Pràctica 2. Neteja i Validació de Dades

1.- Descripció del *dataset*

El conjunt de dades seleccionat conté informació referent a les vendes d'immobles a la ciutat de Melbourne durant els anys 2016, 2017 i 2018. Aquestes dades s'han obtingut del *site* de Kaggle https://www.kaggle.com/anthonypino/melbourne-housing-market

En aquest dataset obtenim les dades de dos fitxers en format CSV:

1. Melbourne_house_prices_LESS.csv
2. Melbourne_housing_FULL.csv

El primer d'ells conté més registres (52964), però menys columnes d'informació (12).
El segon és més complet en quant a dimensions (21), però conté menys registres (34857) .
Es aquest segon fitxer el que escollim per a fer l'estudi, ja que considerem que conté un volum de dades suficient i el nombre d'atributs és més complet.

Detall de les variables del fitxer Melbourne_housing_FULL.csv:

- **Suburb**: barri de l'immoble
- **Address**: adreça de l'immoble
- **Rooms**: número d'habitacions
- **Type**:
  - br - bedroom(s)
  - h - house, cottage, villa, semi, terrace
  - u - unit, duplex
  - t - townhouse

    - dev site - development site
    - o res - other residential
- **Price**: preu de venda en dòlars australians (AUD). A data 02/06/18 un AUD equival a 0.65€
- **Method**: mètode de venda
    - S - property sold
    - SP - property sold prior
    - PI - property passed in
    - PN - sold prior not disclosed
    - SN - sold not disclosed
    - NB - no bid
    - VB - vendor bid
    - W - withdrawn prior to auction
    - SA - sold after auction
    - SS - sold after auction price not disclosed
    - N/A - price or highest bid not available
- **SellerG**: codi de l'agent de venda
- **Date**: data de la venda
- **Distance**: distància al districte financer de Melbourne en quilòmetres
- **PostCode**: codi postal de l'immoble
- **Bedroom2** : número d'habitacions amb llit
- **Bathroom**: número de banys
- **Car**: número de places de garatge
- **Landsize**: tamany del terreny en metres
- **BuildingArea**: tamany de l'immoble en metres
- **YearBuilt**: any de construcció de l'immoble
- **CouncilArea**: ajuntament al qual pertany el terreny
- **Lattitude**: latitud de l'immoble
- **Longtitude**: longitud de l'immoble
- **RegionName**: regió (West, North West, North, North east ...etc)
- **Propertycount**: número total d'immobles registrats al barri on està l'immoble

Aquest dataset s'ha publicat amb una llicència Creative Commons **CC BY-NC-SA 4.0**. Per tant podem:

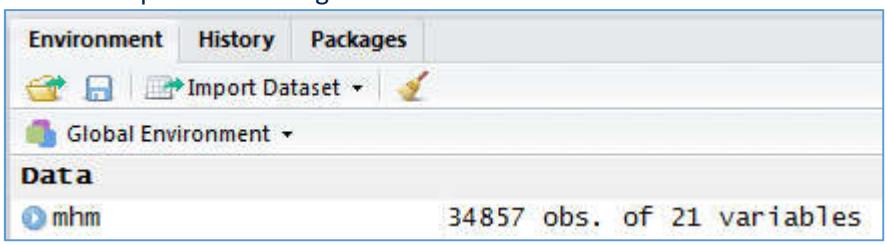
- Compartir – copiar i redistribuir el conjunt de dades en qualsevol medi o format
- Adaptar – barrejar, transformar i construir sobre el conjunt de dades

2.- Selecció i Neteja de dades

En primer lloc es realitza la lectura de les dades a l'entorn R d'anàlisi. En el nostre cas a l'entorn **RStudio**:


```
> #Lectura de dades
> mhm <- read.csv("Melbourne_housing_FULL.csv")
```

Observem que s'han carregat totes les dades:



Veiem la capçalera de dades:

```
> head(mhm)
```

```
      Suburb           Address Rooms Type    Price Method SellerG      Date Distance Postcode Bedroom2 Bathroom
1 Abbotsford      68 Studley St     2    h       NA     SS  Jellis 3/09/2016      2.5     3067        2        1
2 Abbotsford       85 Turner St     2    h  1480000      S  Biggin 3/12/2016      2.5     3067        2        1
3 Abbotsford    25 Bloomburg St     2    h  1035000      S  Biggin 4/02/2016      2.5     3067        2        1
4 Abbotsford 18/659 Victoria St     3    u       NA     VB  Rounds 4/02/2016      2.5     3067        3        2
5 Abbotsford       5 Charles St     3    h  1465000     SP  Biggin 4/03/2017      2.5     3067        3        2
6 Abbotsford  40 Federation La      3    h   850000     PI  Biggin 4/03/2017      2.5     3067        3        2
  Car Landsize BuildingArea YearBuilt       CouncilArea Lattitude Longtitude             Regionname Propertycount
1   1      126           NA        NA Yarra City Council  -37.8014   144.9958 Northern Metropolitan          4019
2   1      202           NA        NA Yarra City Council  -37.7996   144.9984 Northern Metropolitan          4019
3   0      156           79      1900 Yarra City Council  -37.8079   144.9934 Northern Metropolitan          4019
4   1        0           NA        NA Yarra City Council  -37.8114   145.0116 Northern Metropolitan          4019
5   0      134          150      1900 Yarra City Council  -37.8093   144.9944 Northern Metropolitan          4019
6   1       94           NA        NA Yarra City Council  -37.7969   144.9969 Northern Metropolitan          4019
```

