```
library(tidyverse)

## Warning: package 'tidyverse' was built under R version 4.3.3

## Warning: package 'ggplot2' was built under R version 4.3.2

## Warning: package 'tibble' was built under R version 4.3.2

## Warning: package 'tidyr' was built under R version 4.3.3

## Warning: package 'readr' was built under R version 4.3.3

## Warning: package 'purrr' was built under R version 4.3.3

## Warning: package 'dplyr' was built under R version 4.3.2

## Warning: package 'forcats' was built under R version 4.3.3

## Warning: package 'lubridate' was built under R version 4.3.3

## ── Attaching core tidyverse packages ──────────────────────── tidyverse
2.0.0 ──
## ✓ dplyr     1.1.4     ✓ readr     2.1.5
## ✓ forcats   1.0.0     ✓ stringr   1.5.0
## ✓ ggplot2   3.4.4     ✓ tibble    3.2.1
## ✓ lubridate 1.9.3     ✓ tidyr     1.3.1
## ✓ purrr     1.0.2
## ── Conflicts ──────────────────────────────────────────
tidyverse_conflicts() ──
## ✕ dplyr::filter() masks stats::filter()
## ✕ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all
conflicts to become errors

library(packcircles)

## Warning: package 'packcircles' was built under R version 4.3.3

setwd("C:/Users/iceim/Dropbox/Data Analytics DKIT/Year 2/Project")
houses = read_csv("houses.csv")

## Rows: 1994 Columns: 19
## ── Column specification
────────────────────────────────────────────────────────
## Delimiter: ","
## chr (12): full_address, house_number, street_name, locality1, locality2,
loc...
## dbl  (4): id, bed_no, bath_no, size
## num  (3): sold_price_eur, asking_price_eur, price_diff
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this
message.
```

```
str(houses)
```

```
## spc_tbl_ [1,994 × 19] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ id              : num [1:1994] 780 763 1017 764 1036 ...
##  $ full_address    : chr [1:1994] "26 herbert park ballsbridge dublin 4
dublin" "60 ailesbury road ballsbridge dublin 4 dublin" "35 abbotts hill
malahide dublin" "1 argyle road donnybrook dublin 4 dublin" ...
##  $ house_number    : chr [1:1994] "26" "60" "35" "1" ...
##  $ street_name     : chr [1:1994] "herbert park" "ailesbury road" "abbotts
hill" "argyle road" ...
##  $ locality1       : chr [1:1994] "ballsbridge" "ballsbridge" NA
"donnybrook" ...
##  $ locality2       : chr [1:1994] NA NA NA NA ...
##  $ locality3       : chr [1:1994] NA NA NA NA ...
##  $ city_town       : chr [1:1994] "dublin 4" "dublin 4" "malahide" "dublin
4" ...
##  $ county          : chr [1:1994] "dublin" "dublin" "dublin" "dublin" ...
##  $ daft_sticker    : chr [1:1994] NA NA NA NA ...
##  $ ad_info         : chr [1:1994] "ADVANTAGE" NA NA NA ...
##  $ date_of_sale    : chr [1:1994] "23/08/2023" "06/05/2024" "11/04/2024"
"11/10/2023" ...
##  $ sold_price_eur  : num [1:1994] 4700000 3100000 3000000 2500000 2300000
...
##  $ asking_price_eur: num [1:1994] 5000000 3450000 2950000 2250000 2500000
...
##  $ price_diff      : num [1:1994] -300000 -350000 50000 250000 -200000
60000 35000 380000 100000 165000 ...
##  $ bed_no          : num [1:1994] 6 4 5 4 6 4 5 4 6 3 ...
##  $ bath_no         : num [1:1994] 3 4 5 NA 3 3 4 3 7 2 ...
##  $ house_type      : chr [1:1994] "Semi-D" "Detached" "Detached" "Semi-D"
...
##  $ size            : num [1:1994] 460 339 487 277 341 243 300 210 466 100
...
##  - attr(*, "spec")=
##   .. cols(
##   ..   id = col_double(),
##   ..   full_address = col_character(),
##   ..   house_number = col_character(),
##   ..   street_name = col_character(),
##   ..   locality1 = col_character(),
##   ..   locality2 = col_character(),
##   ..   locality3 = col_character(),
##   ..   city_town = col_character(),
##   ..   county = col_character(),
##   ..   daft_sticker = col_character(),
##   ..   ad_info = col_character(),
##   ..   date_of_sale = col_character(),
##   ..   sold_price_eur = col_number(),
##   ..   asking_price_eur = col_number(),
##   ..   price_diff = col_number(),
##   ..   bed_no = col_double(),
##   ..   bath_no = col_double(),
##   ..   house_type = col_character(),
##   ..   size = col_double()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

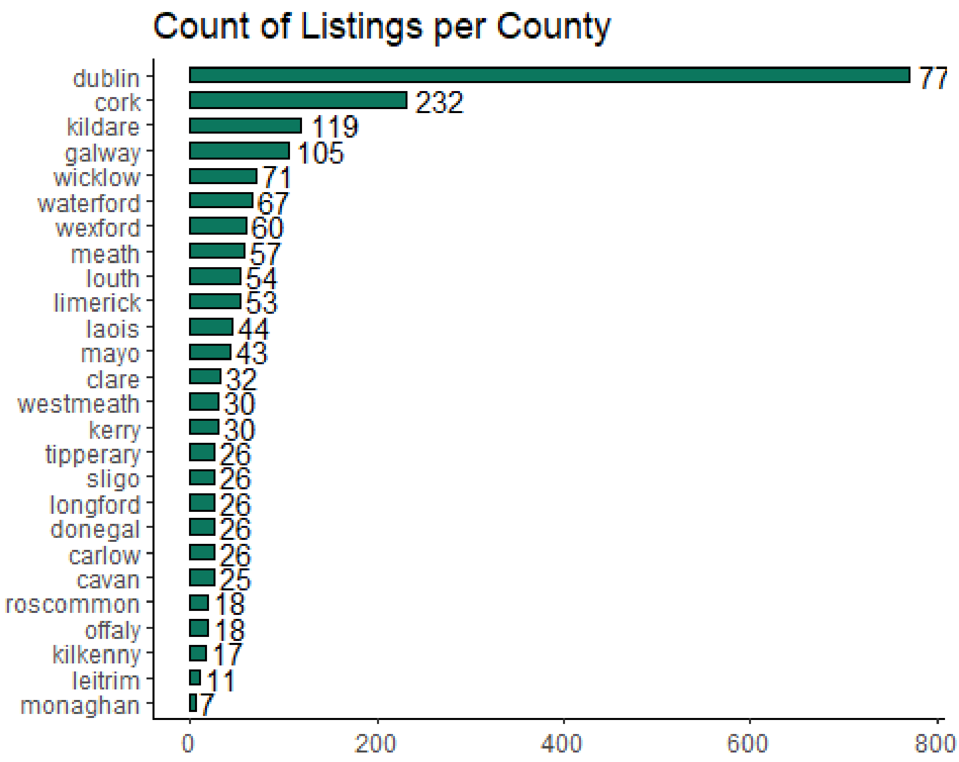
```
summary(houses)
```

