 Min. :0.0000 Min. :0.0000 Length:2240   
## 1st Qu.: 35303 1st Qu.:0.0000 1st Qu.:0.0000 Class :character   
## Median : 51382 Median :0.0000 Median :0.0000 Mode :character   
## Mean : 52247 Mean :0.4442 Mean :0.5062   
## 3rd Qu.: 68522 3rd Qu.:1.0000 3rd Qu.:1.0000   
## Max. :666666 Max. :2.0000 Max. :2.0000   
## NA's :24   
## MntFruits MntMeatProducts MntFishProducts MntSweetProducts  
## Min. : 0.0 Min. : 0.0 Min. : 0.00 Min. : 0.00   
## 1st Qu.: 1.0 1st Qu.: 16.0 1st Qu.: 3.00 1st Qu.: 1.00   
## Median : 8.0 Median : 67.0 Median : 12.00 Median : 8.00   
## Mean : 26.3 Mean : 166.9 Mean : 37.53 Mean : 27.06   
## 3rd Qu.: 33.0 3rd Qu.: 232.0 3rd Qu.: 50.00 3rd Qu.: 33.00   
## Max. :199.0 Max. :1725.0 Max. :259.00 Max. :263.00   
##   
## NumWebPurchases NumCatalogPurchases NumStorePurchases  
## Min. : 0.000 Min. : 0.000 Min. : 0.00   
## 1st Qu.: 2.000 1st Qu.: 0.000 1st Qu.: 3.00   
## Median : 4.000 Median : 2.000 Median : 5.00   
## Mean : 4.085 Mean : 2.662 Mean : 5.79   
## 3rd Qu.: 6.000 3rd Qu.: 4.000 3rd Qu.: 8.00   
## Max. :27.000 Max. :28.000 Max. :13.00   
##

Es pot observar que les variables ‘Education’, ‘Marital Status’ i ‘Dt Customer’ són les úniques expressades com a text. La resta de variables són numèriques.

Primer, es validarà que les variables de text ‘Education’ i ‘Marital Status’ es poden categoritzar:

table(data\_clients$Education)

##   
## 2n Cycle Basic Graduation Master PhD   
## 203 54 1127 370 486

table(data\_clients$Marital\_Status)

##   
## Absurd Alone Divorced Married Single Together Widow YOLO   
## 2 3 232 864 480 580 77 2

Totes dues variables es poden categoritzar fent ús de la funció ‘as.factor’:

data\_clients$Education <- as.factor(data\_clients$Education)  
data\_clients$Marital\_Status <- as.factor(data\_clients$Marital\_Status)

## Zeros o elements buits

**Les dades contenen zeros o elements buits? Com gestionaries aquests casos?**

Les variables numèriques utilitzen el valor 0 per indicar que no es té informació. I les de tipus text ho indiquen mitjançant els elements buits. El primer pas per a fer aquesta comprovació, es validar si existeixen valors buits en les variables seleccionades:

colSums(is.na(data\_clients))

## ID Year\_Birth Education Marital\_Status   
## 0 0 0 0   
## Income Kidhome Teenhome Dt\_Customer   
## 24 0 0 0   
## MntFruits MntMeatProducts MntFishProducts MntSweetProducts   
## 0 0 0 0   
## NumWebPurchases NumCatalogPurchases NumStorePurchases   
## 0 0 0

En aquest cas, la variable ‘Income’ en té 24. Per tal de gestionar aquests valors, hi ha diverses tècniques que es poden aplicar:

* Eliminar aquests registres.
* Aplicar la mitjana dels valors més propers.

Degut a que es tracta d’1% dels registres totals, es preferible eliminar-los ja que l’impacte i la perduda de dades és mínima:

data\_clients <- na.omit(data\_clients)  
colSums(is.na(data\_clients))

## ID Year\_Birth Education Marital\_Status   
## 0 0 0 0   
## Income Kidhome Teenhome Dt\_Customer   
## 0 0 0 0   
## MntFruits MntMeatProducts MntFishProducts MntSweetProducts   
## 0 0 0 0   
## NumWebPurchases NumCatalogPurchases NumStorePurchases   
## 0 0 0

## Identificació i tractament de valors extrems

El següent pas és identificar els valors extrems, que són les dades que es troben allunyades de la distribució normal.

Utilitzarem la funció boxplots.stats() de R per anar revisant variable a variable si trobem registres amb valors extrems que es puguin considerar incorrectes o que ens pugui implicar una desviació important.

VARIABLE **YEAR\_BIRTH**

boxplot.stats(data\_clients$Year\_Birth)$out

## [1] 1900 1893 1899

summary(data\_clients$Year\_Birth)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 1893 1959 1970 1969 1977 1996

## En aquest cas considerem que aquests 3 valors són incorrectes, ja que implicaria que actualment aquests clients tenen més de 130 anys.  
## Decidim canviar-los per el valor mitjà.  
median\_year\_birth <- median(data\_clients$Year\_Birth)  
  
data\_clients$Year\_Birth[(data\_clients$Year\_Birth==1893)] <- median\_year\_birth  
data\_clients$Year\_Birth[(data\_clients$Year\_Birth==1900)] <- median\_year\_birth  
data\_clients$Year\_Birth[(data\_clients$Year\_Birth==1899)] <- median\_year\_birth

VARIABLES **EDUCATION**, **MARITAL STATUS**, **KIDHOME**, **TEENHOME** I **NUMSTORESPURCHASES**:

boxplot.stats(data\_clients$Education)$out

## factor(0)  
## Levels: 2n Cycle Basic Graduation Master PhD

boxplot.stats(data\_clients$Marital\_Status)$out

## factor(0)  
## Levels: Absurd Alone Divorced Married Single Together Widow YOLO

boxplot.stats(data\_clients$Kidhome)$out

## integer(0)

boxplot.stats(data\_clients$Teenhome)$out

## integer(0)

boxplot.stats(data\_clients$NumStorePurchases)$out

## integer(0)

## No sembla que hi hagin valors extrems.   
## Revisem els seus valor amb la funció table  
  
table(data\_clients$Education)

##   
## 2n Cycle Basic Graduation Master PhD   
## 200 54 1116 365 481

table(data\_clients$Marital\_Status)

##   
## Absurd Alone Divorced Married Single Together Widow YOLO   
## 2 3 232 857 471 573 76 2

table(data\_clients$Kidhome)

##   
## 0 1 2   
## 1283 887 46

table(data\_clients$Teenhome)

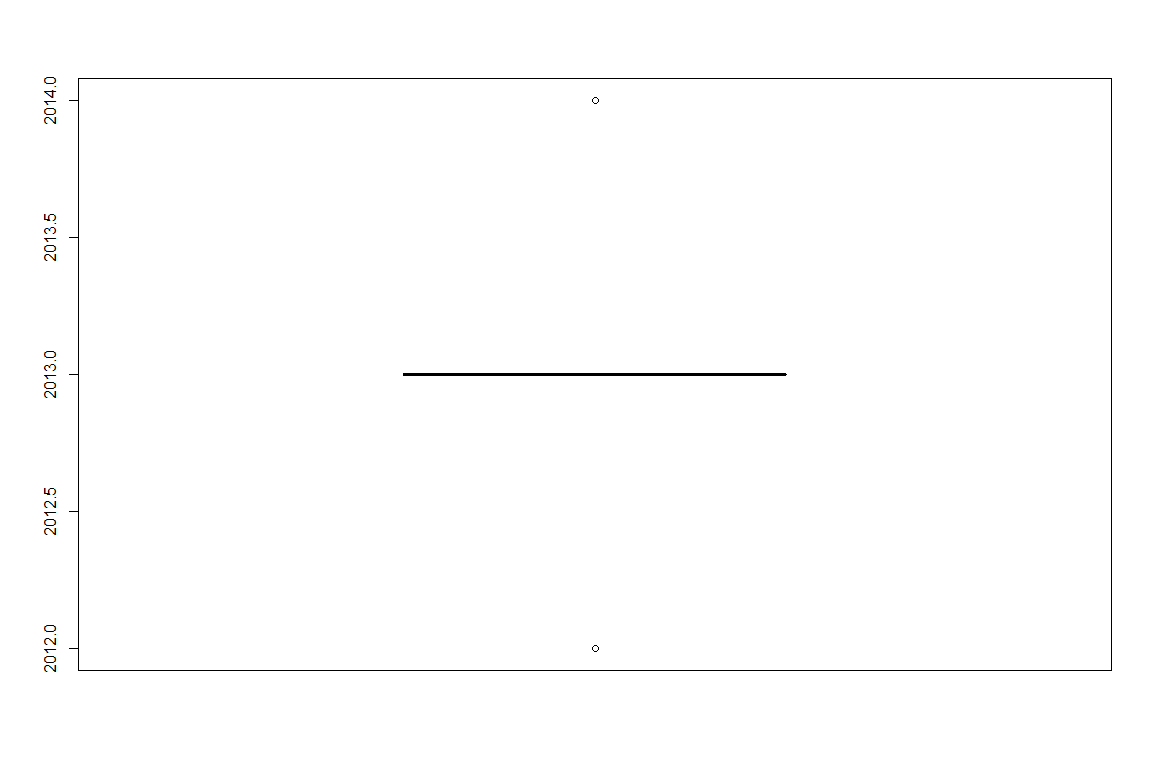
##   
## 0 1 2   
## 1147 1018 51

table(data\_clients$NumStorePurchases)

##   
## 0 1 2 3 4 5 6 7 8 9 10 11 12 13   
## 14 6 220 484 319 211 177 141 147 106 124 80 104 83

VARIABLE **DT\_CUSTOMER**:

# Revisem els anys de les dates de clients.  
  
boxplot(as.numeric(substring(data\_clients$Dt\_Customer,7,10)))



table(as.numeric(substring(data\_clients$Dt\_Customer,7,10)))

##   
## 2012 2013 2014   
## 490 1173 553

## Tant la seva representació en boxplot com en taula és correcta.

VARIABLE **NUMWEBPURCHASES**:

boxplot.stats(data\_clients$NumWebPurchases)$out

## [1] 23 27 25

summary(data\_clients$NumWebPurchases)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.000 2.000 4.000 4.085 6.000 27.000

## Els valors extrems són valors que es poden donar. Els donem per bons.

VARIBALE **NUMCATALOGPURCHAES**:

boxplot.stats(data\_clients$NumCatalogPurchases)$out

## [1] 28 11 22 11 11 11 11 11 28 11 11 11 11 11 11 11 11 28 11 11 11 11 11

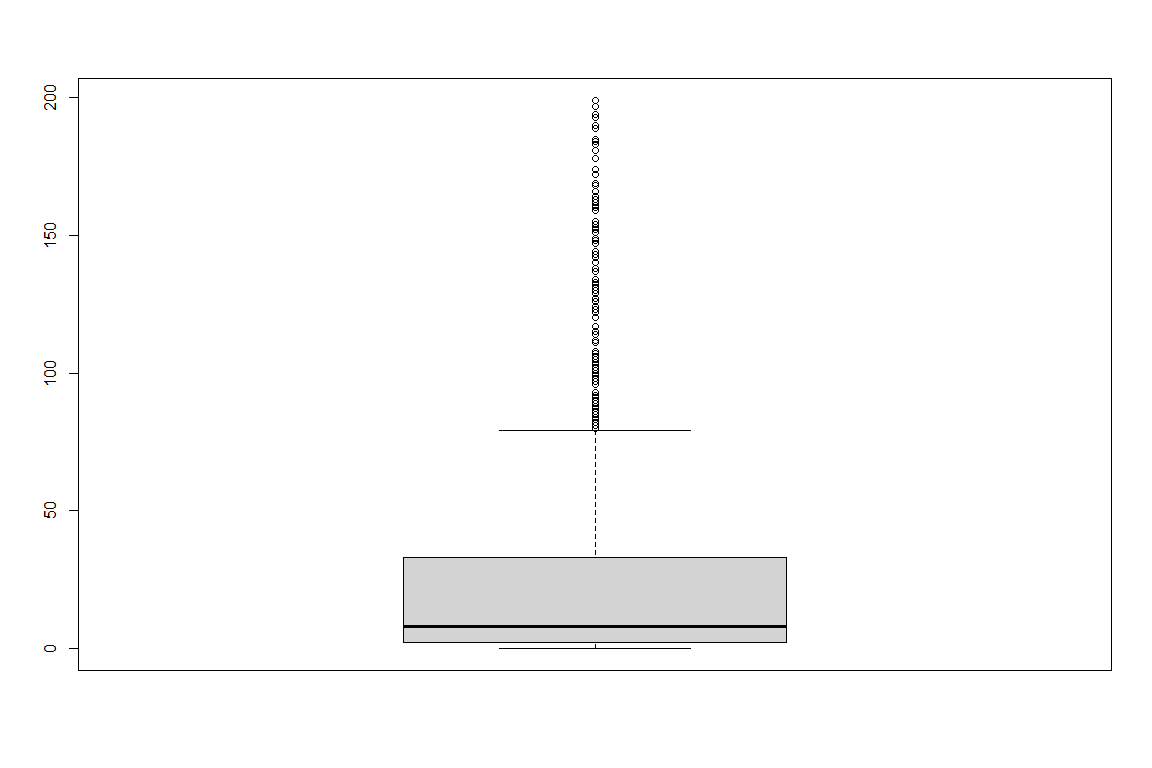
summary(data\_clients$NumCatalogPurchases)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.000 0.000 2.000 2.671 4.000 28.000

## Els valors extrems són valors que es poden donar. Els donem per bons.

VARIABLE **MNTFRUITS**:

## Al tenir tants valors diferents, representem el seu boxplot:  
  
boxplot(data\_clients$MntFruits)



