PRACTICA - 2

Autor

31 de diciembre, 2022

Contents

1. Descripció del dataset

Per a la realització d'aquesta segona pràctica s'utilitza el *dataset* generat a la primera pràctica, i es combina amb altres *datasets* que resultin d'interès, per poder realitzar un anàlisis més profund, tenint en compte factors socioeconòmics.

Per a la correcta execució del *script* és imprescindible definir la ruta on es troba l'arxiu **Practica_2.Rmd** com a *working directory*.

A continuació procedeix a la lectura de l'arxiu de dades amb opcions, escollint el separador i el tipus de codificació.

```
fueldata <- read.csv(paste(datadir, "FuelScraper", "dataset.csv", sep = "/"),
                     encoding="UTF-8", sep=";")
summary(fueldata)
```

```
##  Capture_date       Capture_time         Province            City
##  Length:195357      Length:195357      Length:195357      Length:195357
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##    Address            Road_side         Update_date           Price
##  Length:195357      Length:195357      Length:195357      Min.   :0.768
##  Class :character   Class :character   Class :character   1st Qu.:1.779
##  Mode  :character   Mode  :character   Mode  :character   Median :1.879
##                                                           Mean   :1.836
##                                                           3rd Qu.:1.959
##                                                           Max.   :3.700
##     Brand              Sale_1             Sale_2           Fuel_type
##  Length:195357      Length:195357      Length:195357      Length:195357
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
```

El primer anàlisi del dataset indica que pot ser interessant canviar algunes dades a tipus factor, així com canviar el format de les variables temporals.

```
#Vector de variables a modificar
t_vector<-c("Province","Road_side","Sale_1", "Sale_2", "Fuel_type")

#Loop
for (i in t_vector){
  #Canvi de tipus a factor
  fueldata[,i]<-as.factor(fueldata[,i])
}
```

```
fueldata$Capture_date<-as.Date(fueldata$Capture_date, format = "%Y/%m/%d")
fueldata$Update_date<-as.Date(fueldata$Update_date, format = "%d/%m/%Y")
fueldata$Capture_time<-lubridate::hms(fueldata$Capture_time)
```

Com a darrer pas en la càrrega del *dataset* original es generen variables de *backup* per a *Province* i *City* ja que aquestes posteriorment s'hauran de modificar a través de processos de normalització de noms per tal de fer-les compatibles amb les dades dels altres *datasets* a integrar.

```
fueldata$bckup.Province<-fueldata$Province
fueldata$bckup.City<-fueldata$City
```

2. Integració i selecció

Amb l'objectiu d'obtenir un *dataset* amb més informació integrada, es llegeix un arxiu addicional amb el cens de població per municipis. Aquesta informació es extreta de l'Institut Nacional d'Estadística (INE). En aquest cas, la lectura també es realitza amb opcions.

```
pobdata <- read.csv(paste(datadir, "pobmun", "pobmun22.csv", sep = "/"),
                    encoding="UTF-8", sep=";")
```

Es canvien els noms de les variables i transformem les dades a majúscules per habilitar posteriors comparacions entre els dos *datasets*.

```
#Canvis de noms
names(pobdata)[names(pobdata) == "PROVINCIA"] <- "Province"
names(pobdata)[names(pobdata) == "NOMBRE"] <- "City"
names(pobdata)[names(pobdata) == "CPRO"] <- "P_code"
names(pobdata)[names(pobdata) == "CMUN"] <- "C_code"
names(pobdata)[names(pobdata) == "POB22"] <- "Population"
names(pobdata)[names(pobdata) == "HOMBRES"] <- "P_Male"
names(pobdata)[names(pobdata) == "MUJERES"] <- "P_Female"

#Transformació a majúscules
pobdata$Province<-toupper(pobdata$Province)
pobdata$City<-toupper(pobdata$City)
```

També, es normalitzen les paraules en les variables *Province* i *City* dels dos datasets, eliminant accénts i caràcters especials com la *ñ*. Per fer-ho es canvia el tipus de dades d'aquestes variables de UTF-8 a ASCII.

```
fueldata$Province<-iconv(fueldata$Province, from = 'UTF-8', to = 'ASCII//TRANSLIT')
fueldata$City<-iconv(fueldata$City, from = 'UTF-8', to = 'ASCII//TRANSLIT')
pobdata$Province<-iconv(pobdata$Province, from = 'UTF-8', to = 'ASCII//TRANSLIT')
pobdata$City<-iconv(pobdata$City, from = 'UTF-8', to = 'ASCII//TRANSLIT')
```