```
##        id          full_address        house_number        street_name
##  Min.   :   1.0   Length:1994        Length:1994        Length:1994
##  1st Qu.: 499.2   Class :character   Class :character   Class :character
##  Median : 997.5   Mode  :character   Mode  :character   Mode  :character
##  Mean   : 997.5
##  3rd Qu.:1495.8
##  Max.   :1994.0
##
##   locality1           locality2           locality3           city_town
##  Length:1994        Length:1994        Length:1994        Length:1994
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##     county          daft_sticker         ad_info          date_of_sale
##  Length:1994        Length:1994        Length:1994        Length:1994
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  sold_price_eur    asking_price_eur    price_diff          bed_no
##  Min.   :  55000   Min.   :  45000   Min.   :-359800   Min.   :1.00
##  1st Qu.: 245000   1st Qu.: 240000   1st Qu.:  -2500   1st Qu.:3.00
##  Median : 342750   Median : 325000   Median :  10000   Median :3.00
##  Mean   : 401824   Mean   : 386370   Mean   :  15146   Mean   :3.16
##  3rd Qu.: 479375   3rd Qu.: 458000   3rd Qu.:  33000   3rd Qu.:4.00
##  Max.   :4700000   Max.   :5000000   Max.   : 380000   Max.   :7.00
##
##     bath_no      house_type             size
##  Min.   :1.00   Length:1994        Min.   : 32.0
##  1st Qu.:1.00   Class :character   1st Qu.: 83.0
##  Median :2.00   Mode  :character   Median :104.0
##  Mean   :2.17                      Mean   :112.6
##  3rd Qu.:3.00                      3rd Qu.:130.0
##  Max.   :7.00                         Max.   :520.0  
##  NA's   :21                           NA's   :357