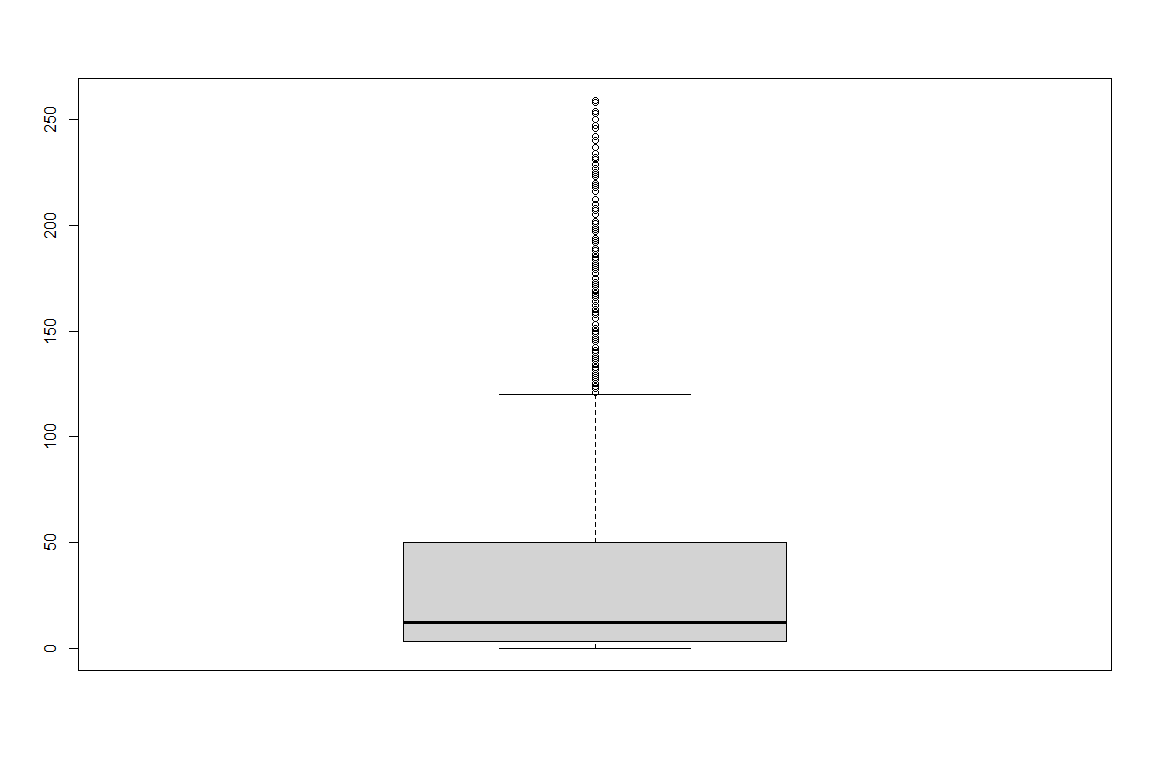
summary(data\_clients$MntFruits)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.00 2.00 8.00 26.36 33.00 199.00

## Veiem que hi han valors que estan fora de l’esperat però no estan allunyats entre ells. A més, són valors possibles.  
## Ho donem per bo.

VARIABLE **MNTFISHPRODUCTS**:

## Al tenir tants valors diferents, representem el seu boxplot  
boxplot(data\_clients$MntFishProducts)



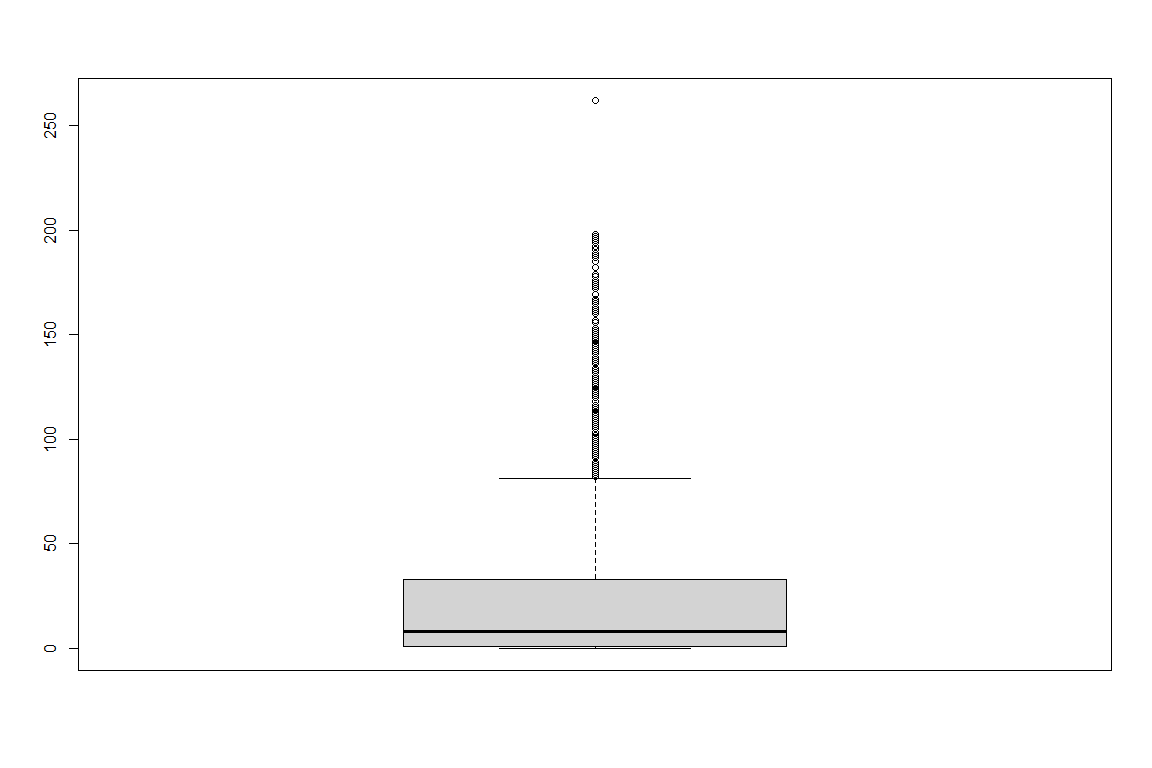
summary(data\_clients$MntFishProducts)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.00 3.00 12.00 37.64 50.00 259.00

## Veiem que hi han valors que estan fora de l’esperat però no estan allunyats entre ells. A més, són valors possibles.  
## Ho donem per bo.

VARIABLE **MNTSWEETPRODUCTS**:

boxplot(data\_clients$MntSweetProducts)



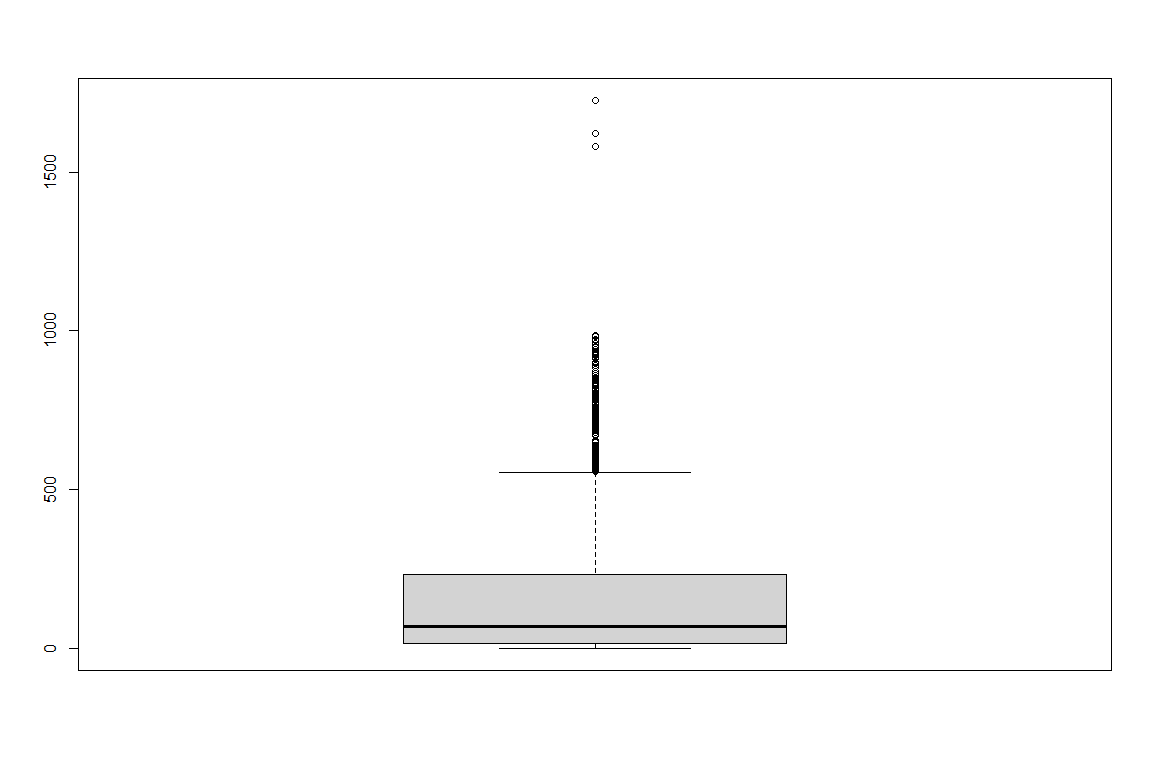
summary(data\_clients$MntSweetProducts)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.00 1.00 8.00 27.03 33.00 262.00

## Veiem que hi han un valor que estan molt fora de ho esperat.  
## Al ser només un el canviem per la mitjana.  
  
data\_clients$MntSweetProducts[(data\_clients$MntSweetProducts==262)] <- median(data\_clients$MntSweetProducts)

VARIABLE **MNTMEATPRODUCTS**;

## Al tenir tants valors diferents, representem el seu boxplot  
boxplot(data\_clients$MntMeatProducts)



summary(data\_clients$MntMeatProducts)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.0 16.0 68.0 167.0 232.2 1725.0

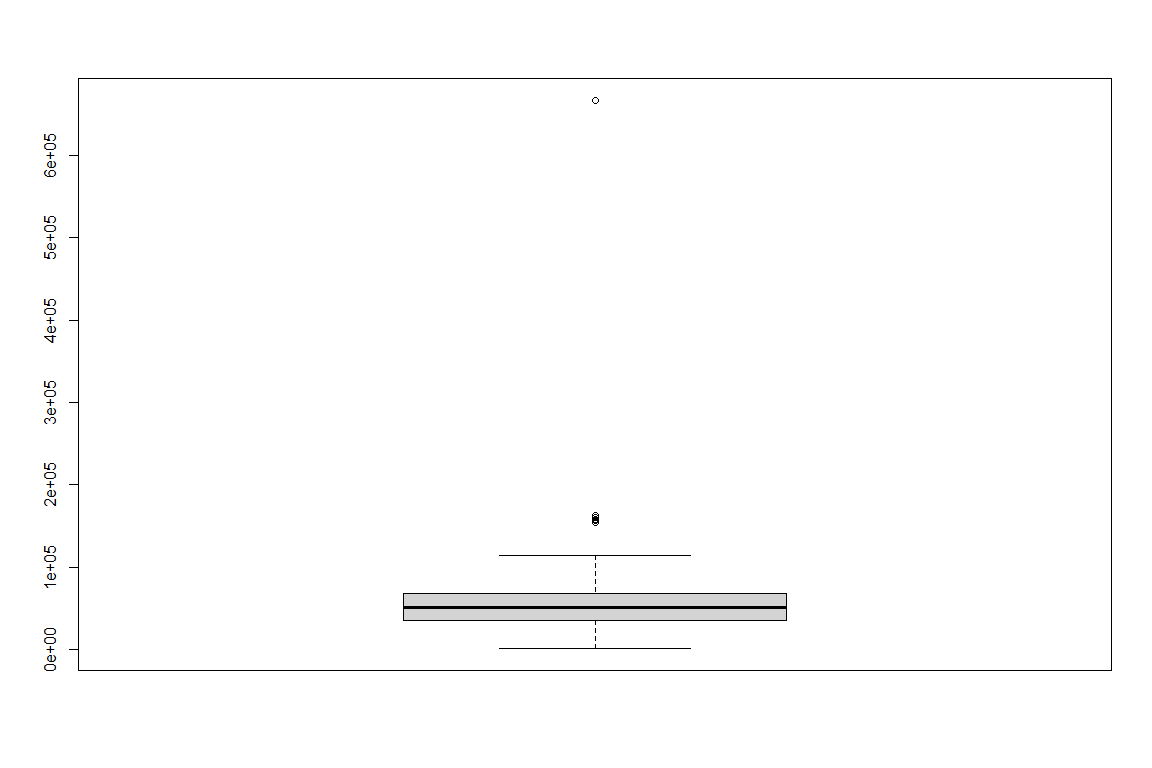
## Sembla que hi han uns 4 valors superiors dels 1500 i diferenciats dels altres.  
table(data\_clients$MntMeatProducts[(data\_clients$MntMeatProducts>1500)] )

##   
## 1582 1622 1725   
## 1 1 2

## Els canviem per la seva mitjana:  
data\_clients$MntMeatProducts[(data\_clients$MntMeatProducts>1500)] <- median(data\_clients$MntMeatProducts)

VARIABLE **INCOME**:

## Al tenir tants valors diferents, representem el seu boxplot  
boxplot(data\_clients$Income)



summary(data\_clients$Income)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 1730 35303 51382 52247 68522 666666

## Hi ha un valor molt superior al altres.  
table(data\_clients$Income[(data\_clients$Income>300000)] )

##   
## 666666   
## 1

## Els canviem per la seva mitjana:  
data\_clients$Income[(data\_clients$Income>300000)] <- median(data\_clients$Income)

## Anàlisi de les dades

### Selecció dels grups de dades que es volen analitzar/comparar (planificació dels anàlisis a aplicar)

La selecció dels grups de dades ha de permetre realitzar les proves estadístiques en funció dels grups a comparar o analitzar.

Per això, es seleccionaran les dades en funció de les variables ‘Education’ i ‘Marital Status’, aquestes són les que s’han categoritzat anteriorment.