I ara analitzem el tipus de dades d'aquestes variables:

```
> sapply(mhm, function(x) class(x))
```

```
      Suburb      Address        Rooms         Type        Price       Method      SellerG         Date
    "factor"     "factor"    "integer"     "factor"    "integer"     "factor"     "factor"     "factor"
    Distance     Postcode     Bedroom2     Bathroom          Car     Landsize BuildingArea    YearBuilt
    "factor"     "factor"    "integer"    "integer"    "integer"    "integer"    "numeric"    "integer"
 CouncilArea    Lattitude   Longtitude  Regionname Propertycount
    "factor"    "numeric"    "numeric"    "factor"     "factor"
```

Observem que tenim:

- 11 variables de tipus **factor** (categòriques)
- 7 variables de tipus **integer**
- 3 variables de tipus **numeric**

```
> str(mhm)
```

```
'data.frame':   34857 obs. of  21 variables:
 $ Suburb       : Factor w/ 351 levels "Abbotsford","Aberfeldie",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Address      : Factor w/ 34009 levels "1 Abercrombie St",..: 29459 32513 15390 9769 25129 23201 27095 8333 26797 33
959 ...
 $ Rooms        : int  2 2 2 3 3 3 4 4 2 2 ...
 $ Type         : Factor w/ 3 levels "h","t","u": 1 1 1 3 1 1 1 1 1 1 ...
 $ Price        : int  NA 1480000 1035000 NA 1465000 850000 1600000 NA NA NA ...
 $ Method       : Factor w/ 9 levels "PI","PN","S",..: 7 3 3 8 6 1 8 5 3 3 ...
 $ SellerG      : Factor w/ 388 levels "@Realty","A",..: 171 34 34 313 34 34 246 246 34 76 ...
 $ Date         : Factor w/ 78 levels "1/07/2017","10/02/2018",..: 59 61 64 64 65 65 66 70 70 70 ...
 $ Distance     : Factor w/ 216 levels "#N/A","0","0.7",..: 82 82 82 82 82 82 82 82 82 82 ...
 $ Postcode     : Factor w/ 212 levels "#N/A","3000",..: 55 55 55 55 55 55 55 55 55 55 ...
 $ Bedroom2     : int  2 2 2 3 3 3 3 3 4 3 ...
 $ Bathroom     : int  1 1 1 2 2 2 1 2 1 2 ...
 $ Car          : int  1 1 0 1 0 1 2 2 2 1 ...
 $ Landsize     : int  126 202 156 0 134 94 120 400 201 202 ...
 $ BuildingArea : num  NA NA 79 NA 150 NA 142 220 NA NA ...
 $ YearBuilt    : int  NA NA 1900 NA 1900 NA 2014 2006 1900 1900 ...
 $ CouncilArea  : Factor w/ 34 levels "#N/A","Banyule City Council",..: 33 33 33 33 33 33 33 33 33 33 ...
 $ Lattitude    : num  -37.8 -37.8 -37.8 -37.8 -37.8 ...
 $ Longtitude   : num  145 145 145 145 145 ...
 $ Regionname   : Factor w/ 9 levels "#N/A","Eastern Metropolitan",..: 4 4 4 4 4 4 4 4 4 4 ...
 $ Propertycount: Factor w/ 343 levels "#N/A","1008",..: 191 191 191 191 191 191 191 191 191 191 ...
```

Veiem que tenim alguns atributs de tipus *factor* que contenen valors "#N/A". Per tal de tractar-los correctament els convertim a valors desconeguts (NA's):

```
> levels(mhm$Distance) <- sub("#N/A", NA, levels(mhm$Distance))
> levels(mhm$Postcode) <- sub("#N/A", NA, levels(mhm$Postcode))
> levels(mhm$CouncilArea) <- sub("#N/A", NA, levels(mhm$CouncilArea))
> levels(mhm$Regionname) <- sub("#N/A", NA, levels(mhm$Regionname))
> levels(mhm$Propertycount) <- sub("#N/A", NA, levels(mhm$Propertycount))
```

Comprobem:

```
> sum(is.na(mhm$Distance))
[1] 1
> sum(is.na(mhm$Postcode))
[1] 1
> sum(is.na(mhm$CouncilArea))
[1] 3
> sum(is.na(mhm$Regionname))
[1] 3
> sum(is.na(mhm$Propertycount))
[1] 3
```

L'atribut **Distance** de tipus *factor* el convertim a *numeric*:

```
> mhm$Distance <- as.numeric(as.character(mhm$Distance))
```

El mateix fem amb l'atribut **Propertycount**:

```
> mhm$Propertycount <- as.numeric(as.character(mhm$Propertycount))
```

Per últim convertim l'atribut Date a tipus date. El format d'aquest atribut és "dd/mm/yyyy":

```
> mhm$Date <- as.Date(mhm$Date, format = "%d/%m/%Y")
```

Pel nostre estudi no considerem necessaris els següents atributs:

- **Address**: volem analitzar els preus dels immobles a nivell de barri, no pas de carrers
- **SellerG**: no considerem que el venedor tingui influència en el preu de venda de l'immoble

```
> #Eliminem atribut Address
```

```
> mhm <- mhm[,-2]
> #Eliminem atribut SellerG
> mhm <- mhm[,-6]
```