# Converting the dataframe into a tibble
as_tibble(houses)

## # A tibble: 1,994 × 19
##       id full_address     house_number street_name locality1 locality2
locality3
##    <dbl> <chr>            <chr>        <chr>       <chr>     <chr>    
<chr>    
##  1   780 26 herbert park… 26           herbert pa… ballsbri… <NA>     
<NA>     
##  2   763 60 ailesbury ro… 60           ailesbury … ballsbri… <NA>     
<NA>     
##  3  1017 35 abbotts hill… 35           abbotts hi… <NA>      <NA>     
<NA>     
##  4   764 1 argyle road d… 1            argyle road donnybro… <NA>     
<NA>     
##  5  1036 4 willow bank m… 4            willow bank <NA>      <NA>     
<NA>     
##  6   772 135 strand road… 135          strand road sandymou… <NA>     
<NA>     
##  7   957 24 corrig avenu… 24           corrig ave… <NA>      <NA>     
<NA>     
##  8   859 159 templeogue … 159          templeogue… terenure  <NA>     
<NA>     
##  9  1969 54 eagle valley… 54           eagle vall… <NA>      <NA>     
<NA>     
## 10   683 17 lad lane upp… 17           lad lane u… <NA>      <NA>     
<NA>     
## # ℹ 1,984 more rows
## # ℹ 12 more variables: city_town <chr>, county <chr>, daft_sticker <chr>,
## #   ad_info <chr>, date_of_sale <chr>, sold_price_eur <dbl>,
## #   asking_price_eur <dbl>, price_diff <dbl>, bed_no <dbl>, bath_no <dbl>,
## #   house_type <chr>, size <dbl>
```

—————- UNIVARIATE ANALYSIS —————-

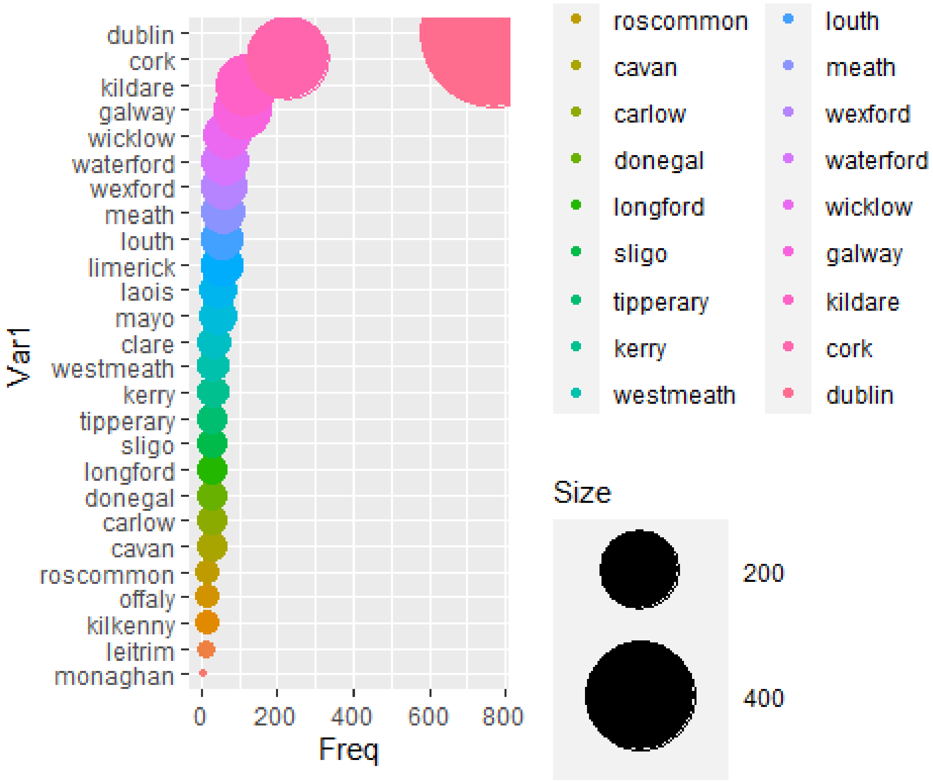
```
# Creating a table from the county column, count in decreasing order, to
prepare for the barplot
table_county <- table(houses$county)
table_county <- table_county[order(table_county, decreasing=FALSE)]
table_county