#Agrupació per estudis:  
  
data\_clients.basic <- data\_clients[data\_clients$Education == "Basic",]  
data\_clients.graduation <- data\_clients[data\_clients$Education == "Graduation",]  
data\_clients.cycle <- data\_clients[data\_clients$Education == "2n Cycle",]  
data\_clients.master <- data\_clients[data\_clients$Education == "Master",]  
data\_clients.phd <- data\_clients[data\_clients$Education == "PhD",]  
  
#Agrupació per estat civil:  
data\_clients.divorced <- data\_clients[data\_clients$Marital\_Status == "Divorced",]  
data\_clients.married <- data\_clients[data\_clients$Marital\_Status == "Married",]  
data\_clients.single <- data\_clients[data\_clients$Marital\_Status == "Single" || data\_clients$Education == "Alone",]  
data\_clients.together <- data\_clients[data\_clients$Marital\_Status == "Together",]  
data\_clients.widow <- data\_clients[data\_clients$Marital\_Status == "Widow",]

### Comprovació de la normalitat i homogeneïtat de la variància

Per a poder realitzar les proves estadístiques, cal comprovar la normalitat i la homogeneïtat de la variància.

En aquest cas utilitzarem el test de Shaprio-Wilk, on diu que si el p-valor és més petit que α=0,05, es conclou que les dades no estan seguint una distribució normal. Per contra, si p-valor es major que α=0,05, les dades sí que estan seguint una distribució normal.

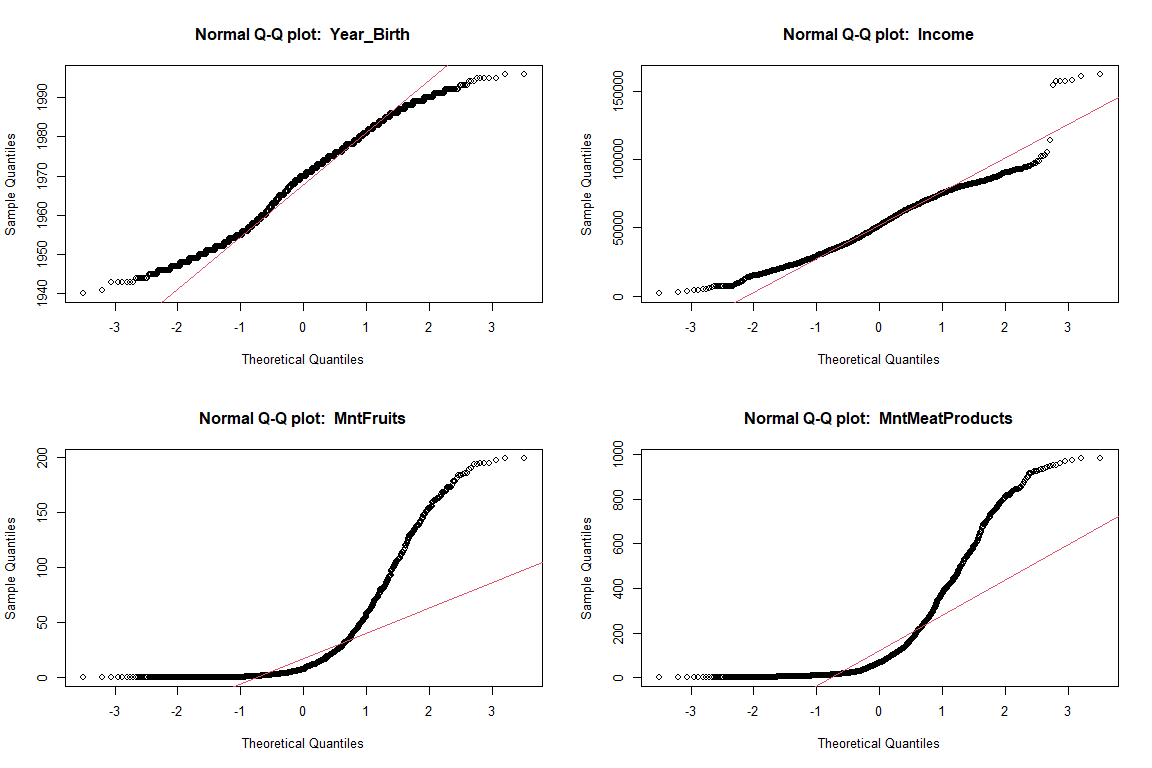
Les variables amb les que es mesurarà la normalitat són les del tipus numèric, i que no siguin la variable ID (corresponen a l’identificador), ni Kidhome ni Teenhome ja que corresponen a valors booleans del tipus 0 i 1.

options(scipen = 100)  
par(mfrow=c(2,2))  
  
p\_value <- 0.05  
  
columnes <- colnames(data\_clients)  
  
colnames\_not\_check <- c("ID", "Kidhome", "Teenhome")  
  
for (i in 1:ncol(data\_clients)) {  
 col <- data\_clients[,i]  
 if(is.numeric(col) && !is.element(colnames(data\_clients)[i], colnames\_not\_check)){  
 qqnorm(col, main = paste("Normal Q-Q plot: ", colnames(data\_clients)[i]));  
 qqline(col, col = 2)  
   
 normality\_pvalue <- shapiro.test(col)$p.value  
   
 if(normality\_pvalue < p\_value){  
 cat("Distribució NO normalitzada de la columna: ", colnames(data\_clients)[i])  
 }else{  
 cat("Distribució normal de la columna: ", colnames(data\_clients)[i])  
 }  
 cat("\n")  
 }  
}

## Distribució NO normalitzada de la columna: Year\_Birth

## Distribució NO normalitzada de la columna: Income

## Distribució NO normalitzada de la columna: MntFruits

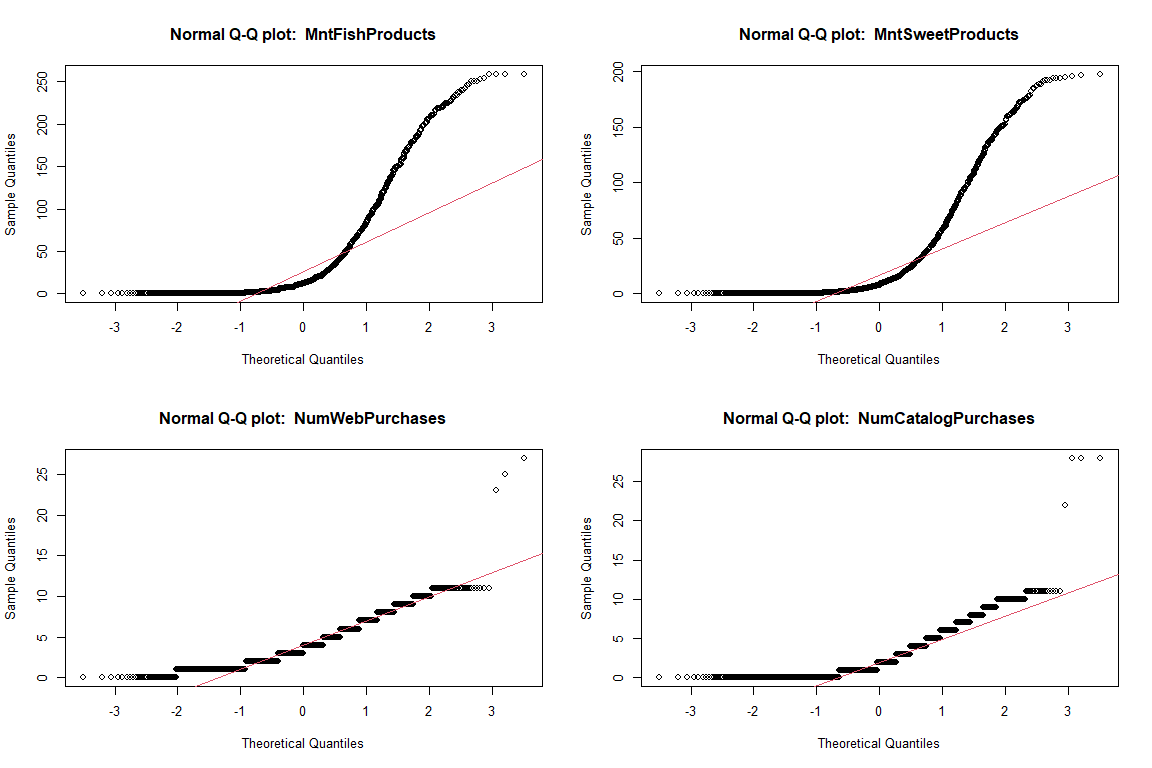


## Distribució NO normalitzada de la columna: MntMeatProducts

## Distribució NO normalitzada de la columna: MntFishProducts

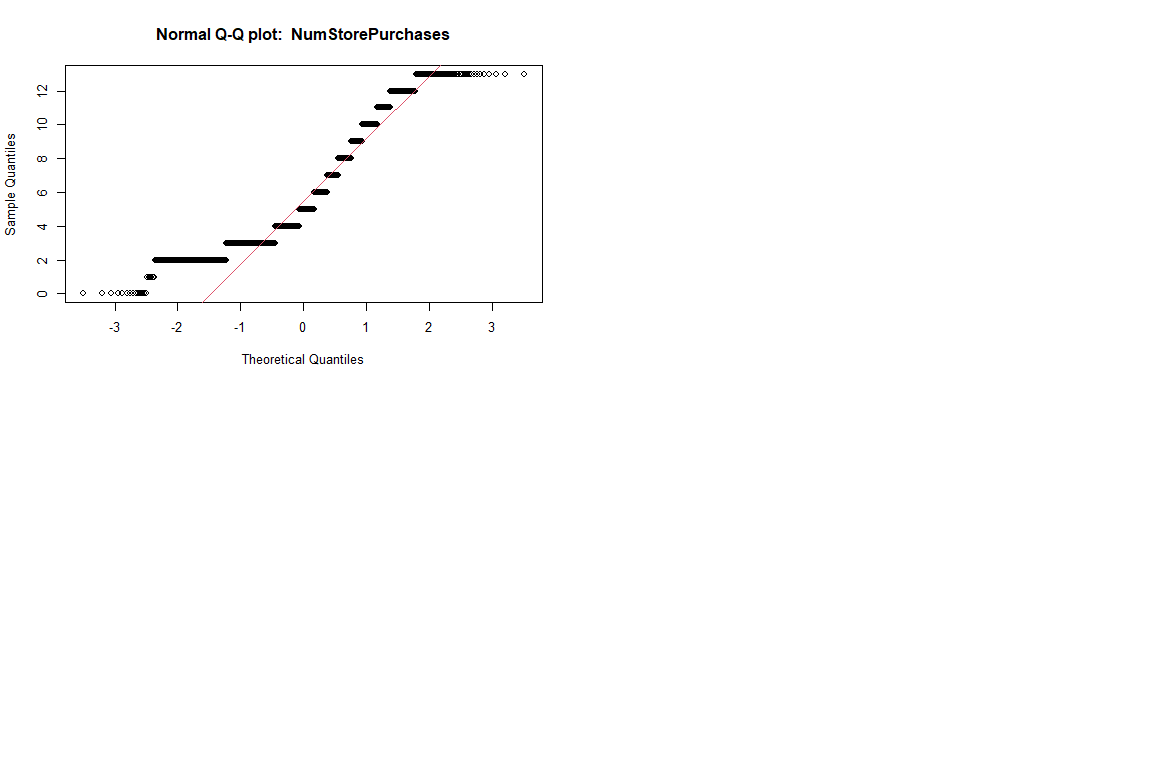
## Distribució NO normalitzada de la columna: MntSweetProducts

## Distribució NO normalitzada de la columna: NumWebPurchases



## Distribució NO normalitzada de la columna: NumCatalogPurchases

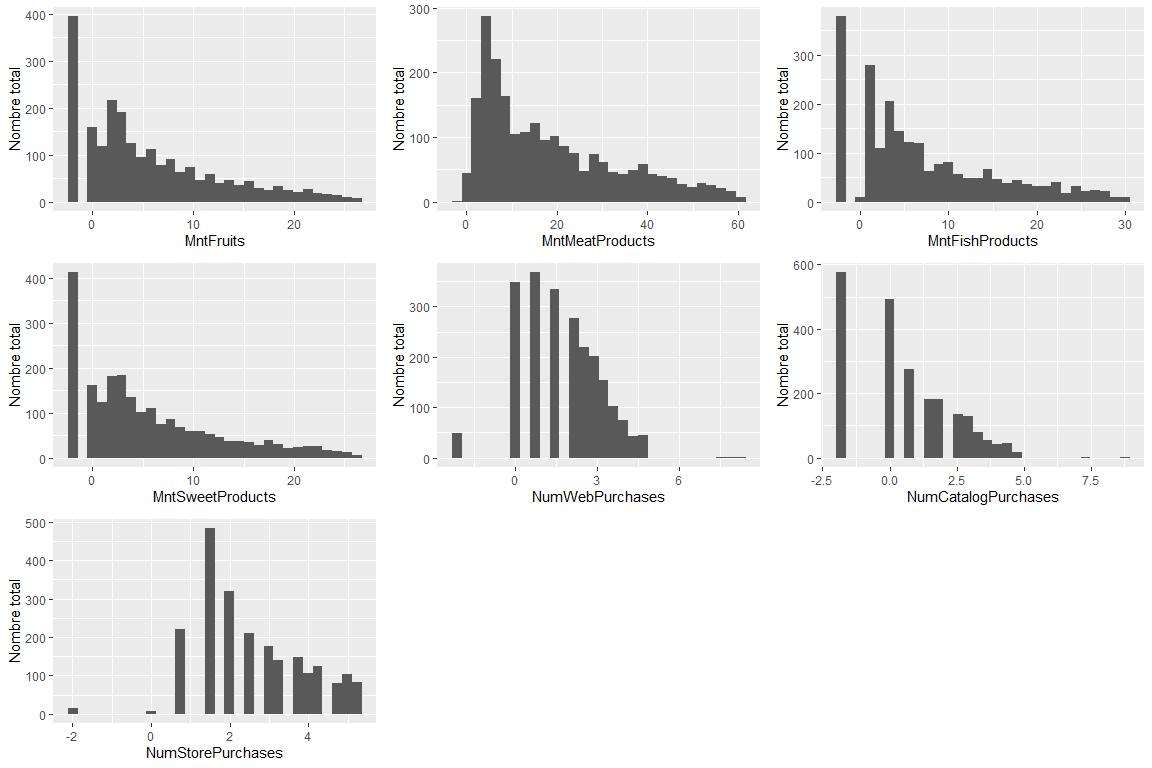
## Distribució NO normalitzada de la columna: NumStorePurchases



Es pot observar en el diagrama Q-Q i fent el test Shapiro-Wilk, que cap de les variables està normalitzada.