Finalment reubiquem la columna **Price** a la última posició del *Dataframe*:

```
> col_price <- grep("Price", names(mhm))
> mhm <- mhm[, c((1:ncol(mhm))[-col_price], col_price)]
```

Anem ara a consultar el número de registres que no informen de **Price**:

```
> sapply(mhm, function(x) sum(is.na(x)))
```

Suburb	Rooms	Type	Price	Method	Date	Distance	Postcode
0	0	0	7610	0	0	0	0
Bedroom2	Bathroom	Car	Landsize	BuildingArea	YearBuilt	CouncilArea	Lattitude
8217	8226	8728	11810	21115	19306	0	7976
Longtitude	Regionname	Propertycount					
7976	0	0					

Observem que tenim **7610** immobles dels quals no tenim informació sobre el preu de venda. Com que aquest és l'atribut objectiu eliminarem aquests registres:

```
> mhm <- mhm[!is.na(mhm$Price),]
> dim(mhm)
[1] 27247    19
```

2.1 Tractament de valors nuls

No considerem la opció d'eliminar els registres en els que algun atribut té valor nul (NA), ja que estaríem perdent molta informació que pot ser rellevant de cara al nostre estudi.

Per tal de donar solució a aquesta situació aplicarem un valor a cadascun d'aquests atributs nuls basat en l'algoritme dels k-veïns més propers.

Per tal d'aplicar aquest algoritme s'ha de carregar la llibreria **VIM**:

```
> load(VIM)
```

Apliquem l'algoritme als atributs:

```
> mhm$Bedroom2 <- kNN(mhm)$Bedroom2
> mhm$Bathroom <- kNN(mhm)$Bathroom
> mhm$Car <- kNN(mhm)$Car
> mhm$Postcode <- kNN(mhm)$Postcode
> mhm$Regionname <- kNN(mhm)$Regionname
> mhm$Landsize <- kNN(mhm)$Landsize
> mhm$BuildingArea <- kNN(mhm)$BuildingArea
> mhm$YearBuilt <- kNN(mhm)$YearBuilt
> mhm$CouncilArea <- kNN(mhm)$CouncilArea
> mhm$Lattitude <- kNN(mhm)$Lattitude
> mhm$Longtitude <- kNN(mhm)$Longtitude
> mhm$Propertycount <- kNN(mhm)$Propertycount
```

Ara ja tenim tots els atributs amb tots els seus valors informats:

```
> sapply(mhm, function(x) sum(is.na(x)))
```

```
      Suburb        Rooms         Type        Price       Method         Date     Distance     Postcode     Bedroom2
           0            0            0            0            0            0            0            0            0
    Bathroom          Car     Landsize BuildingArea    YearBuilt   CouncilArea     Lattitude    Longtitude   Regionname
           0            0            0            0            0            0            0            0            0
Propertycount
           0
```

I el *Dataframe* de la següent manera:

```
> str(mhm)
'data.frame':   27247 obs. of  19 variables:
 $ Suburb        : Factor w/ 351 levels "Abbotsford","Aberfeldie",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Rooms         : num  2 2 3 3 4 2 3 2 3 2 ...
 $ Type          : Factor w/ 3 levels "h","t","u": 1 1 1 1 1 1 1 1 1 2 ...
 $ Method        : Factor w/ 9 levels "PI","PN","S",..: 3 3 6 1 8 3 3 3 3 3 ...
 $ Date          : Date, format: "2016-12-03" "2016-02-04" "2017-03-04" "2017-03-04" ...
 $ Distance      : num  2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5 ...
 $ Postcode      : Factor w/ 211 levels "3000","3002",..: 54 54 54 54 54 54 54 54 54 54 ...
 $ Bedroom2      : num  2 2 3 3 3 2 4 2 3 2 ...
 $ Bathroom      : num  1 1 2 2 1 1 2 1 1 1 ...
 $ Car           : num  1 0 0 1 2 0 0 2 2 1 ...
 $ Landsize      : int  202 156 134 94 120 181 245 256 263 321 ...
 $ BuildingArea  : num  87 79 150 112 142 ...
 $ YearBuilt     : int  1995 1900 1900 1900 2014 1890 1910 1890 2014 1960 ...
 $ CouncilArea   : Factor w/ 33 levels "Banyule City Council",..: 32 32 32 32 32 32 32 32 32 32 ...
 $ Lattitude     : num  -37.8 -37.8 -37.8 -37.8 -37.8 ...
 $ Longtitude    : num  145 145 145 145 145 ...
 $ Regionname    : Factor w/ 8 levels "Eastern Metropolitan",..: 3 3 3 3 3 3 3 3 3 3 ...
 $ Propertycount: num  4019 4019 4019 4019 4019 ...
 $ Price         : int  1480000 1035000 1465000 850000 1600000 941000 1876000 1636000 1000000 745000 ...
```