Es canvia la denominació de 3 províncies per tal de fer la informació compatible entre els *datasets* de preus de combustibles i de població per municipis.

```
fueldata[fueldata$Province=="ALICANTE","Province"]<-"ALICANTE/ALACANT"
fueldata[fueldata$Province=="VALENCIA / VALENCIA","Province"]<-"VALENCIA/VALENCIA"
fueldata[fueldata$Province=="CASTELLON / CASTELLO","Province"]<-"CASTELLON/CASTELLO"
```

A continuació es modifica l'ús d'articles en els camps *Province* i *City* utilitzant RegEx, també buscant la compatiblitat entre *datasets*.

```
fueldata$Province<-sub("(\\w+) \\((\\w+)\\)","\\1, \\2",fueldata$Province, fixed=FALSE)
fueldata$City<-sub("((\\w| )+) \\(((\\w|')+)\\)","\\1, \\3",fueldata$City, fixed=FALSE)
```

Finalment es realitzen tot un seguit de canvis individuals (que no es mostren en la memòria, però si en el codi), per tal de maximitzar la informació disponible en el *dataset* resultant.

S'integren els dos *datasets* amb l'objectiu d'obtenir un únic *dataset* resultant que contingui tota la informació combinada. Aquesta integració es realitza de manera completa (all = TRUE), per tal de garantir que les dades que no tenen una parella en l'altre *dataset* es mantenen afegint *NA* en la informació.

```
total<-merge(fueldata, pobdata, by=c("Province", "City"), all = TRUE)
```

Finalment, i per acabar amb les tasques de selecció es seleccionen les dades:

- del dia 16 de Novembre
- els combustibles
  - Gasóleo A habitual
  - Gasolina 95 E5

```
data<-total[total$Capture_date == as.Date("2022/11/16", format ="%Y/%m/%d"),]
data<-data[data$Fuel_type == "Gasóleo A habitual" | data$Fuel_type == "Gasolina 95 E5",]
```