Es proposa normalitzar les variables següents, ja que la resta, es bo que es vegin els valors reals: MntFruits, MntMeatProducts, MntFishProducts, MntSweetProducts, NumWebPurchases, NumCatalogPurchases, NumStorePurchases. En aquest cas s’aplica la transformació Box-Cox de la llibreria forecast:

if (!require('ggplot2')) install.packages('ggplot2'); library('ggplot2')  
if (!require('forecast')) install.packages('forecast'); library('forecast')  
if (!require('gridExtra')) install.packages('gridExtra'); library('gridExtra')  
  
par(mfrow=c(2,2))  
  
data\_clients\_normalitzat <- data\_clients  
data\_clients\_normalitzat$MntFruits <- BoxCox(data\_clients[,c('MntFruits')], 1/2)  
data\_clients\_normalitzat$MntMeatProducts <- BoxCox(data\_clients[,c('MntMeatProducts')], 1/2)  
data\_clients\_normalitzat$MntFishProducts <- BoxCox(data\_clients[,c('MntFishProducts')], 1/2)  
data\_clients\_normalitzat$MntSweetProducts <- BoxCox(data\_clients[,c('MntSweetProducts')], 1/2)  
data\_clients\_normalitzat$NumWebPurchases <- BoxCox(data\_clients[,c('NumWebPurchases')], 1/2)  
data\_clients\_normalitzat$NumCatalogPurchases <- BoxCox(data\_clients[,c('NumCatalogPurchases')], 1/2)  
data\_clients\_normalitzat$NumStorePurchases <- BoxCox(data\_clients[,c('NumStorePurchases')], 1/2)  
  
ggplotFruits <- ggplot(data\_clients\_normalitzat, aes(x = MntFruits)) +  
 geom\_histogram() +  
 ylab("Nombre total") +  
 xlab("MntFruits")  
  
ggplotMeat <- ggplot(data\_clients\_normalitzat, aes(x = MntMeatProducts)) +  
 geom\_histogram() +  
 ylab("Nombre total") +  
 xlab("MntMeatProducts")  
  
ggplotFish <- ggplot(data\_clients\_normalitzat, aes(x = MntFishProducts)) +  
 geom\_histogram() +  
 ylab("Nombre total") +  
 xlab("MntFishProducts")  
  
ggplotSweet <- ggplot(data\_clients\_normalitzat, aes(x = MntSweetProducts)) +  
 geom\_histogram() +  
 ylab("Nombre total") +  
 xlab("MntSweetProducts")  
  
ggplotWebPurchases <- ggplot(data\_clients\_normalitzat, aes(x = NumWebPurchases)) +  
 geom\_histogram() +  
 ylab("Nombre total") +  
 xlab("NumWebPurchases")  
  
ggplotCatalogPurchases <- ggplot(data\_clients\_normalitzat, aes(x = NumCatalogPurchases)) +  
 geom\_histogram() +  
 ylab("Nombre total") +  
 xlab("NumCatalogPurchases")  
  
ggplotStorePurchases <- ggplot(data\_clients\_normalitzat, aes(x = NumStorePurchases)) +  
 geom\_histogram() +  
 ylab("Nombre total") +  
 xlab("NumStorePurchases")  
  
grid.arrange(ggplotFruits, ggplotMeat, ggplotFish, ggplotSweet, ggplotWebPurchases, ggplotCatalogPurchases, ggplotStorePurchases)



Un cop es tenen les dades normalitzades, es fa l’estudi de l’homoscedasticitat. S’utilitzarà el test de Fligner-Killeen ja que es volen comparar les dades normalitzades amb les que no ho han estat.

Tal i com passava amb el test Shapiro-Wilk, si el p-value és inferior a 0.05, es conclou que les dades no són homogènies:

fligner.test(Year\_Birth ~ MntFruits, data = data\_clients\_normalitzat)

##   
## Fligner-Killeen test of homogeneity of variances  
##   
## data: Year\_Birth by MntFruits  
## Fligner-Killeen:med chi-squared = 192.72, df = 157, p-value = 0.02758

fligner.test(Year\_Birth ~ MntMeatProducts, data = data\_clients\_normalitzat)

##   
## Fligner-Killeen test of homogeneity of variances  
##   
## data: Year\_Birth by MntMeatProducts  
## Fligner-Killeen:med chi-squared = 659.96, df = 550, p-value = 0.0008539

fligner.test(Year\_Birth ~ MntFishProducts, data = data\_clients\_normalitzat)

##   
## Fligner-Killeen test of homogeneity of variances  
##   
## data: Year\_Birth by MntFishProducts  
## Fligner-Killeen:med chi-squared = 211.5, df = 181, p-value = 0.05992

fligner.test(Year\_Birth ~ MntSweetProducts, data = data\_clients\_normalitzat)

##   
## Fligner-Killeen test of homogeneity of variances  
##   
## data: Year\_Birth by MntSweetProducts  
## Fligner-Killeen:med chi-squared = 235.24, df = 174, p-value = 0.001366

fligner.test(Year\_Birth ~ NumWebPurchases, data = data\_clients\_normalitzat)

##   
## Fligner-Killeen test of homogeneity of variances  
##   
## data: Year\_Birth by NumWebPurchases  
## Fligner-Killeen:med chi-squared = 19.778, df = 14, p-value = 0.1373

fligner.test(Year\_Birth ~ NumCatalogPurchases, data = data\_clients\_normalitzat)

##   
## Fligner-Killeen test of homogeneity of variances  
##   
## data: Year\_Birth by NumCatalogPurchases  
## Fligner-Killeen:med chi-squared = 52.732, df = 13, p-value =  
## 0.000001006

fligner.test(Year\_Birth ~ NumStorePurchases, data = data\_clients\_normalitzat)

##   
## Fligner-Killeen test of homogeneity of variances  
##   
## data: Year\_Birth by NumStorePurchases  
## Fligner-Killeen:med chi-squared = 33.543, df = 13, p-value = 0.001412

Les variables següents són homogènies ja que el p-value és superior a 0.05: Year\_Birth by MntFishProducts, Year\_Birth by NumWebPurchases. La resta no ho són.

### Aplicació de proves estadístiques per comparar els grups de dades

En funció de les dades i de l’objectiu de l’estudi, aplicar proves de contrast d’hipòtesis, correlacions, regressions etc.

#### Anàlisi de correlació:

Procedim a realitzar un anàlisis de correlació amb les diferents variables i així verificar si hi ha alguna variable que tingui una dependència prou gran per creure que pot influir amb el valor d’una altre variable:

* MntFruits
* MntMeatProducts
* MntFishProducts
* MntSweetProducts

Per fer l’estudi ytilitzarem les dades normalitzades (data\_clients\_normalitzat):

##### MntFruits

corr\_matrix <- matrix(nc = 2, nr = 0)  
colnames(corr\_matrix) <- c("estimate", "p-value")  
  
for (i in 1:(ncol(data\_clients\_normalitzat) - 1)) {  
 if (is.integer(data\_clients\_normalitzat[,i]) | is.numeric(data\_clients\_normalitzat[,i])) {  
 spearman\_test = cor.test(data\_clients\_normalitzat[,i],  
 data\_clients\_normalitzat[,9],  
 method = "spearman")  
 corr\_coef = spearman\_test$estimate  
 p\_val = spearman\_test$p.value  
 # Add row to matrix  
 pair = matrix(ncol = 2, nrow = 1)  
 pair[1][1] = corr\_coef  
 pair[2][1] = p\_val  
 corr\_matrix <- rbind(corr\_matrix, pair)  
 rownames(corr\_matrix)[nrow(corr\_matrix)] <- colnames(data\_clients\_normalitzat)[i]  
 }  
}  
print(corr\_matrix)

## estimate p-value  
## ID -0.02019695 3.419497e-01  
## Year\_Birth -0.02601272 2.209336e-01  
## Income 0.58255411 1.438579e-201  
## Kidhome -0.45291012 1.591442e-112  
## Teenhome -0.20209700 7.431139e-22  
## MntFruits 1.00000000 0.000000e+00  
## MntMeatProducts 0.71781609 0.000000e+00  
## MntFishProducts 0.70431621 0.000000e+00  
## MntSweetProducts 0.69218571 6.554068e-316  
## NumWebPurchases 0.47333580 3.993894e-124  
## NumCatalogPurchases 0.63308087 1.608609e-248

Les variables amb més correlació amb la quantitat de fruita comprada són els que estan més a prop dels valors -1 i 1.

En aquest cas les variables amb una correlació més alta són MntMeatProducts i MntSweetProducts però en cap cas amb un valor prou rellevant.

##### MntMeatProducts

corr\_matrix <- matrix(nc = 2, nr = 0)  
colnames(corr\_matrix) <- c("estimate", "p-value")  
  
for (i in 1:(ncol(data\_clients\_normalitzat) - 1)) {  
 if (is.integer(data\_clients\_normalitzat[,i]) | is.numeric(data\_clients\_normalitzat[,i])) {  
 spearman\_test = cor.test(data\_clients\_normalitzat[,i],  
 data\_clients\_normalitzat[,10],  
 method = "spearman")  
 corr\_coef = spearman\_test$estimate  
 p\_val = spearman\_test$p.value  
 # Add row to matrix  
 pair = matrix(ncol = 2, nrow = 1)  
 pair[1][1] = corr\_coef  
 pair[2][1] = p\_val  
 corr\_matrix <- rbind(corr\_matrix, pair)  
 rownames(corr\_matrix)[nrow(corr\_matrix)] <- colnames(data\_clients\_normalitzat)[i]  
 }  
}  
print(corr\_matrix)

## estimate p-value  
## ID -0.01317448 5.353513e-01  
## Year\_Birth -0.11763515 2.795690e-08  
## Income 0.81762150 0.000000e+00  
## Kidhome -0.55161676 1.152051e-176  
## Teenhome -0.12673293 2.141878e-09  
## MntFruits 0.71781609 0.000000e+00  
## MntMeatProducts 1.00000000 0.000000e+00  
## MntFishProducts 0.73033757 0.000000e+00  
## MntSweetProducts 0.70418375 0.000000e+00  
## NumWebPurchases 0.69084232 3.381804e-314  
## NumCatalogPurchases 0.85072506 0.000000e+00

Les variables amb més correlació amb la quantitat de carn comprada són els que estan més a prop dels valors -1 i 1.

En aquest cas la variable amb una correlació més alta és NumCatalogPurchases i Income per tant podem concloure que l’ingrés familiar i el número de compres realitzades per catàleg pot està relacionat amb les compres de carn d’un client.

##### MntFishProducts

corr\_matrix <- matrix(nc = 2, nr = 0)  
colnames(corr\_matrix) <- c("estimate", "p-value")  
  
for (i in 1:(ncol(data\_clients\_normalitzat) - 1)) {  
 if (is.integer(data\_clients\_normalitzat[,i]) | is.numeric(data\_clients\_normalitzat[,i])) {  
 spearman\_test = cor.test(data\_clients\_normalitzat[,i],  
 data\_clients\_normalitzat[,11],  
 method = "spearman")  
 corr\_coef = spearman\_test$estimate  
 p\_val = spearman\_test$p.value  
 # Add row to matrix  
 pair = matrix(ncol = 2, nrow = 1)  
 pair[1][1] = corr\_coef  
 pair[2][1] = p\_val  
 corr\_matrix <- rbind(corr\_matrix, pair)  
 rownames(corr\_matrix)[nrow(corr\_matrix)] <- colnames(data\_clients\_normalitzat)[i]  
 }  
}  
print(corr\_matrix)

## estimate p-value  
## ID -0.02975398 1.614622e-01  
## Year\_Birth -0.02943297 1.660348e-01  
## Income 0.57799738 9.874191e-198  
## Kidhome -0.45381093 5.089328e-113  
## Teenhome -0.23261773 1.301273e-28  
## MntFruits 0.70431621 0.000000e+00  
## MntMeatProducts 0.73033757 0.000000e+00  
## MntFishProducts 1.00000000 0.000000e+00  
## MntSweetProducts 0.70153023 0.000000e+00  
## NumWebPurchases 0.46629580 4.886843e-120  
## NumCatalogPurchases 0.65587758 1.041648e-272

Les variables amb més correlació amb la quantitat de peix comprada són els que estan més a prop dels valors -1 i 1.

En aquest cas les variables amb una correlació més alta són MntSweetProducts , MntMeatProducts i MntFruits , però en cap cas prou rellevant.

##### MntSweetProducts

corr\_matrix <- matrix(nc = 2, nr = 0)  
colnames(corr\_matrix) <- c("estimate", "p-value")  
  
for (i in 1:(ncol(data\_clients\_normalitzat) - 1)) {  
 if (is.integer(data\_clients\_normalitzat[,i]) | is.numeric(data\_clients\_normalitzat[,i])) {  
 spearman\_test = cor.test(data\_clients\_normalitzat[,i],  
 data\_clients\_normalitzat[,12],  
 method = "spearman")  
 corr\_coef = spearman\_test$estimate  
 p\_val = spearman\_test$p.value  
 # Add row to matrix  
 pair = matrix(ncol = 2, nrow = 1)  
 pair[1][1] = corr\_coef  
 pair[2][1] = p\_val  
 corr\_matrix <- rbind(corr\_matrix, pair)  
 rownames(corr\_matrix)[nrow(corr\_matrix)] <- colnames(data\_clients\_normalitzat)[i]  
 }  
}  
print(corr\_matrix)

## estimate p-value  
## ID -0.032605269 1.249274e-01  
## Year\_Birth 0.002926198 8.905000e-01  
## Income 0.567311438 5.730248e-189  
## Kidhome -0.442429025 7.153047e-107  
## Teenhome -0.199040076 3.102985e-21  
## MntFruits 0.692185706 6.554068e-316  
## MntMeatProducts 0.704183755 0.000000e+00  
## MntFishProducts 0.701530231 0.000000e+00  
## MntSweetProducts 1.000000000 0.000000e+00  
## NumWebPurchases 0.461072186 4.578519e-117  
## NumCatalogPurchases 0.629511822 6.469646e-245

Les variales amb més correlació amb la quantitat de dolços comprada són els que estan més a prop dels valors -1 i 1.