2.2 Tractament d'outliers

```
> summary(mhm)
          Suburb           Rooms           Type           Price             Method          Date           Distance         Postcode       Bedroom2     
 Reservoir     :  727   Min.   : 1.000   h:18472   Min.   :   85000   S      :17515   28/10/2017:  879   11.2   : 1112   3073   :  727   Min.   : 0.00  
 Bentleigh East:  493   1st Qu.: 2.000   t: 2866   1st Qu.:  635000   SP     : 3603   17/03/2018:  753   13.8   :  558   3046   :  545   1st Qu.: 2.00  
 Richmond      :  439   Median : 3.000   u: 5909   Median :  870000   PI     : 3255   24/02/2018:  723   10.5   :  526   3020   :  544   Median : 3.00  
 Preston       :  415   Mean   : 2.992             Mean   : 1050173   VB     : 2684   9/12/2017 :  723   5.2    :  486   3165   :  493   Mean   : 3.02  
 Brunswick     :  387   3rd Qu.: 4.000             3rd Qu.: 1295000   SA     :  190   25/11/2017:  682   7.8    :  461   3121   :  489   3rd Qu.: 4.00  
 Essendon      :  361   Max.   :16.000             Max.   :11200000   PN     :    0   18/11/2017:  681   9.2    :  459   3040   :  466   Max.   :20.00  
 (Other)       :24425                                                 (Other):    0   (Other)   :22806   (Other):23645   (Other):23983                  
    Bathroom          Car           Landsize       BuildingArea       YearBuilt                    CouncilArea     Lattitude      Longtitude    
 Min.   :0.000   Min.   : 0.000   Min.   :     0.0   Min.   :    0.0   Min.   :1196   Boroondara City Council : 2520   Min.   :-38.19   Min.   :144.4  
 1st Qu.:1.000   1st Qu.: 1.000   1st Qu.:   263.0   1st Qu.:   90.0   1st Qu.:1955   Darebin City Council    : 2349   1st Qu.:-37.85   1st Qu.:144.9  
 Median :1.000   Median : 2.000   Median :   334.0   Median :  133.0   Median :1975   Moreland City Council   : 1790   Median :-37.80   Median :145.0  
 Mean   :1.514   Mean   : 1.695   Mean   :   508.8   Mean   :  142.2   Mean   :1974   Glen Eira City Council  : 1643   Mean   :-37.80   Mean   :145.0  
 3rd Qu.:2.000   3rd Qu.: 2.000   3rd Qu.:   615.0   3rd Qu.:  169.0   3rd Qu.:2006   Moonee Valley City Council: 1584   3rd Qu.:-37.74   3rd Qu.:145.1  
 Max.   :9.000   Max.   :18.000   Max.   :433014.0   Max.   :44515.0   Max.   :2019   Melbourne City Council  : 1502   Max.   :-37.40   Max.   :145.5  
                                                                                     (Other)                 :15859                                    
                   Regionname   Propertycount  
 Southern Metropolitan     :8524   21650  :  727  
 Northern Metropolitan     :7864   8870   :  609  
 Western Metropolitan      :5815   10969  :  493  
 Eastern Metropolitan      :3272   14949  :  439  
 South-Eastern Metropolitan:1341   14577  :  415  
 Eastern Victoria          : 166   11918  :  387  
 (Other)                   : 265   (Other):24177
```

Observem que **Lattitude** i **Longtitude** no tenen *outliers*:

```
   Lattitude        Longtitude    
 Min.   :-38.19   Min.   :144.4  
 1st Qu.:-37.85   1st Qu.:144.9  
 Median :-37.80   Median :145.0  
 Mean   :-37.80   Mean   :145.0  
 3rd Qu.:-37.74   3rd Qu.:145.1  
 Max.   :-37.40   Max.   :145.5
```

YearBuilt: observem que el valor 2019 és incorrecte, ja que l'any actual és 2018:

```
   YearBuilt   
 Min.   :1196  
 1st Qu.:1955  
 Median :1975  
 Mean   :1974  
 3rd Qu.:2006  
 Max.   :2019
```

```
> sum(mhm$YearBuilt[mhm$YearBuilt==2019])
[1] 2019
```

En aquest cas el substituirem pel valor de *Median*:

```
> mhm$YearBuilt[mhm$YearBuilt==2019] <- median((mhm$YearBuilt))
```

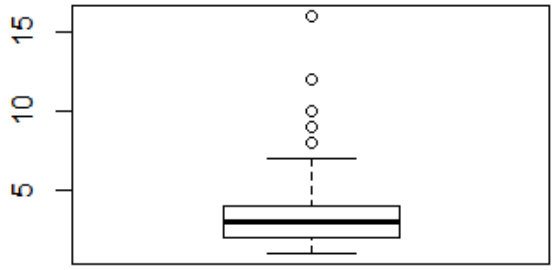
Ho comprovem:

```
> summary(mhm$YearBuilt)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   1196    1955    1975    1974    2006    2018
```

La resta *d'outliers* els visualitzarem amb **boxplots**:

Rooms:

```
> boxplot(mhm$Rooms)
```



No sembla molt normal tenir immobles amb més de 8 habitacions. Hem vist anteriorment que el conjunt de dades que estem tractant conté com a tipus d'immobles:

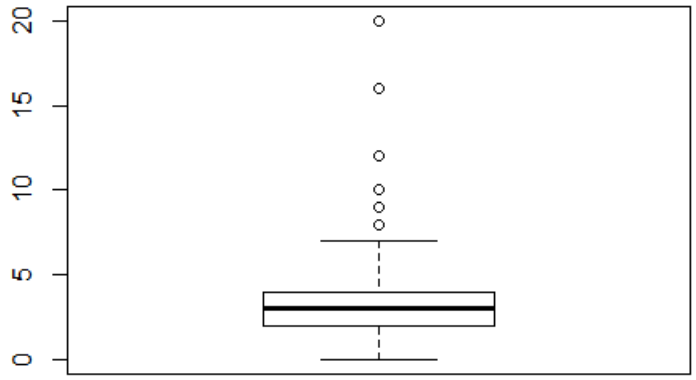
- h: house, cottage, villa, semi, terrace
- t: townhouse
- u: unit, duplex