3. Neteja de dades

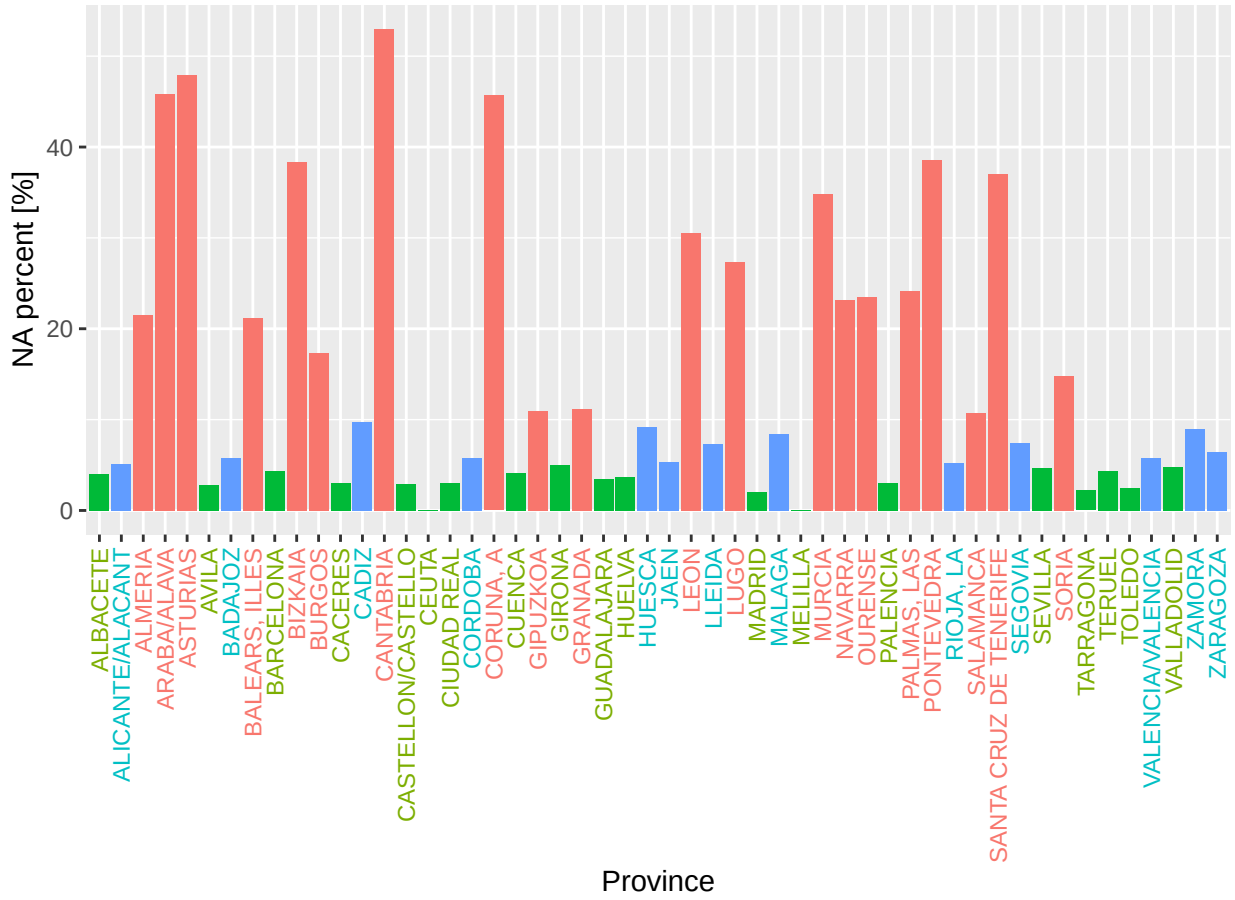
Zeros i elements buits

Com a primer pas en la neteja de dades, es procedeix a eliminar tots aquells registres del *dataset* resultant on el camp *Capture_date* sigui *NA*. Aquests seran municipis que apareixen en el cens de població, però no tenen benzinera. Aquests han aparegut en el *dataset* quan s'ha realitzat l'operació de combinació plena o *FULL JOINT* en el pas anterior.

```
data<-data[!is.na(data$Capture_date),]
```

Actualment el *dataset* conté 21329 registres, dels quals 2840 són registres dels quals no se'n coneix el cens. Això representa un 13% del total de registres.

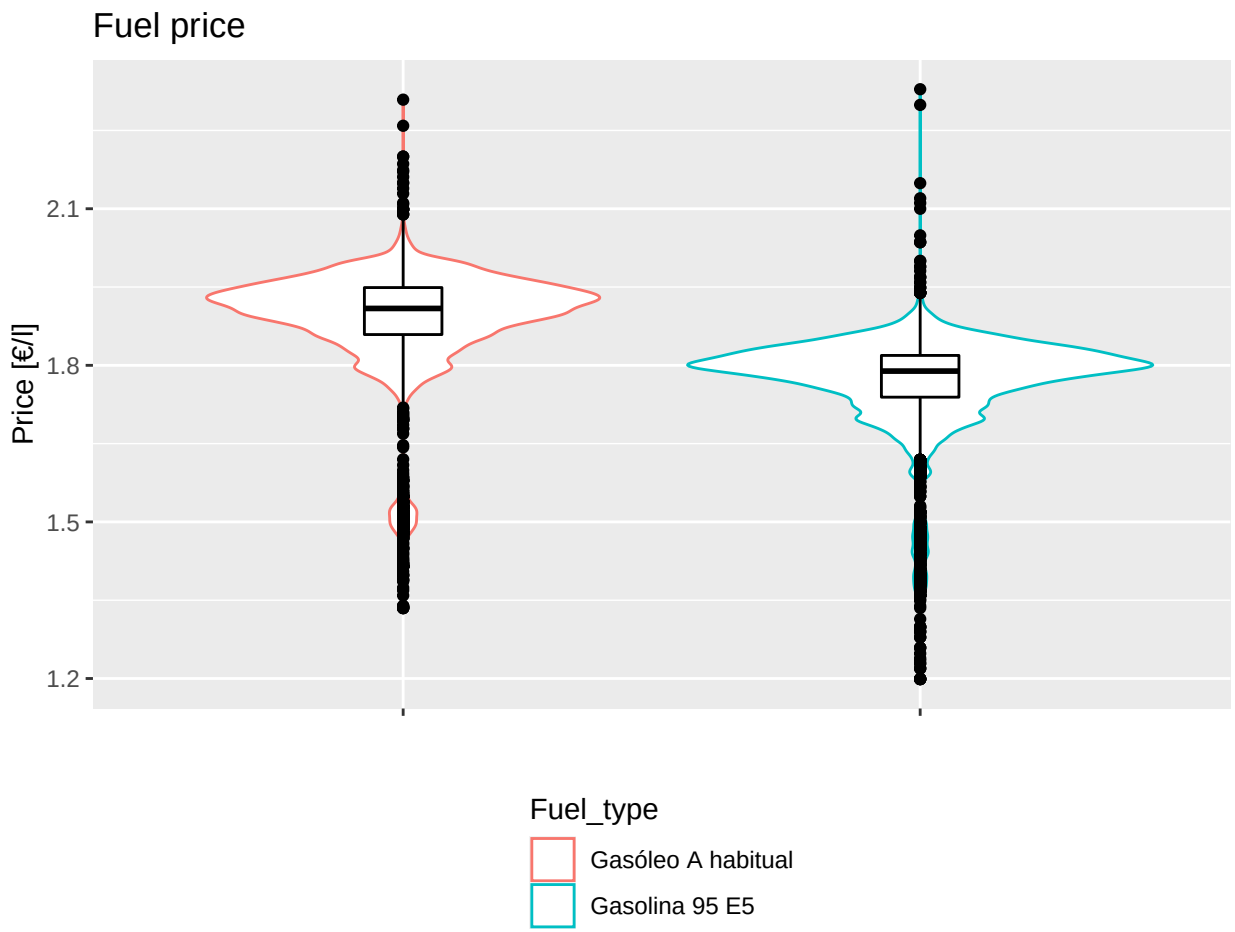
Quan es procedeix a realitzar aquest mateix estudi per a cada una de les províncies amb l'objectiu d'identificar aquelles que tinguin una representació més pobre s'obté el següent gràfic en el que es marquen en verd, aquelles províncies amb un percentatge de NA inferior a 5%, en blau les províncies amb un percentatge entre 5 i 10% i en vermell aquelles províncies amb un percentatge de NA superior al 10%.