En aquest cas les variables amb una correlació més alta són MntMeatProducts i MntFruits , però en cap cas prou rellevant.

La conclusió que podem extreure en aquest cas és que els ingressos i l’ús del catàleg com a forma de comprar pot està relacionat amb els clients que compren Carn. Hauríem de fer un anàlisi més extens per poder veure com.

#### Anàlisi de contrast:

Mirem ara de fer una prova de contrast d’hipòtesi:

Com que quasi totes les relacions entre les variables del dataset original no són homogènies, utilitzarem el tests de kruskal entre els diferents variables i les variables d’estudi, revisant quin valor de p-value dona. Si és menor a 0,05 conclourem que els valors de la variable d’estudi és variant segons els diferents valors de l’altra variable.

##### MntFruits

cont\_matrix <- matrix(nc = 2, nr = 0)  
colnames(cont\_matrix) <- c("statistic", "p-value")  
  
for (i in 1:(ncol(data\_clients) - 1)) {  
 if (is.integer(data\_clients[,i]) | is.numeric(data\_clients[,i])) {  
 kruskal\_test = kruskal.test(MntFruits ~ data\_clients[,i] , data = data\_clients)  
 statistic = kruskal\_test$statistic  
 p\_val = kruskal\_test$p.value  
 # Add row to matrix  
 pair = matrix(ncol = 2, nrow = 1)  
 pair[1][1] = statistic  
 pair[2][1] = p\_val  
 cont\_matrix <- rbind(cont\_matrix, pair)  
 rownames(cont\_matrix)[nrow(cont\_matrix)] <- colnames(data\_clients)[i]  
 }  
}  
print(cont\_matrix)

## statistic p-value  
## ID 2215.00000 4.960041e-01  
## Year\_Birth 99.07386 2.494312e-04  
## Income 2194.88845 3.148948e-04  
## Kidhome 454.39833 2.131359e-99  
## Teenhome 95.90606 1.493700e-21  
## MntFruits 2215.00000 0.000000e+00  
## MntMeatProducts 1485.30082 5.015243e-87  
## MntFishProducts 1239.32770 4.516294e-157  
## MntSweetProducts 1197.88360 2.635318e-152  
## NumWebPurchases 607.19929 1.561218e-120  
## NumCatalogPurchases 929.68934 2.170981e-190

En aquest cas no tots els valors són inferior a 0,05. La variable ID no té rellevància com a dada informativa.

##### MntMeatProducts

cont\_matrix <- matrix(nc = 2, nr = 0)  
colnames(cont\_matrix) <- c("statistic", "p-value")  
  
for (i in 1:(ncol(data\_clients) - 1)) {  
 if (is.integer(data\_clients[,i]) | is.numeric(data\_clients[,i])) {  
 kruskal\_test = kruskal.test(MntMeatProducts ~ data\_clients[,i] , data = data\_clients)  
 statistic = kruskal\_test$statistic  
 p\_val = kruskal\_test$p.value  
 # Add row to matrix  
 pair = matrix(ncol = 2, nrow = 1)  
 pair[1][1] = statistic  
 pair[2][1] = p\_val  
 cont\_matrix <- rbind(cont\_matrix, pair)  
 rownames(cont\_matrix)[nrow(cont\_matrix)] <- colnames(data\_clients)[i]  
 }  
}  
print(cont\_matrix)

## statistic p-value  
## ID 2215.0000 4.960041e-01  
## Year\_Birth 124.6843 2.510831e-07  
## Income 2203.3756 1.969276e-04  
## Kidhome 676.6741 1.153689e-147  
## Teenhome 39.4317 2.738511e-09  
## MntFruits 1326.1974 4.353329e-184  
## MntMeatProducts 2215.0000 5.336403e-198  
## MntFishProducts 1360.7837 8.099178e-180  
## MntSweetProducts 1293.6684 3.084702e-170  
## NumWebPurchases 1244.1462 5.584708e-257  
## NumCatalogPurchases 1638.5785 0.000000e+00

En aquest cas numCatalogPurchases té un valor superior a 0,05. Un resultat coherent amb el resultat que ens ha donat anteriorment revisant la correlació de les variables.

##### MntFishProducts

cont\_matrix <- matrix(nc = 2, nr = 0)  
colnames(cont\_matrix) <- c("statistic", "p-value")  
for (i in 1:(ncol(data\_clients) - 1)) {  
 if (is.integer(data\_clients[,i]) | is.numeric(data\_clients[,i])) {  
 kruskal\_test = kruskal.test(MntFishProducts ~ data\_clients[,i] , data = data\_clients)  
 statistic = kruskal\_test$statistic  
 p\_val = kruskal\_test$p.value  
 # Add row to matrix  
 pair = matrix(ncol = 2, nrow = 1)  
 pair[1][1] = statistic  
 pair[2][1] = p\_val  
 cont\_matrix <- rbind(cont\_matrix, pair)  
 rownames(cont\_matrix)[nrow(cont\_matrix)] <- colnames(data\_clients)[i]  
 }  
}  
print(cont\_matrix)

## statistic p-value  
## ID 2215.00000 4.960041e-01  
## Year\_Birth 99.63201 2.177717e-04  
## Income 2197.07637 2.793712e-04  
## Kidhome 456.97052 5.889966e-100  
## Teenhome 121.55439 4.025305e-27  
## MntFruits 1220.39681 6.604070e-164  
## MntMeatProducts 1494.00385 3.193048e-88  
## MntFishProducts 2215.00000 0.000000e+00  
## MntSweetProducts 1216.04113 1.093422e-155  
## NumWebPurchases 589.96661 7.256462e-117  
## NumCatalogPurchases 988.19320 6.000256e-203

En aquest cas no tots els valors són inferior a 0,05. La variable ID no té rellevància com a dada informativa.

##### MntSweetProducts

cont\_matrix <- matrix(nc = 2, nr = 0)  
colnames(cont\_matrix) <- c("statistic", "p-value")  
for (i in 1:(ncol(data\_clients) - 1)) {  
 if (is.integer(data\_clients[,i]) | is.numeric(data\_clients[,i])) {  
 kruskal\_test = kruskal.test(MntSweetProducts ~ data\_clients[,i] , data = data\_clients)  
 statistic = kruskal\_test$statistic  
 p\_val = kruskal\_test$p.value  
 # Add row to matrix  
 pair = matrix(ncol = 2, nrow = 1)  
 pair[1][1] = statistic  
 pair[2][1] = p\_val  
 cont\_matrix <- rbind(cont\_matrix, pair)  
 rownames(cont\_matrix)[nrow(cont\_matrix)] <- colnames(data\_clients)[i]  
 }  
}  
print(cont\_matrix)

## statistic p-value  
## ID 2215.00000 4.960041e-01  
## Year\_Birth 99.45531 2.273524e-04  
## Income 2195.59668 3.029574e-04  
## Kidhome 433.57424 7.088414e-95  
## Teenhome 88.24296 6.891047e-20  
## MntFruits 1200.90595 3.245908e-160  
## MntMeatProducts 1480.38729 2.364737e-86  
## MntFishProducts 1256.57003 2.797089e-160  
## MntSweetProducts 2215.00000 0.000000e+00  
## NumWebPurchases 589.35758 9.778927e-117  
## NumCatalogPurchases 903.79680 7.795259e-185

En aquest cas no tots els valors són inferior a 0,05. La variable ID no té rellevància com a dada informativa.

Fem també una prova de t-test per les dades preparades anteriorment, segons als estudis del client.

Per molt que no siguin normals, en els casos que tenim una mida prou gran, podem utilitzar la prova.

Revisem el test entre els diferents grups d’estudis i estat civil per veure si hi ha alguna relació:

Revisem el número d’observacions per dataset:

print(paste("Número de Registres clients.basic: ", nrow(data\_clients.basic)))

## [1] "Número de Registres clients.basic: 54"

print(paste("Número de Registres clients.graduation: ",nrow(data\_clients.graduation)))

## [1] "Número de Registres clients.graduation: 1116"

print(paste("Número de Registres clients.cycle: ",nrow(data\_clients.cycle)))

## [1] "Número de Registres clients.cycle: 200"

print(paste("Número de Registres clients.master: ",nrow(data\_clients.master)))

## [1] "Número de Registres clients.master: 365"

print(paste("Número de Registres clients.phd: ",nrow(data\_clients.phd)))

## [1] "Número de Registres clients.phd: 481"

print(paste("Número de Registres clients.divorced: ",nrow(data\_clients.divorced)))

## [1] "Número de Registres clients.divorced: 232"

print(paste("Número de Registres clients.married: ",nrow(data\_clients.married)))

## [1] "Número de Registres clients.married: 857"

print(paste("Número de Registres clients.single: ",nrow(data\_clients.single)))

## [1] "Número de Registres clients.single: 2216"

print(paste("Número de Registres clients.together: ",nrow(data\_clients.together)))

## [1] "Número de Registres clients.together: 573"

print(paste("Número de Registres clients.widow: ",nrow(data\_clients.widow)))

## [1] "Número de Registres clients.widow: 76"

En el nostre cas, totes les mostres són majors de 30 registres per tant podem utilitzar l’anàlisi de contrast de t-student amb aquests conjunts de dades:

Fem un estudi respecte el que es gasta en fruita els diferents subconjunts que anteriorment hem calculat:

print(paste("t-students basic vs divorced:",t.test(data\_clients.basic$MntFruits, data\_clients.divorced$MntFruits,alternative = "less")$p.value))

## [1] "t-students basic vs divorced: 0.00000452988169591452"

print(paste("t-students basic vs married:",t.test(data\_clients.basic$MntFruits, data\_clients.married$MntFruits,alternative = "less")$p.value))

## [1] "t-students basic vs married: 0.000000472148918764225"

print(paste("t-students basic vs single:",t.test(data\_clients.basic$MntFruits, data\_clients.single$MntFruits,alternative = "less")$p.value))

## [1] "t-students basic vs single: 0.0000000530279715675004"

print(paste("t-students basic vs together:",t.test(data\_clients.basic$MntFruits, data\_clients.together$MntFruits,alternative = "less")$p.value))

## [1] "t-students basic vs together: 0.00000138112034268432"

print(paste("t-students basic vs widow:",t.test(data\_clients.basic$MntFruits, data\_clients.widow$MntFruits,alternative = "less")$p.value))

## [1] "t-students basic vs widow: 0.000312520499936441"

print(paste("t-students graduation vs divorced:",t.test(data\_clients.graduation$MntFruits, data\_clients.divorced$MntFruits,alternative = "less")$p.value))

## [1] "t-students graduation vs divorced: 0.877344370383466"

print(paste("t-students graduation vs married:",t.test(data\_clients.graduation$MntFruits, data\_clients.married$MntFruits,alternative = "less")$p.value))

## [1] "t-students graduation vs married: 0.997190965699978"

print(paste("t-students graduation vs single:",t.test(data\_clients.graduation$MntFruits, data\_clients.single$MntFruits,alternative = "less")$p.value))

## [1] "t-students graduation vs single: 0.997943367490982"

print(paste("t-students graduation vs together:",t.test(data\_clients.graduation$MntFruits, data\_clients.together$MntFruits,alternative = "less")$p.value))

## [1] "t-students graduation vs together: 0.994727751050491"

print(paste("t-students graduation vs widow:",t.test(data\_clients.graduation$MntFruits, data\_clients.widow$MntFruits,alternative = "less")$p.value))

## [1] "t-students graduation vs widow: 0.427855727431361"

print(paste("t-students cycle vs divorced:",t.test(data\_clients.cycle$MntFruits, data\_clients.divorced$MntFruits,alternative = "less")$p.value))

## [1] "t-students cycle vs divorced: 0.693880968375202"

print(paste("t-students cycle vs married:",t.test(data\_clients.cycle$MntFruits, data\_clients.married$MntFruits,alternative = "less")$p.value))

## [1] "t-students cycle vs married: 0.887083179738674"

print(paste("t-students cycle vs single:",t.test(data\_clients.cycle$MntFruits, data\_clients.single$MntFruits,alternative = "less")$p.value))

## [1] "t-students cycle vs single: 0.851280305888474"

print(paste("t-students cycle vs together:",t.test(data\_clients.cycle$MntFruits, data\_clients.together$MntFruits,alternative = "less")$p.value))

## [1] "t-students cycle vs together: 0.886174495597257"

print(paste("t-students cycle vs widow:",t.test(data\_clients.cycle$MntFruits, data\_clients.widow$MntFruits,alternative = "less")$p.value))

## [1] "t-students cycle vs widow: 0.338998229703505"

print(paste("t-students master vs divorced:",t.test(data\_clients.master$MntFruits, data\_clients.divorced$MntFruits,alternative = "less")$p.value))

## [1] "t-students master vs divorced: 0.0296423743908536"

print(paste("t-students master vs married:",t.test(data\_clients.master$MntFruits, data\_clients.married$MntFruits,alternative = "less")$p.value))

## [1] "t-students master vs married: 0.0289903069300666"

print(paste("t-students master vs single:",t.test(data\_clients.master$MntFruits, data\_clients.single$MntFruits,alternative = "less")$p.value))

## [1] "t-students master vs single: 0.00647927083873104"

print(paste("t-students master vs together:",t.test(data\_clients.master$MntFruits, data\_clients.together$MntFruits,alternative = "less")$p.value))

## [1] "t-students master vs together: 0.0450495707945353"

print(paste("t-students master vs widow:",t.test(data\_clients.master$MntFruits, data\_clients.widow$MntFruits,alternative = "less")$p.value))

## [1] "t-students master vs widow: 0.0334564827754901"

print(paste("t-students phd vs divorced:",t.test(data\_clients.phd$MntFruits, data\_clients.divorced$MntFruits,alternative = "less")$p.value))

## [1] "t-students phd vs divorced: 0.00845656164570771"

print(paste("t-students phd vs married:",t.test(data\_clients.phd$MntFruits, data\_clients.married$MntFruits,alternative = "less")$p.value))

## [1] "t-students phd vs married: 0.00312850427171141"

print(paste("t-students phd vs single:",t.test(data\_clients.phd$MntFruits, data\_clients.single$MntFruits,alternative = "less")$p.value))

## [1] "t-students phd vs single: 0.00016314988965631"

print(paste("t-students phd vs together:",t.test(data\_clients.phd$MntFruits, data\_clients.together$MntFruits,alternative = "less")$p.value))

## [1] "t-students phd vs together: 0.00776505558380226"

print(paste("t-students phd vs widow:",t.test(data\_clients.phd$MntFruits, data\_clients.widow$MntFruits,alternative = "less")$p.value))

## [1] "t-students phd vs widow: 0.0190675903734455"

En aquest cas veiem que els p-valors resultants de la prova t de Student més alts són els resultants dels clients que tenen estudis de graduació i de cicle per a qualsevol estat marital. Això vol dir que no s’observen diferències estadísticament significatives.