Per tant, optarem per assignar als immobles amb més de 8 habitacions el valor de 8:

```
> sum(mhm$Rooms>8)
[1] 9
> mhm$Rooms[mhm$Rooms>8] <- 8
```

Bedroom2:

```
> boxplot(mhm$Bedroom2)
```

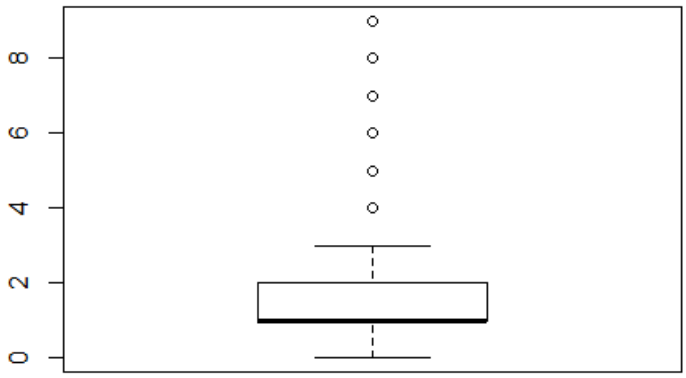


Seguint el mateix criteri comentat anteriorment, optarem per assignar als immobles amb més de 8 habitacions amb llit el valor de 8:

```
> sum(mhm$Bedroom2>8)
[1] 10
> mhm$Bedroom2[mhm$Bedroom2>8] <- 8
```

Bathroom:

```
> boxplot(mhm$Bathroom)
```

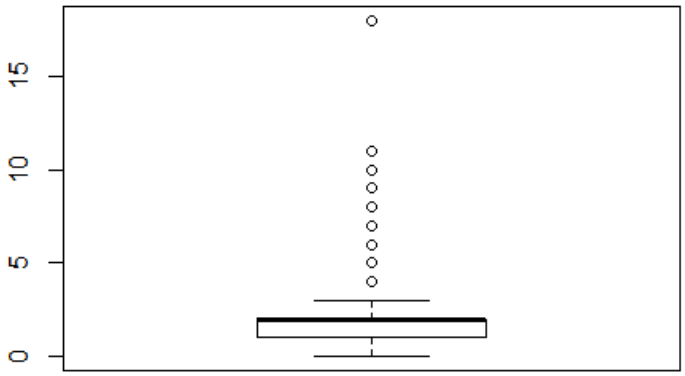


Optarem per assignar als immobles amb més de 5 quarts de bany el valor de 5:

```
> sum(mhm$Bathroom>5)
[1] 16
> mhm$Bathroom[mhm$Bathroom>5] <- 5
```

Car:

```
> boxplot(mhm$Car)
```

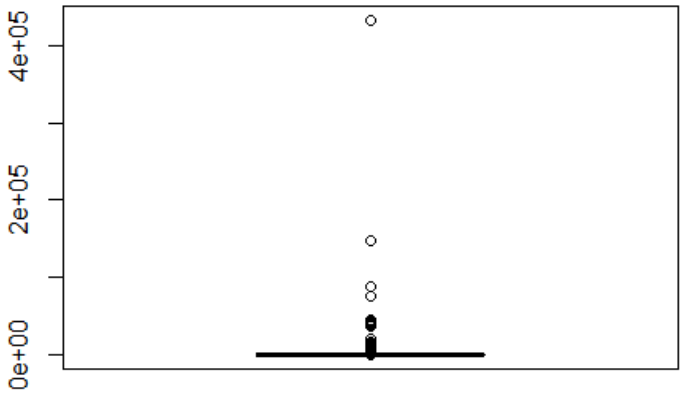


Optarem per assignar als immobles amb més de 5 garatges el valor de 5:

```
> sum(mhm$Car>5)
[1] 151
> mhm$Car[mhm$Car>5] <- 5
```

Landsize:

```
> boxplot(mhm$Landsize)
```



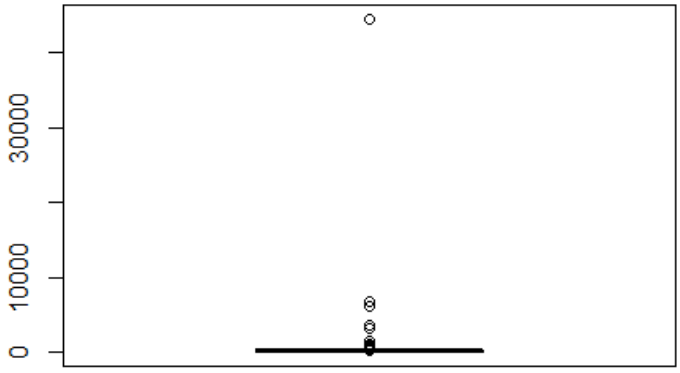

```
> sum(mhm$Landsize>50000)
```

```
[1] 5
```

Tenim un total de 5 habitatges amb més de 50000metres de superfície. Aquests valors els deixarem així, per que pot ser el cas de immobles que estan en llocs amb molt terreny.