## 
##   monaghan   leitrim  kilkenny    offaly roscommon     cavan    carlow
donegal 
##          7        11        17        18        18        25        26
26 
##   longford     sligo tipperary     kerry westmeath     clare      mayo
```

```
laois
##        26        26        26        30        30        32        43
44
##  limerick     louth     meath   wexford waterford   wicklow    galway
kildare
##        53        54        57        60        67        71       105
119
##      cork    dublin
##       232       771

# Converting the table into a Data Frame for ggplot2
df_county <- data.frame(table_county)
df_county

##         Var1 Freq
## 1   monaghan    7
## 2    leitrim   11
## 3   kilkenny   17
## 4     offaly   18
## 5  roscommon   18
## 6      cavan   25
## 7     carlow   26
## 8    donegal   26
## 9   longford   26
## 10     sligo   26
## 11 tipperary   26
## 12     kerry   30
## 13 westmeath   30
## 14     clare   32
## 15      mayo   43
## 16     laois   44
## 17  limerick   53
## 18     louth   54
## 19     meath   57
## 20   wexford   60
## 21 waterford   67
## 22   wicklow   71
## 23    galway  105
## 24   kildare  119
## 25      cork  232
## 26    dublin  771

ggplot(df_county, aes(x=Freq, y=Var1)) +
  geom_col(color = "black", fill = "#0c775e", width = 0.6) +
  labs(title="Count of Listings per County", x=NULL, y=NULL) +
  geom_text(aes(label = Freq), hjust = -0.2)+
  theme_classic()
```



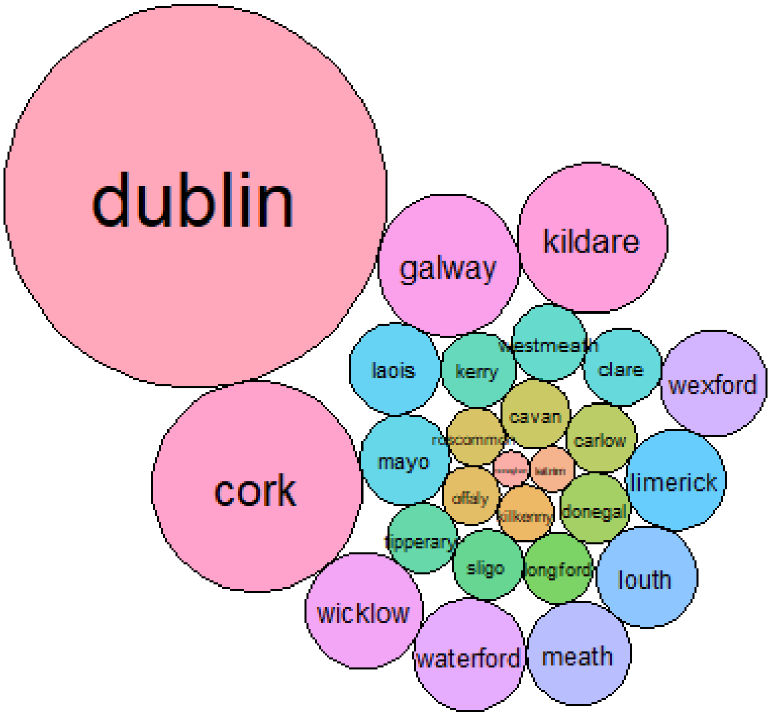
Over 1/3 of the listings are located in Dublin, while the other listings are spread unevenly between the other 25 counties. Merging those counties into meaningful groups has to be considered for efficient analysis.