En canvi, en els altres casos, el p-valor és inferior a 0.05, per tant en aquets casos podem concloure que és més rellevant l’estat marital que els estudis a l’hora de comprar fruita.

Farem el mateix estudi amb les altres variables on treurem unes conclusions semblants en tots els casos:

##### MntMeatProducts

print(paste("t-students basic vs divorced:",t.test(data\_clients.basic$MntMeatProducts, data\_clients.divorced$MntMeatProducts,alternative = "less")$p.value))

## [1] "t-students basic vs divorced: 0.0000000000000000000000106683868459825"

print(paste("t-students basic vs married:",t.test(data\_clients.basic$MntMeatProducts, data\_clients.married$MntMeatProducts,alternative = "less")$p.value))

## [1] "t-students basic vs married: 0.000000000000000000000000000000000000000000000000000000000000000000081171180399022"

print(paste("t-students basic vs single:",t.test(data\_clients.basic$MntMeatProducts, data\_clients.single$MntMeatProducts,alternative = "less")$p.value))

## [1] "t-students basic vs single: 1.20871312295872e-125"

print(paste("t-students basic vs together:",t.test(data\_clients.basic$MntMeatProducts, data\_clients.together$MntMeatProducts,alternative = "less")$p.value))

## [1] "t-students basic vs together: 0.000000000000000000000000000000000000000000000000000173413914770287"

print(paste("t-students basic vs widow:",t.test(data\_clients.basic$MntMeatProducts, data\_clients.widow$MntMeatProducts,alternative = "less")$p.value))

## [1] "t-students basic vs widow: 0.000000000407817974644617"

print(paste("t-students graduation vs divorced:",t.test(data\_clients.graduation$MntMeatProducts, data\_clients.divorced$MntMeatProducts,alternative = "less")$p.value))

## [1] "t-students graduation vs divorced: 0.974058938138027"

print(paste("t-students graduation vs married:",t.test(data\_clients.graduation$MntMeatProducts, data\_clients.married$MntMeatProducts,alternative = "less")$p.value))

## [1] "t-students graduation vs married: 0.988193901006762"

print(paste("t-students graduation vs single:",t.test(data\_clients.graduation$MntMeatProducts, data\_clients.single$MntMeatProducts,alternative = "less")$p.value))

## [1] "t-students graduation vs single: 0.950841784310524"

print(paste("t-students graduation vs together:",t.test(data\_clients.graduation$MntMeatProducts, data\_clients.together$MntFruits,alternative = "less")$p.value))

## [1] "t-students graduation vs together: 1"

print(paste("t-students graduation vs widow:",t.test(data\_clients.graduation$MntMeatProducts, data\_clients.widow$MntMeatProducts,alternative = "less")$p.value))

## [1] "t-students graduation vs widow: 0.378878727878976"

print(paste("t-students cycle vs divorced:",t.test(data\_clients.cycle$MntMeatProducts, data\_clients.divorced$MntMeatProducts,alternative = "less")$p.value))

## [1] "t-students cycle vs divorced: 0.206130641913088"

print(paste("t-students cycle vs married:",t.test(data\_clients.cycle$MntMeatProducts, data\_clients.married$MntMeatProducts,alternative = "less")$p.value))

## [1] "t-students cycle vs married: 0.0953053949320644"

print(paste("t-students cycle vs single:",t.test(data\_clients.cycle$MntMeatProducts, data\_clients.single$MntMeatProducts,alternative = "less")$p.value))

## [1] "t-students cycle vs single: 0.0230222844863549"

print(paste("t-students cycle vs together:",t.test(data\_clients.cycle$MntMeatProducts, data\_clients.together$MntMeatProducts,alternative = "less")$p.value))

## [1] "t-students cycle vs together: 0.0405216750043374"

print(paste("t-students cycle vs widow:",t.test(data\_clients.cycle$MntMeatProducts, data\_clients.widow$MntMeatProducts,alternative = "less")$p.value))

## [1] "t-students cycle vs widow: 0.0388312320769009"

print(paste("t-students master vs divorced:",t.test(data\_clients.master$MntMeatProducts, data\_clients.divorced$MntMeatProducts,alternative = "less")$p.value))

## [1] "t-students master vs divorced: 0.77355834021978"

print(paste("t-students master vs married:",t.test(data\_clients.master$MntMeatProducts, data\_clients.married$MntMeatProducts,alternative = "less")$p.value))

## [1] "t-students master vs married: 0.709468196727142"

print(paste("t-students master vs single:",t.test(data\_clients.master$MntMeatProducts, data\_clients.single$MntMeatProducts,alternative = "less")$p.value))

## [1] "t-students master vs single: 0.461952541364877"

print(paste("t-students master vs together:",t.test(data\_clients.master$MntMeatProducts, data\_clients.together$MntMeatProducts,alternative = "less")$p.value))

## [1] "t-students master vs together: 0.482616837621924"

print(paste("t-students master vs widow:",t.test(data\_clients.master$MntMeatProducts, data\_clients.widow$MntMeatProducts,alternative = "less")$p.value))

## [1] "t-students master vs widow: 0.206626746542938"

print(paste("t-students phd vs divorced:",t.test(data\_clients.phd$MntMeatProducts, data\_clients.divorced$MntMeatProducts,alternative = "less")$p.value))

## [1] "t-students phd vs divorced: 0.80160757513977"

print(paste("t-students phd vs married:",t.test(data\_clients.phd$MntMeatProducts, data\_clients.married$MntMeatProducts,alternative = "less")$p.value))

## [1] "t-students phd vs married: 0.749489704805712"

print(paste("t-students phd vs single:",t.test(data\_clients.phd$MntMeatProducts, data\_clients.single$MntMeatProducts,alternative = "less")$p.value))

## [1] "t-students phd vs single: 0.471333066464326"

print(paste("t-students phd vs together:",t.test(data\_clients.phd$MntMeatProducts, data\_clients.together$MntMeatProducts,alternative = "less")$p.value))

## [1] "t-students phd vs together: 0.49394976105415"

print(paste("t-students phd vs widow:",t.test(data\_clients.phd$MntMeatProducts, data\_clients.widow$MntMeatProducts,alternative = "less")$p.value))

## [1] "t-students phd vs widow: 0.203924333993742"

##### MntFishProducts

print(paste("t-students basic vs divorced:",t.test(data\_clients.basic$MntFishProducts, data\_clients.divorced$MntFishProducts,alternative = "less")$p.value))

## [1] "t-students basic vs divorced: 0.000839768774094359"

print(paste("t-students basic vs married:",t.test(data\_clients.basic$MntFishProducts, data\_clients.married$MntFishProducts,alternative = "less")$p.value))

## [1] "t-students basic vs married: 0.000122982774680697"

print(paste("t-students basic vs single:",t.test(data\_clients.basic$MntFishProducts, data\_clients.single$MntFishProducts,alternative = "less")$p.value))

## [1] "t-students basic vs single: 0.00001569423945349"

print(paste("t-students basic vs together:",t.test(data\_clients.basic$MntFishProducts, data\_clients.together$MntFishProducts,alternative = "less")$p.value))

## [1] "t-students basic vs together: 0.0000163105953403171"

print(paste("t-students basic vs widow:",t.test(data\_clients.basic$MntFishProducts, data\_clients.widow$MntFishProducts,alternative = "less")$p.value))

## [1] "t-students basic vs widow: 0.0000792474277954104"

print(paste("t-students graduation vs divorced:",t.test(data\_clients.graduation$MntFishProducts, data\_clients.divorced$MntFishProducts,alternative = "less")$p.value))

## [1] "t-students graduation vs divorced: 0.984559761237548"

print(paste("t-students graduation vs married:",t.test(data\_clients.graduation$MntFishProducts, data\_clients.married$MntFishProducts,alternative = "less")$p.value))

## [1] "t-students graduation vs married: 0.999220185632399"

print(paste("t-students graduation vs single:",t.test(data\_clients.graduation$MntFishProducts, data\_clients.single$MntFishProducts,alternative = "less")$p.value))

## [1] "t-students graduation vs single: 0.996851055039929"

print(paste("t-students graduation vs together:",t.test(data\_clients.graduation$MntFishProducts, data\_clients.together$MntFishProducts,alternative = "less")$p.value))

## [1] "t-students graduation vs together: 0.929039966415018"

print(paste("t-students graduation vs widow:",t.test(data\_clients.graduation$MntFishProducts, data\_clients.widow$MntFishProducts,alternative = "less")$p.value))

## [1] "t-students graduation vs widow: 0.19069714998261"

print(paste("t-students cycle vs divorced:",t.test(data\_clients.cycle$MntFishProducts, data\_clients.divorced$MntFishProducts,alternative = "less")$p.value))

## [1] "t-students cycle vs divorced: 0.989242933263301"

print(paste("t-students cycle vs married:",t.test(data\_clients.cycle$MntFishProducts, data\_clients.married$MntFishProducts,alternative = "less")$p.value))

## [1] "t-students cycle vs married: 0.995318802635232"

print(paste("t-students cycle vs single:",t.test(data\_clients.cycle$MntFishProducts, data\_clients.single$MntFishProducts,alternative = "less")$p.value))

## [1] "t-students cycle vs single: 0.9875223876298"

print(paste("t-students cycle vs together:",t.test(data\_clients.cycle$MntFishProducts, data\_clients.together$MntFishProducts,alternative = "less")$p.value))

## [1] "t-students cycle vs together: 0.961654708933979"

print(paste("t-students cycle vs widow:",t.test(data\_clients.cycle$MntFishProducts, data\_clients.widow$MntFishProducts,alternative = "less")$p.value))

## [1] "t-students cycle vs widow: 0.412565013298578"

print(paste("t-students master vs divorced:",t.test(data\_clients.master$MntFishProducts, data\_clients.divorced$MntFishProducts,alternative = "less")$p.value))

## [1] "t-students master vs divorced: 0.207199167569599"

print(paste("t-students master vs married:",t.test(data\_clients.master$MntFishProducts, data\_clients.married$MntFishProducts,alternative = "less")$p.value))

## [1] "t-students master vs married: 0.107507622033249"

print(paste("t-students master vs single:",t.test(data\_clients.master$MntFishProducts, data\_clients.single$MntFishProducts,alternative = "less")$p.value))

## [1] "t-students master vs single: 0.0173493040351401"

print(paste("t-students master vs together:",t.test(data\_clients.master$MntFishProducts, data\_clients.together$MntFishProducts,alternative = "less")$p.value))

## [1] "t-students master vs together: 0.0170846926028519"

print(paste("t-students master vs widow:",t.test(data\_clients.master$MntFishProducts, data\_clients.widow$MntFishProducts,alternative = "less")$p.value))

## [1] "t-students master vs widow: 0.008916691226454"

print(paste("t-students phd vs divorced:",t.test(data\_clients.phd$MntFishProducts, data\_clients.divorced$MntFishProducts,alternative = "less")$p.value))

## [1] "t-students phd vs divorced: 0.0190813859689535"

print(paste("t-students phd vs married:",t.test(data\_clients.phd$MntFishProducts, data\_clients.married$MntFishProducts,alternative = "less")$p.value))

## [1] "t-students phd vs married: 0.000496948293659841"

print(paste("t-students phd vs single:",t.test(data\_clients.phd$MntFishProducts, data\_clients.single$MntFishProducts,alternative = "less")$p.value))

## [1] "t-students phd vs single: 0.000000763092319544463"

print(paste("t-students phd vs together:",t.test(data\_clients.phd$MntFishProducts, data\_clients.together$MntFishProducts,alternative = "less")$p.value))

## [1] "t-students phd vs together: 0.0000330559195393481"

print(paste("t-students phd vs widow:",t.test(data\_clients.phd$MntFishProducts, data\_clients.widow$MntFishProducts,alternative = "less")$p.value))

## [1] "t-students phd vs widow: 0.00127582635080299"

##### MntSweetProducts

print(paste("t-students basic vs divorced:",t.test(data\_clients.basic$MntSweetProducts, data\_clients.divorced$MntSweetProducts,alternative = "less")$p.value))

## [1] "t-students basic vs divorced: 0.0000834863193781449"

print(paste("t-students basic vs married:",t.test(data\_clients.basic$MntSweetProducts, data\_clients.married$MntSweetProducts,alternative = "less")$p.value))

## [1] "t-students basic vs married: 0.00000314876391932562"

print(paste("t-students basic vs single:",t.test(data\_clients.basic$MntSweetProducts, data\_clients.single$MntSweetProducts,alternative = "less")$p.value))

## [1] "t-students basic vs single: 0.000000982009610404422"

print(paste("t-students basic vs together:",t.test(data\_clients.basic$MntSweetProducts, data\_clients.together$MntSweetProducts,alternative = "less")$p.value))

## [1] "t-students basic vs together: 0.0000107371405344713"

print(paste("t-students basic vs widow:",t.test(data\_clients.basic$MntSweetProducts, data\_clients.widow$MntSweetProducts,alternative = "less")$p.value))

## [1] "t-students basic vs widow: 0.0000484024189980965"

print(paste("t-students graduation vs divorced:",t.test(data\_clients.graduation$MntSweetProducts, data\_clients.divorced$MntSweetProducts,alternative = "less")$p.value))