BuildingArea:

```
> boxplot(mhm$BuildingArea)
```




```
> sum(mhm$BuildingArea>5000)
[1] 3
```

Com en el cas de l'atribut anterior, poden tenir alguns terrenys molt grans. Per tant deixem aquests outliers com estan.

3.- Anàlisi de les dades

3.1 Atributs categòrics

Observem que els atributs **Suburb** i **Postcode** estan relacionats, en el sentit que cada barri té un codi postal assignat:

```
> suburb_pcode <- mhm[,c('Suburb','Postcode')]
> suburb_pcode <- unique(suburb_pcode)
> suburb_pcode
```

	Suburb	Postcode
1	Abbotsford	3067
67	Airport West	3042
134	Albert Park	3206
195	Alphington	3078
231	Altona	3018
284	Altona North	3025
365	Armadale	3143
485	Ascot Vale	3032
598	Ashburton	3147
680	Ashwood	3147
748	Avondale Heights	3034
824	Balaclava	3183
847	Balwyn	3103
1011	Balwyn North	3104
1210	Bentleigh	3204
1356	Bentleigh East	3165
1629	Box Hill	3128
1688	Braybrook	3019

Veiem que hi han diferents barris que pertanyen al mateix codi postal, com per exemple Ashburton i Ashwood, que pertanyen al codi postal 3147.

De la mateixa manera podríem pensar que els atributs **Postcode** i **CouncilArea** estan relacionats, en el sentit que cada codi postal està assignat a un ajuntament:

```
> pcode_council <- mhm[,c('Postcode','CouncilArea')]
> pcode_council <- unique(pcode_council)
> head(pcode_council,20)
```

	Postcode	CouncilArea
1	3067	Yarra City Council
67	3042	Moonee Valley City Council
134	3206	Port Phillip City Council
195	3078	Darebin City Council
231	3018	Hobsons Bay City Council
284	3025	Hobsons Bay City Council
365	3143	Stonnington City Council
485	3032	Moonee Valley City Council
598	3147	Boroondara City Council
680	3147	Monash City Council
748	3034	Moonee Valley City Council
824	3183	Port Phillip City Council
847	3103	Boroondara City Council
1011	3104	Boroondara City Council
1210	3204	Glen Eira City Council
1356	3165	Glen Eira City Council
1629	3128	Whitehorse City Council
1688	3019	Maribyrnong City Council
1733	3186	Bayside City Council
1946	3187	Bayside City Council
2140	3056	Moreland City Council
2332	3055	Moreland City Council
2433	3105	Manningham City Council

Però veiem que no. Trobem algun cas en que un mateix codi postal pertany a diferents ajuntaments.

3.2 Normalització de les dades

A continuació estudiarem si les variables quantitatives del *dataset* estan normalitzades.

Farem servir la prova de normalitat d'**Anderson Darling**. Si el valor de p-value és inferior al nivell de significació prefixat α = 0, 05 llavors la variable en qüestió no segueix una distribució normal.

El primer pas serà carregar la llibreria **nortest** de R, la qual implementa entre d'altres tests de normalització el d'Anderson Darling (ad.test).

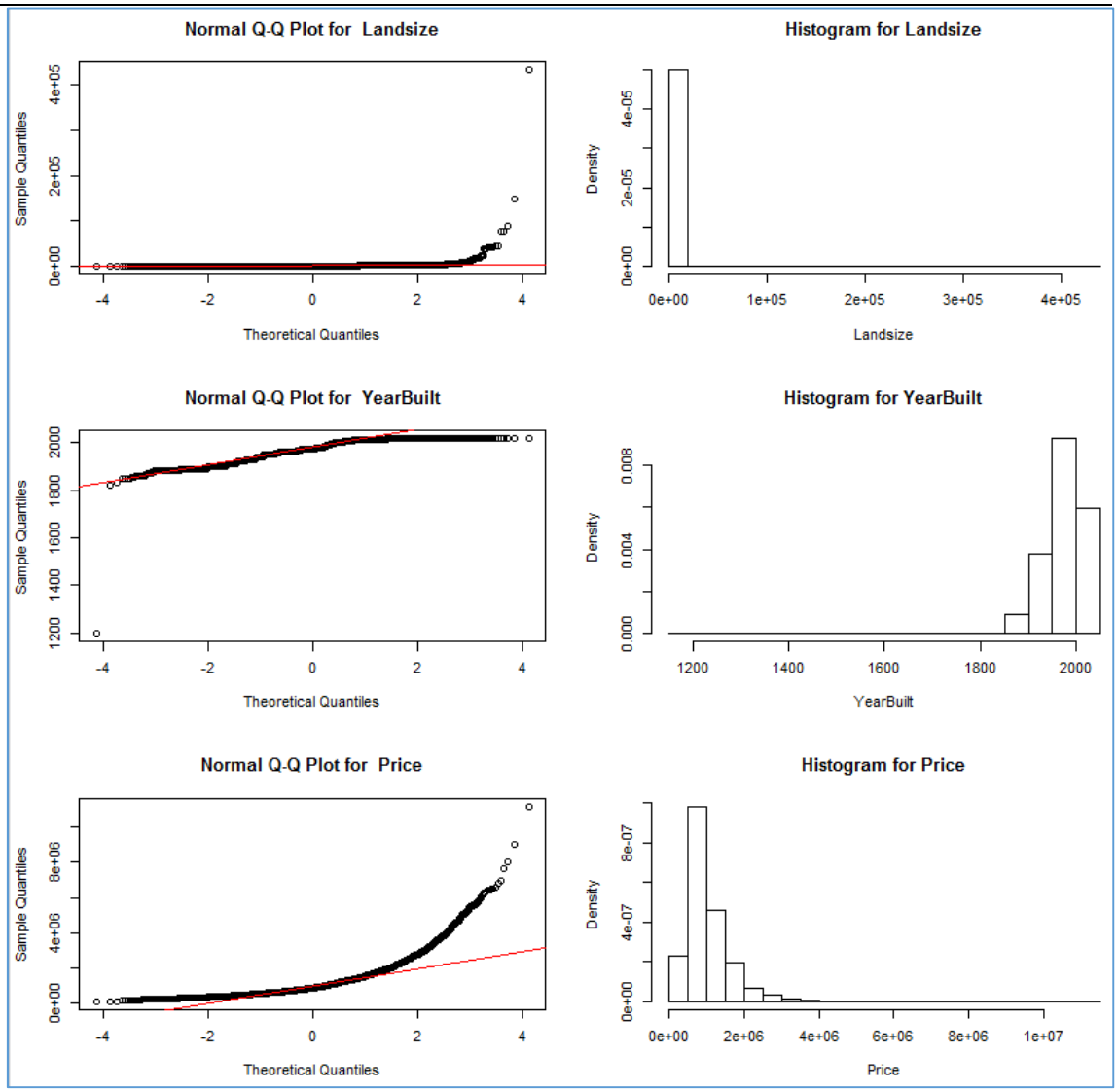
```
> library(nortest)
> alpha = 0.05
> col.names = colnames(mhm)
> for (i in 1:ncol(mhm)) {
+     if (is.integer(mhm[,i]) | is.numeric(mhm[,i])) {
+         p_val = ad.test(mhm[,i])$p.value
+         if (p_val < alpha) {
+             cat("-> ",col.names[i])
+             cat("\n")
```

```
+       }
+    }
+ }
-> Rooms
-> Date
-> Postcode
-> Bedroom2
-> Bathroom
-> Car
-> Landsize
-> BuildingArea
-> CouncilArea
-> Lattitude
-> Regionname
-> Propertycount
```