```r
ggplot(df_county, aes(x = Freq, y = Var1, size = Freq, color = Var1)) +
  geom_point() +
  scale_size(name = "Size", range = c(1, 26))
```



```
packing <- circleProgressiveLayout(df_county$Freq, sizetype='area')

# We can add these packing information to the initial data frame
data <- cbind(df_county, packing)

# Check that radius is proportional to value. We don't want a linear
relationship, since it is the AREA that must be proportionnal to the value
#plot(data$radius, data$value)

# The next step is to go from one center + a radius to the coordinates of a
circle that
# is drawn by a multitude of straight lines.
dat.gg <- circleLayoutVertices(packing, npoints=50)

# Make the plot
ggplot() +

  # Make the bubbles
  geom_polygon(data = dat.gg, aes(x, y, group = id, fill=as.factor(id)),
colour = "black", alpha = 0.6) +

  # Add text in the center of each bubble + control its size
  geom_text(data = data, aes(x, y, size=Freq, label = Var1)) +
  scale_size_continuous(range = c(1,10)) +

  # General theme:
```

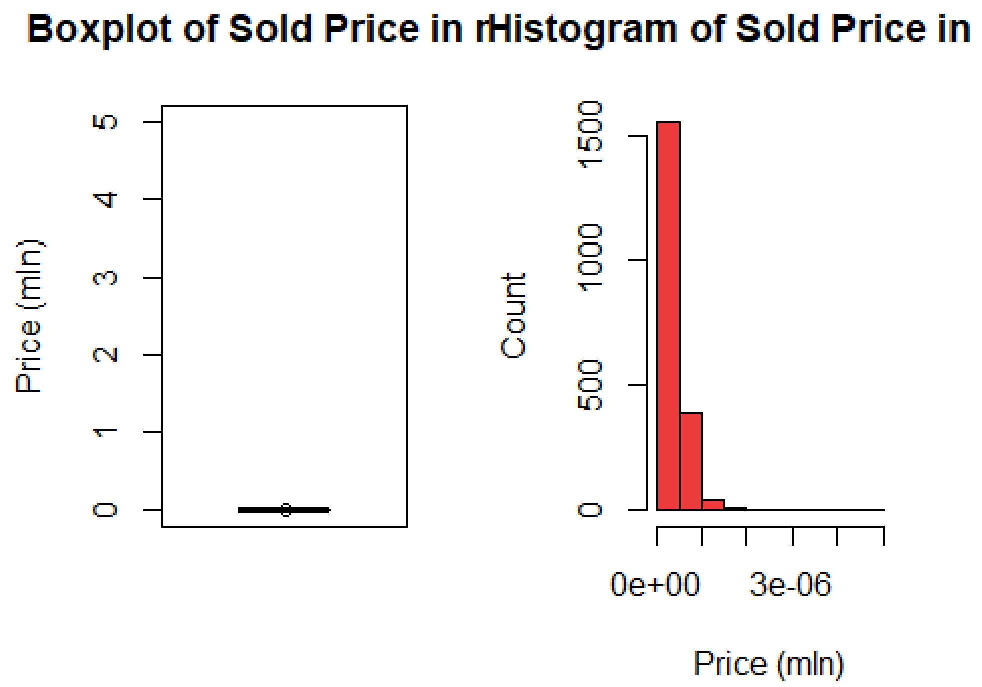
```
  theme_void() +
  theme(legend.position="none") +
  coord_equal()
```



```
# Transforming price variable into millions of Euro to make the graphs more
legible
house_price_mln <- houses %>%
  mutate(sold_price_eur = sold_price_eur/1000000,
         asking_price_eur = asking_price_eur/1000000)

head(house_price_mln)