## [1] "t-students graduation vs divorced: 0.930414781728455"

print(paste("t-students graduation vs married:",t.test(data\_clients.graduation$MntSweetProducts, data\_clients.married$MntSweetProducts,alternative = "less")$p.value))

## [1] "t-students graduation vs married: 0.990589521437538"

print(paste("t-students graduation vs single:",t.test(data\_clients.graduation$MntSweetProducts, data\_clients.single$MntSweetProducts,alternative = "less")$p.value))

## [1] "t-students graduation vs single: 0.99719744263374"

print(paste("t-students graduation vs together:",t.test(data\_clients.graduation$MntSweetProducts, data\_clients.together$MntSweetProducts,alternative = "less")$p.value))

## [1] "t-students graduation vs together: 0.991368510400776"

print(paste("t-students graduation vs widow:",t.test(data\_clients.graduation$MntSweetProducts, data\_clients.widow$MntSweetProducts,alternative = "less")$p.value))

## [1] "t-students graduation vs widow: 0.13392388261565"

print(paste("t-students cycle vs divorced:",t.test(data\_clients.cycle$MntSweetProducts, data\_clients.divorced$MntSweetProducts,alternative = "less")$p.value))

## [1] "t-students cycle vs divorced: 0.963303328895706"

print(paste("t-students cycle vs married:",t.test(data\_clients.cycle$MntSweetProducts, data\_clients.married$MntSweetProducts,alternative = "less")$p.value))

## [1] "t-students cycle vs married: 0.982896510606811"

print(paste("t-students cycle vs single:",t.test(data\_clients.cycle$MntSweetProducts, data\_clients.single$MntSweetProducts,alternative = "less")$p.value))

## [1] "t-students cycle vs single: 0.984973278678693"

print(paste("t-students cycle vs together:",t.test(data\_clients.cycle$MntSweetProducts, data\_clients.together$MntSweetProducts,alternative = "less")$p.value))

## [1] "t-students cycle vs together: 0.985963039068751"

print(paste("t-students cycle vs widow:",t.test(data\_clients.cycle$MntSweetProducts, data\_clients.widow$MntSweetProducts,alternative = "less")$p.value))

## [1] "t-students cycle vs widow: 0.320313245778639"

print(paste("t-students master vs divorced:",t.test(data\_clients.master$MntSweetProducts, data\_clients.divorced$MntSweetProducts,alternative = "less")$p.value))

## [1] "t-students master vs divorced: 0.032585798684264"

print(paste("t-students master vs married:",t.test(data\_clients.master$MntSweetProducts, data\_clients.married$MntSweetProducts,alternative = "less")$p.value))

## [1] "t-students master vs married: 0.00468586443324562"

print(paste("t-students master vs single:",t.test(data\_clients.master$MntSweetProducts, data\_clients.single$MntSweetProducts,alternative = "less")$p.value))

## [1] "t-students master vs single: 0.00114878210062781"

print(paste("t-students master vs together:",t.test(data\_clients.master$MntSweetProducts, data\_clients.together$MntSweetProducts,alternative = "less")$p.value))

## [1] "t-students master vs together: 0.0134153318253001"

print(paste("t-students master vs widow:",t.test(data\_clients.master$MntSweetProducts, data\_clients.widow$MntSweetProducts,alternative = "less")$p.value))

## [1] "t-students master vs widow: 0.00285890572734954"

print(paste("t-students phd vs divorced:",t.test(data\_clients.phd$MntSweetProducts, data\_clients.divorced$MntSweetProducts,alternative = "less")$p.value))

## [1] "t-students phd vs divorced: 0.0120635571715529"

print(paste("t-students phd vs married:",t.test(data\_clients.phd$MntSweetProducts, data\_clients.married$MntSweetProducts,alternative = "less")$p.value))

## [1] "t-students phd vs married: 0.000367966652499097"

print(paste("t-students phd vs single:",t.test(data\_clients.phd$MntSweetProducts, data\_clients.single$MntSweetProducts,alternative = "less")$p.value))

## [1] "t-students phd vs single: 0.0000205220098071218"

print(paste("t-students phd vs together:",t.test(data\_clients.phd$MntSweetProducts, data\_clients.together$MntSweetProducts,alternative = "less")$p.value))

## [1] "t-students phd vs together: 0.00204737475189758"

print(paste("t-students phd vs widow:",t.test(data\_clients.phd$MntSweetProducts, data\_clients.widow$MntSweetProducts,alternative = "less")$p.value))

## [1] "t-students phd vs widow: 0.0015798113444258"

#### Anàlisi de regressió:

Per obtenir un model de regressió lineal eficient ho farem a partir de les variables amb més correlació que haguem trobat de les variables en estudi.

En aquest cas, i segons els estudis anteriors, ens centrarem amb la variable MntMeatProducts i

* Income
* NumCatalogPurchase

##### Generació del model:

model1 <- lm(data\_clients$MntMeatProducts ~ data\_clients$Income , data = data\_clients)  
model2 <- lm(data\_clients$MntMeatProducts ~ data\_clients$NumCatalogPurchase , data = data\_clients)  
model3 <- lm(data\_clients$MntMeatProducts ~ data\_clients$Income + data\_clients$NumCatalogPurchase , data = data\_clients)  
model4 <- lm(data\_clients$MntMeatProducts ~ data\_clients$Income + data\_clients$NumCatalogPurchase + data\_clients$MntFruits , data = data\_clients)  
model5 <- lm(data\_clients$MntMeatProducts ~ data\_clients$Income + data\_clients$NumCatalogPurchase + data\_clients$MntFruits + data\_clients$MntFishProducts , data = data\_clients)  
  
tabla.quoficients <- matrix(c(  
1, summary(model1)$r.squared,  
2, summary(model2)$r.squared,  
3, summary(model3)$r.squared,  
4, summary(model4)$r.squared,  
5, summary(model5)$r.squared),  
  
ncol = 2, byrow = TRUE)  
colnames(tabla.quoficients) <- c("Model", "R^2")  
tabla.quoficients

## Model R^2  
## [1,] 1 0.4637420  
## [2,] 2 0.4304784  
## [3,] 3 0.5279354  
## [4,] 4 0.5767456  
## [5,] 5 0.6011374

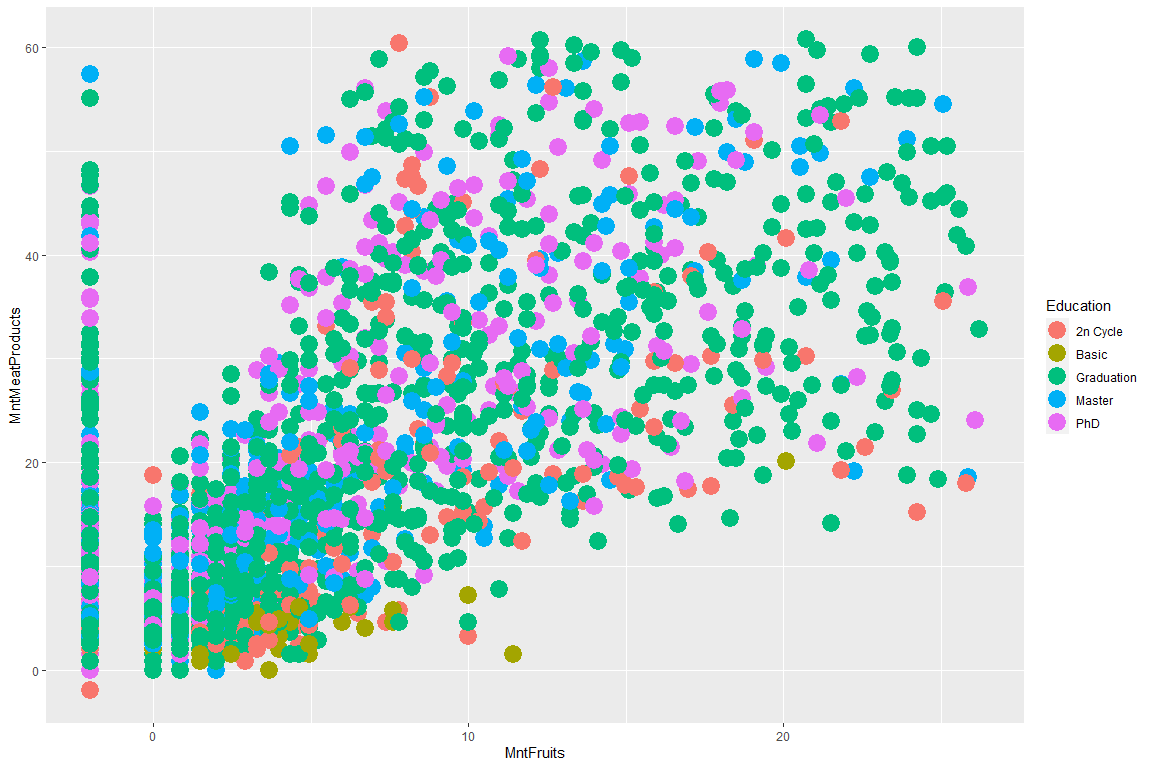
En aquest cas veiem que tots tenen una coeficient de determinació baix, per tant ens quedem l’últim model per fer la predicció del número de diners gastats per un client amb carn.

## Representació dels resultats a partir de taules gràfiques

Les taules gràfiques han d’ajudar a representat els resultats mostrats anteriorment.

Primerament, es comprova que les compres realitzades a la fruita i la carn en relació als estudis que tenen (s’ha comprovat en les proves estadístiques que els estudis i les compres de fruita tenen relació, s’ha afegit en aquest gràfic les compres de carn per a poder contrastar el diferents productes):

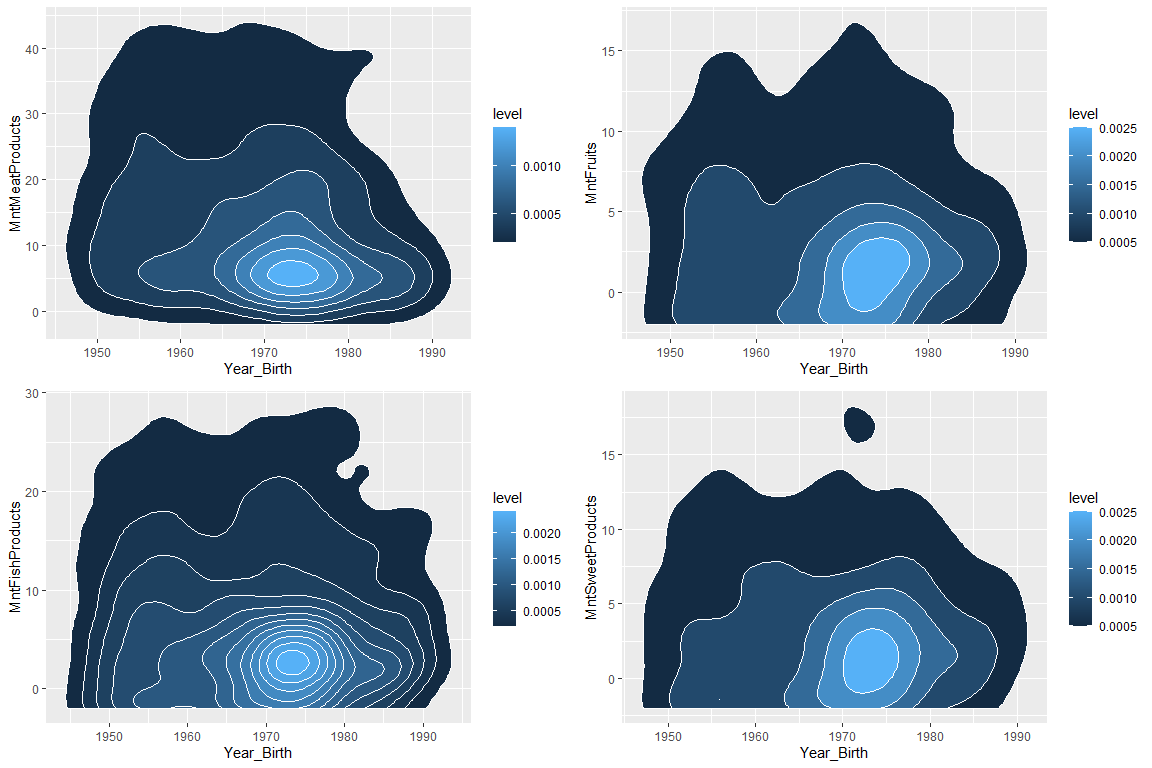
if (!require('ggplot2')) install.packages('ggplot2'); library(ggplot2)  
  
ggplot(data\_clients\_normalitzat, aes(x=MntFruits, y=MntMeatProducts, color=Education)) +   
 geom\_point(size=6)



Es pot observar que hi ha una gran diferència de les persones amb estudis bàsics, que gasten menys diners en la compra de fruita i de carn. Tot i així, les persones amb la resta d’estudis, compren per sobre de la mitjana.