Observem que tenim un total de 12 variables que no segueixen una distribució normal.

A continuació mostrem les gràfiques de Quantile-Quantile Plot per variable de tipus enter i el respectiu Histograma.

```
par(mfrow=c(3,2))
for(i in 1:ncol(mhm)) {
 if (is.integer(mhm[,i])) {
          qqnorm(mhm[,i],main = paste("Normal Q-Q Plot for ",colnames(mhm)[i]))
          qqline(mhm[,i],col="red")
          hist(mhm[,i], main=paste("Histogram for", colnames(mhm)[i]),
                  xlab=colnames(mhm)[i], freq = FALSE)
 }
}
```



3.3 Variància de les dades

Per tal d'analitzar la variància de les dades farem servir el test de **Fligner-Killeen**, el qual es basa en la mediana. Escollim aquest test ja que no es compleix la condició de normalitat en les observacions.

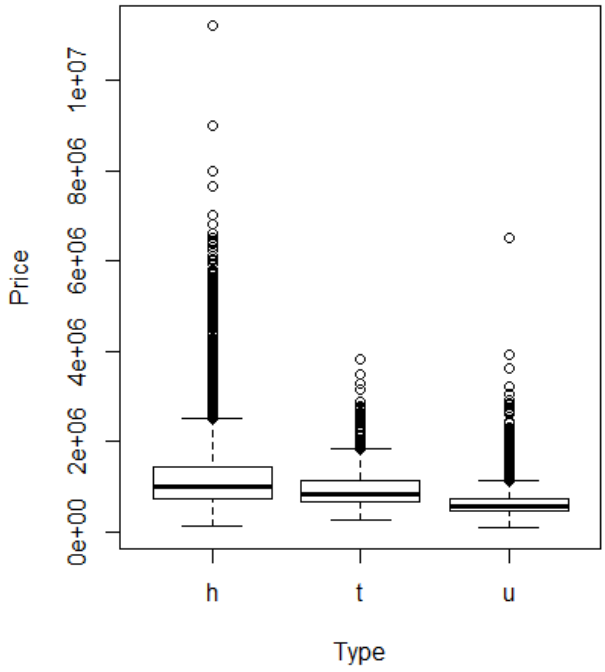
Mirem la homogeneïtat dels grups formats pel tipus d'immoble (atribut **Type**):

```
> fligner.test(Price ~ Type, data = mhm)
```

```
        Fligner-Killeen test of homogeneity of variances

data:  Price by Type
Fligner-Killeen:med chi-squared = 3030.2, df = 2, p-value < 2.2e-16
```

```
> plot(Price ~ Type, data = mhm)
```



Ara apliquem el mateix test al barri de l'immoble (atribut **Suburb**):

```
> fligner.test(Price ~ Suburb, data = mhm)
```

```
        Fligner-Killeen test of homogeneity of variances

data:  Price by Suburb
Fligner-Killeen:med chi-squared = 9019.7, df = 344, p-value < 2.2e-16
```

I per últim apliquem el test per nom de la regió de l'immoble (atribut **Regionname**):

```
> fligner.test(Price ~ Regionname, data = mhm)
```

```
        Fligner-Killeen test of homogeneity of variances

data:  Price by Regionname
Fligner-Killeen:med chi-squared = 4270.3, df = 7, p-value < 2.2e-16
```

Observem que en els 3 test realitzats el valor de **p-value** és molt inferior a 0.05. Per tant acceptem la hipòtesi que les variàncies de les mostres no són homogènies.