Conseqüentment, per a estudis que no considerin dades de cens per municipis, es pot utilitzar el *dataset* complert ja que aquest prové de l'operació de *scraping* i no conté zeros o valors nuls. Tanmateix, quan l'objectiu de l'estudi requereixi considerar informació del cens; per raons de representativitat, es recomana limitar l'estudi a les províncies anteriorment identificades en verd i en blau.

Valors extrems

Per tal d'identificar els valors extrems es genera un *violin plot* amb un *boxplot* al interior per cada tipus de combustible. Aquest tipus de gràfic ampliat ens permet, per una banda analitzar els *outliers* a través de la visualització *boxplot*, alhora que ens permet conèixer la distribució de la població a través de la visualització *violin*. Així doncs permet, d'un cop de vista, veure si la informació aporta el *boxplot* es consistent amb la distribució de les dades.



Observant l'existència de valors extrems s'aprofundeix en l'anàlisi, inicialment obtenint els valors característics per a cada un dels *boxplots*

	Min	Q1	Med	Q3	Max
Gasóleo A habitual	1.727	1.859	1.909	1.949	2.079
Gasolina 95 E5	1.622	1.739	1.789	1.819	1.933

S'observa que la diferència de preus entre les medianes de la població *Gasóleo A habitual* i *Gasolina 95 E5* és de 0.12€/l, sent el combustible *Gasóleo A habitual* el més car.

Valors extrems superiors

S'obtenen els registres que son valors extrems superiors tant pel cas del combustible *Gasóleo A habitual* com pel *Gasolina 95 E5*.

```
st<-boxplot.stats(data[data$Fuel_type=="Gasóleo A habitual","Price"])
DiesUP<-data[data$Fuel_type =="Gasóleo A habitual" & data$Price>st$stats[5],] %>%
  group_by(Province, Brand) %>%
  as.data.frame()

st<-boxplot.stats(data[data$Fuel_type=="Gasolina 95 E5","Price"])
GasUP<-data[data$Fuel_type =="Gasolina 95 E5" & data$Price>st$stats[5],] %>%
```

```
  group_by(Province, Brand) %>%
  as.data.frame()
```

Combinant la informació anterior, s'obté les dades de les estacions de servei on els dos combustibles es consideren *outliers*. Fet que indica que la estació de servei en general té uns preus més cars que la mitjana.

```
UPs<-merge(DiesUP, GasUP, by=c("Province", "City", "Capture_date", "Address", "Brand"),
           all = FALSE)
UPs<-UPs[c("Province", "City", "Address", "Brand", "Fuel_type.x", "Price.x",
           "Fuel_type.y", "Price.y")] %>%
  arrange(Province, City, Brand, Address)
```

Finalment es comprova quines estacions de servei (de les marcades com a cares) no mantenen la diferència de medianes de preus de combustibles obtinguda anteriorment. Per tal de ser una mica més flexibles, es considerarà com a límit el 80% de la diferència anterior. Per tant es seleccionen els registres on la diferència de preu entre els dos combustibles sigui inferior a 0.1€/l.

Els registres d'aquestes estacions de servei s'eliminen de l'estudi per a ser *outliers* i no mostrar prou consistència en els seus valors, que ens permetin garantir que els preus no continguin errades.

```
UPs<-UPs[UPs$Price.x-UPs$Price.y < round(0.8*(stt[1,3]-stt[2,3]),2),]
data<-anti_join(data,UPs, by=c("Province", "City", "Address", "Brand"))
```