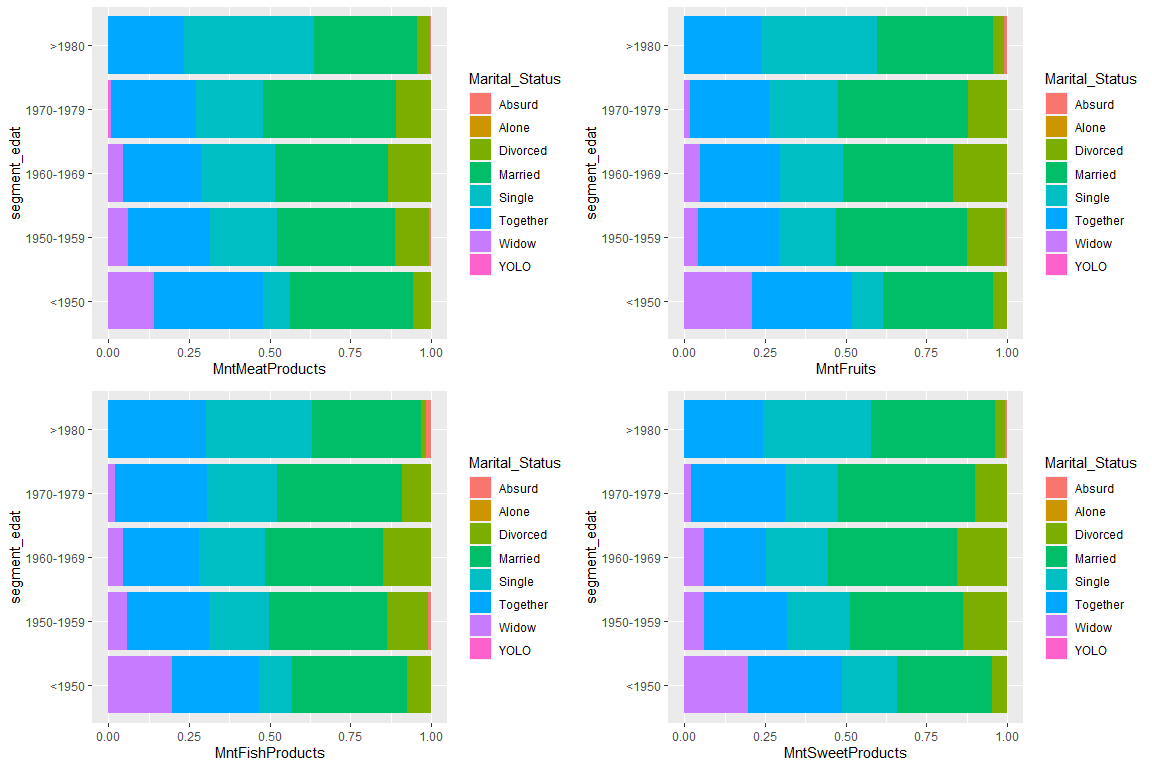
Degut a que l’educació no és un factor rellevant, es decideix fer ús de l’edat amb les variables en referència a la compra de fruita, carn, peix i dolços:

plotMeat <- ggplot(data\_clients\_normalitzat, aes(x=Year\_Birth, y=MntMeatProducts) ) +  
 stat\_density\_2d(aes(fill = ..level..), geom = "polygon", colour="white")  
  
plotFruits <- ggplot(data\_clients\_normalitzat, aes(x=Year\_Birth, y=MntFruits) ) +  
 stat\_density\_2d(aes(fill = ..level..), geom = "polygon", colour="white")  
  
plotFish <- ggplot(data\_clients\_normalitzat, aes(x=Year\_Birth, y=MntFishProducts) ) +  
 stat\_density\_2d(aes(fill = ..level..), geom = "polygon", colour="white")  
  
plotSweet <- ggplot(data\_clients\_normalitzat, aes(x=Year\_Birth, y=MntSweetProducts) ) +  
 stat\_density\_2d(aes(fill = ..level..), geom = "polygon", colour="white")  
  
grid.arrange(plotMeat, plotFruits, plotFish, plotSweet)



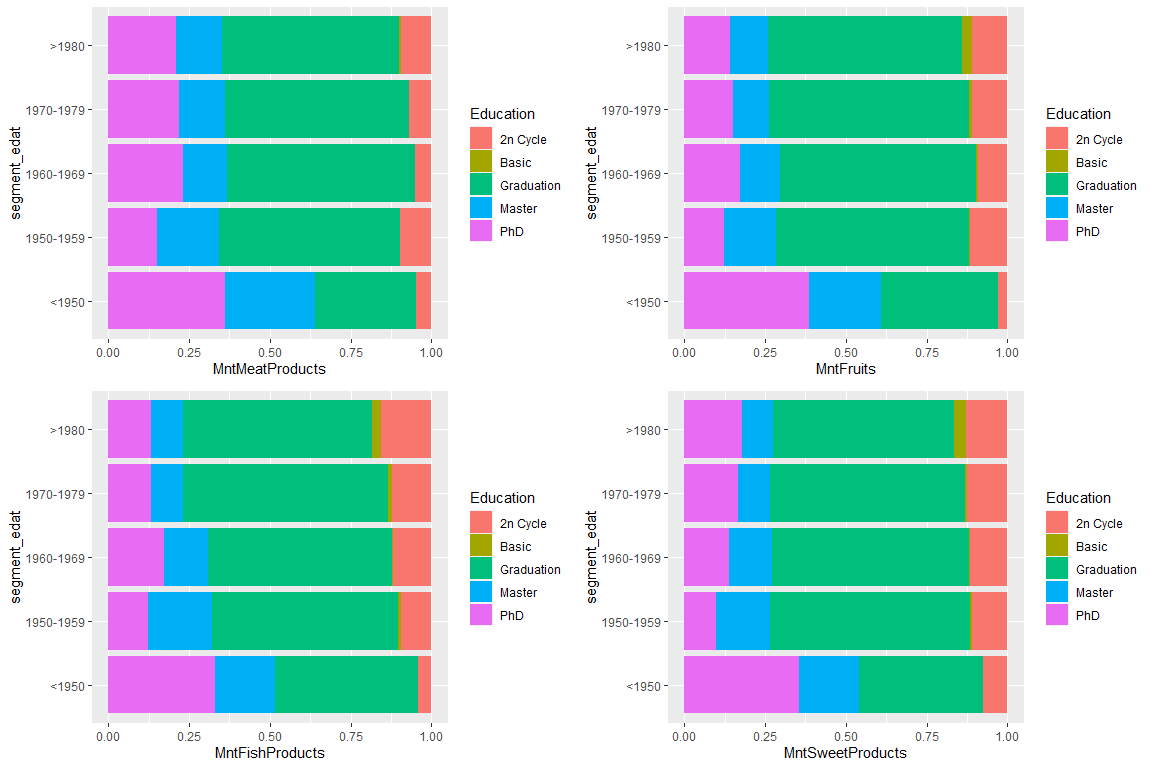
En aquest cas s’observa que l’edat (any de naixement) és una bona variable que influeix en la compra de productes. Per això es discretitza l’edat en 5 segments, i es visualitza per a cada segment d’edat i estat civil, el nombre de productes comprats.

data\_clients["segment\_edat"] <- cut(data\_clients$Year\_Birth, breaks = c(0, 1950, 1960, 1970, 1980, Inf),  
 labels = c("<1950", "1950-1959", "1960-1969", "1970-1979", ">1980"))  
  
  
plotMeat <- ggplot(data\_clients, aes(fill=Marital\_Status, y=segment\_edat, x=MntMeatProducts)) +   
 geom\_bar(position="fill", stat="identity")  
  
plotFish <- ggplot(data\_clients, aes(fill=Marital\_Status, y=segment\_edat, x=MntFishProducts)) +   
 geom\_bar(position="fill", stat="identity")  
  
plotFruits <- ggplot(data\_clients, aes(fill=Marital\_Status, y=segment\_edat, x=MntFruits)) +   
 geom\_bar(position="fill", stat="identity")  
  
plotSweet <- ggplot(data\_clients, aes(fill=Marital\_Status, y=segment\_edat, x=MntSweetProducts)) +   
 geom\_bar(position="fill", stat="identity")  
  
grid.arrange(plotMeat, plotFruits, plotFish, plotSweet)



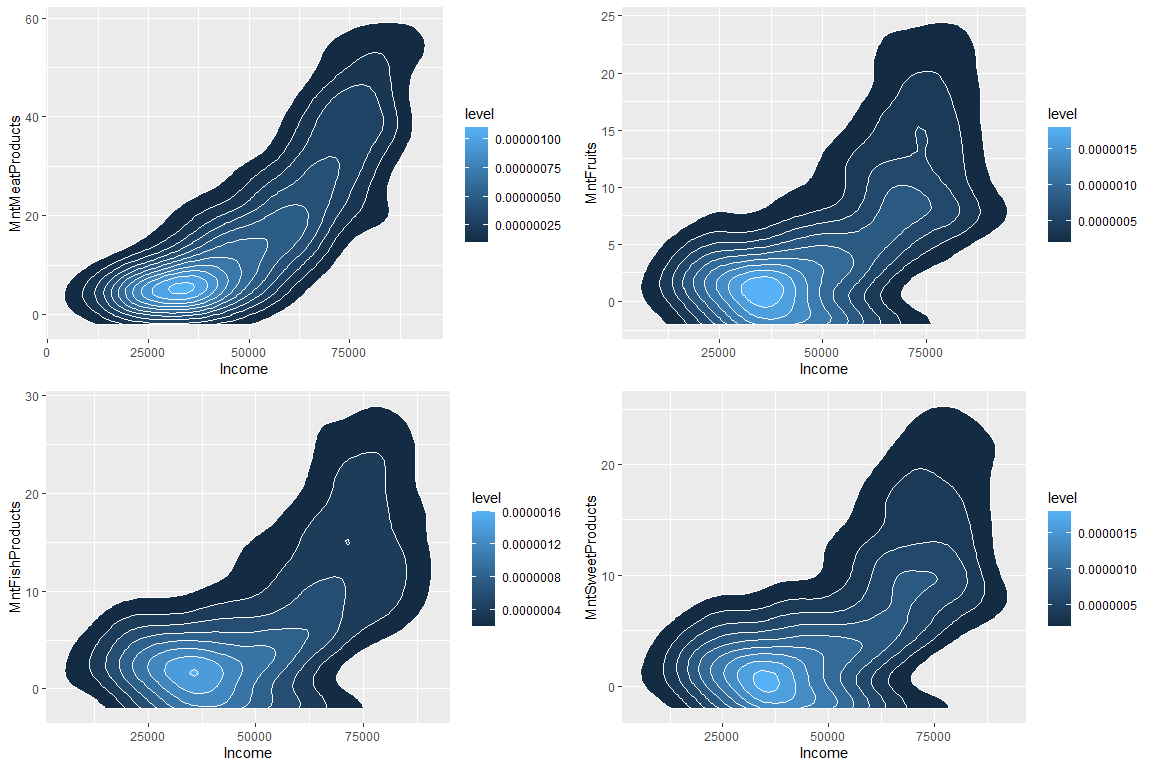
Es fa el mateix per a cada segment d’edat i l’educació, visualitzar el nombre de productes comprats:

plotMeat <- ggplot(data\_clients, aes(fill=Education, y=segment\_edat, x=MntMeatProducts)) +   
 geom\_bar(position="fill", stat="identity")  
  
plotFish <- ggplot(data\_clients, aes(fill=Education, y=segment\_edat, x=MntFishProducts)) +   
 geom\_bar(position="fill", stat="identity")  
  
plotFruits <- ggplot(data\_clients, aes(fill=Education, y=segment\_edat, x=MntFruits)) +   
 geom\_bar(position="fill", stat="identity")  
  
plotSweet <- ggplot(data\_clients, aes(fill=Education, y=segment\_edat, x=MntSweetProducts)) +   
 geom\_bar(position="fill", stat="identity")  
  
grid.arrange(plotMeat, plotFruits, plotFish, plotSweet)



Finalment, es comprova les compres realitzades segons els diners gastats en les compres dels productes:

plotMeat <- ggplot(data\_clients\_normalitzat, aes(x=Income, y=MntMeatProducts) ) +  
 stat\_density\_2d(aes(fill = ..level..), geom = "polygon", colour="white")  
  
plotFruits <- ggplot(data\_clients\_normalitzat, aes(x=Income, y=MntFruits) ) +  
 stat\_density\_2d(aes(fill = ..level..), geom = "polygon", colour="white")  
  
plotFish <- ggplot(data\_clients\_normalitzat, aes(x=Income, y=MntFishProducts) ) +  
 stat\_density\_2d(aes(fill = ..level..), geom = "polygon", colour="white")  
  
plotSweet <- ggplot(data\_clients\_normalitzat, aes(x=Income, y=MntSweetProducts) ) +  
 stat\_density\_2d(aes(fill = ..level..), geom = "polygon", colour="white")  
  
grid.arrange(plotMeat, plotFruits, plotFish, plotSweet)



## Resolució del problema

A partir dels resultats obtinguts, quines són les conclusions? Els resultats permeten respondre al problema?

L’objectiu d’aquest estudi és identificar el tipus de client que compra cada producte i crear així campanyes de màrqueting específiques.

Abans de començar amb l’anàlisi de les dades, s’han seleccionat aquelles variables que aporten més a la resolució del problema plantejar (són variables relacionades amb el perfil del client i el gasto total segons el tipus de producte).

Posteriorment, s’han eliminat els elements buits de la variable ‘Income’, ja que era l’única, la resta tenien els valors informats. I s’han tractat els valors extrems de ‘Year\_Birth’, ‘MntSweetProducts’, ‘MntMeatProducts’ i ‘Income’.

Un cop les dades estaves ja preparades, s’ha iniciat l’anàlisi estadístic. Primerament, s’ha aplicat un anàlisi de correlació per identificar les dependències entre les variables. Es comprova que hi ha una dependència entre les compres dels diferents productes carn, dolços, peix i fruita. I més concretament, entre els productes de carn, l’ingrés familiar i el número de compres realitzada. Després, es fa un anàlisi de contrast d’hipòtesi que permet comparar les mitjanes entre els diferents grups. Aquí hem pogut observar que l’estat marital és més rellevant en les compres que no pas els estudis.

Finalment, amb la representació dels resultats es conclou el següent:

* Els aspectes que més influeixen en la compra són els ingressos anuals i l’edat del client. L’educació ens indica que els clients amb estudis bàsics, compren menys, però això es degut a que tenen feines menys qualificades i amb menys ingressos.
* Els clients més habituals són els nascuts entre 1970 i 1975, però el gasto per a cada producte és baix.
* Hi ha una relació directa entre els ingressos mensuals i el gasto fet en les compres dels productes. Tot i així, és important saber que la majoria dels clients tenen un ingrés de 35,000 dollars anuals. I el gasto de productes augmenta a partir dels 45,000 i 50,000 dollars.
* Pel que fa als estudis, els clients més habituals són els graduats, a excepció dels nascuts abans del 1950 que tenen un PhD.

|  |  |
| --- | --- |
| Contribucions | Signatura |
| Investigació prèvia | Isabel Barrera Benavent, Maria Font Sánchez |
| Redacció de les respostes | Isabel Barrera Benavent, Maria Font Sánchez |
| Desenvolupament del codi | Isabel Barrera Benavent, Maria Font Sánchez |