4.- Proves estadístiques

Quines són les variables que més influeixen en el preu de l'immoble?

Per tal de comprovar-ho anem a realitzar una anàlisi de la correlació lineal entre les diferents variables quantitatives del *dataset*. Es farà servir la correlació d'**Spearman**, ja que les variables no ténen una distribució normal.

Creem una matriu de correlació, amb les columnes 'estimate' i 'p-value' i calculem els coeficients per cadascuna de les variables (exceptuant **Price**):

```
> corr_matrix <- matrix(nc = 2, nr = 0)
> colnames(corr_matrix) <- c("estimate", "p-value")
```

```
> for (i in 1:(ncol(mhm) - 1)) {
+   if (is.integer(mhm[,i]) | is.numeric(mhm[,i])) {
+     spearman_test = cor.test(mhm[,i], mhm[,length(mhm)], 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(mhm)[i]
+   }
+ }
```

```
Warning messages:
1: In cor.test.default(mhm[, i], mhm[, length(mhm)], method = "spearman") :
  Cannot compute exact p-value with ties
2: In cor.test.default(mhm[, i], mhm[, length(mhm)], method = "spearman") :
  Cannot compute exact p-value with ties
3: In cor.test.default(mhm[, i], mhm[, length(mhm)], method = "spearman") :
  Cannot compute exact p-value with ties
4: In cor.test.default(mhm[, i], mhm[, length(mhm)], method = "spearman") :
  Cannot compute exact p-value with ties
5: In cor.test.default(mhm[, i], mhm[, length(mhm)], method = "spearman") :
  Cannot compute exact p-value with ties
6: In cor.test.default(mhm[, i], mhm[, length(mhm)], method = "spearman") :
  Cannot compute exact p-value with ties
7: In cor.test.default(mhm[, i], mhm[, length(mhm)], method = "spearman") :
  Cannot compute exact p-value with ties
8: In cor.test.default(mhm[, i], mhm[, length(mhm)], method = "spearman") :
  Cannot compute exact p-value with ties
9: In cor.test.default(mhm[, i], mhm[, length(mhm)], method = "spearman") :
  Cannot compute exact p-value with ties
```

Els missatges d'error no invaliden la prova, ja que només avisa que no es pot calcular el valor exacte de 'p' degut a la presència d'empats (parell de dades iguals).

```
> print(corr_matrix)
```

	estimate	p-value
Rooms	0.50429707	0.000000e+00
Distance	-0.18810737	1.886959e-215
Bedroom2	0.43326932	0.000000e+00
Bathroom	0.35847703	0.000000e+00
Car	0.25676798	0.000000e+00
Landsize	0.25754910	0.000000e+00
BuildingArea	0.46483594	0.000000e+00
YearBuilt	-0.26284133	0.000000e+00
Lattitude	-0.34461770	0.000000e+00
Longtitude	0.27837500	0.000000e+00
Propertycount	-0.04247446	2.316639e-12

Observem que la variable **Rooms** és la que més pes té a l'hora de determinar el preu de l'immoble, seguida de **BuildingArea** i **Bedroom2**.

5.- Conclusions

En aquest estudi d'un dataset hem realitzat un preprocessament de dades, fent selecció d'atributs, neteja de registres, conversió de dades i tractament de valors nuls i d'outliers.

Hem comprovat si les dades del dataset estaven normalitzades, així com la homogeneitat de la variància d'alguns grups.

Per tal de saber quin és l'atribut que més influeix en el preu final d'un immoble hem fet servir una matriu de correlacions, en particular la correlació d'Spearman. Mitjançant aquesta matriu hem detectat que l'atribut més influent és el numero d'habitacions que té l'immoble, seguit dels metres construits i a continuació del numero de dormitoris.

6.- Codi

La realització d'aquesta pràctica ha estat amb el llenguatge R, utilitzant com a eina el programari Rstudio, version 1.0.153.

Les dades una vegada aplicat el Cleaning Data s'han deixat en el fitxer "Mhm_data_clean.csv" amb la instrucció:

```
> write.csv(mhm, "Mhm_data_clean.csv")
```

Tots els fitxers referents a aquesta pràctica es poden trobar en el repositori de Github:

https://github.com/alnape/M2.951_practica2

7.- Recursos

Els següents recursos són d'utilitat per la realització de la pràctica:

- Megan Squire (2015). Clean Data. Packt Publishing Ltd.
- Jiawei Han, Micheine Kamber, Jian Pei (2012). Data mining: concepts and techniques. Morgan Kaufmann.
- Jason W. Osborne (2010). Data Cleaning Basics: Best Practices in Dealing with Extreme Scores. Newborn and Infant Nursing Reviews; 10 (1): pp. 1527-3369.
- Peter Dalgaard (2008). Introductory statistics with R. Springer Science & Business Media.
- https://en.wikipedia.org/wiki/Anderson%E2%80%93Darling_test
- https://cran.r-project.org/web/packages/nortest/nortest.pdf
- https://rpro.wikispaces.com/Prueba+de+Fligner-Killeen
- https://rpro.wikispaces.com/Estad%C3%ADstica+y+programaci%C3%B3n+con+R