## # A tibble: 6 × 19
##       id full_address       house_number street_name locality1 locality2
locality3
##    <dbl> <chr>              <chr>        <chr>       <chr>     <chr>
<chr>
## 1   780 26 herbert park … 26           herbert pa… ballsbri… <NA>
<NA>
## 2   763 60 ailesbury roa… 60           ailesbury … ballsbri… <NA>
<NA>
## 3  1017 35 abbotts hill … 35           abbotts hi… <NA>      <NA>
<NA>
## 4   764 1 argyle road do… 1            argyle road donnybro… <NA>
<NA>
## 5  1036 4 willow bank mo… 4            willow bank <NA>      <NA>
<NA>
```

```
## 6   772 135 strand road … 135          strand road sandymou… <NA>
<NA>
## # ℹ 12 more variables: city_town <chr>, county <chr>, daft_sticker <chr>,
## #   ad_info <chr>, date_of_sale <chr>, sold_price_eur <dbl>,
## #   asking_price_eur <dbl>, price_diff <dbl>, bed_no <dbl>, bath_no <dbl>,
## #   house_type <chr>, size <dbl>
```

```
par(mfrow=c(1,2))
boxplot(house_price_mln$sold_price_eur/1000000, main="Boxplot of Sold Price
in mln", ylab="Price (mln)", ylim=c(0,5), col="brown2")
hist(house_price_mln$sold_price_eur/1000000, main="Histogram of Sold Price in
mln", xlab="Price (mln)", ylab="Count", col="brown2")
```



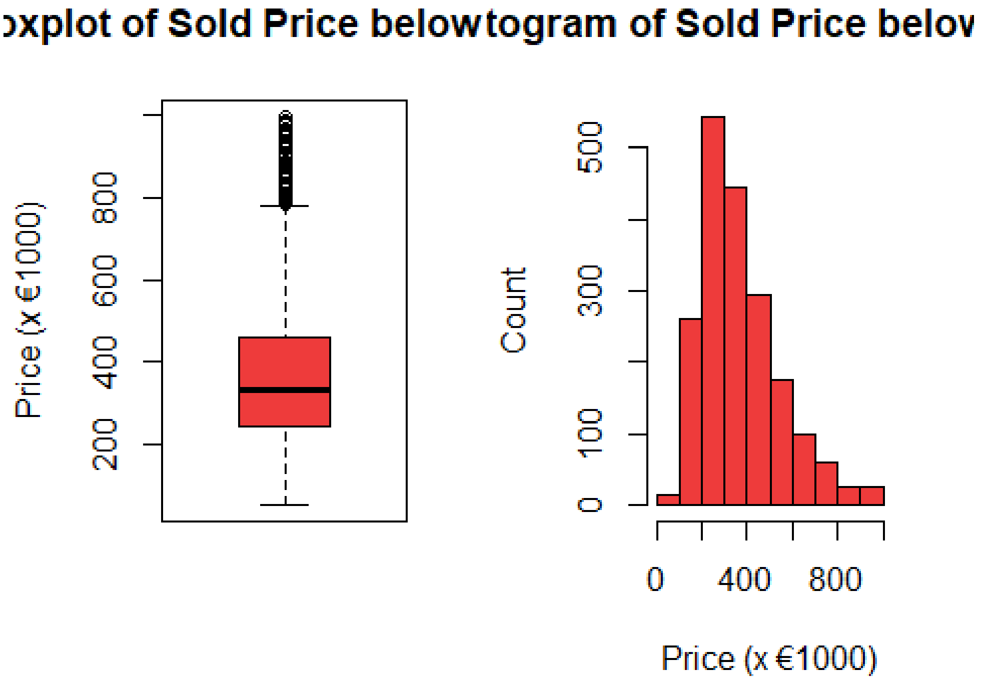
The graphs show that the median house price is around €300,000 (€342,000), and that most houses in this dataset were sold for below €1mln. There are a good few outliers between €1mln and €2mln, and five extreme outliers - properties that were sold for between €2mln and €5mln. A subset has to be created to take a closer look at the distribution of the houses with sold price below €1mln.

```
houses_below1m <- subset(houses, sold_price_eur<1000000)
houses_below1m %>% arrange(desc(sold_price_eur)) %>% head()
```

```
## # A tibble: 6 × 19
##       id full_address      house_number street_name locality1 locality2
locality3
##    <dbl> <chr>             <chr>        <chr>       <chr>     <chr>
<chr>
```

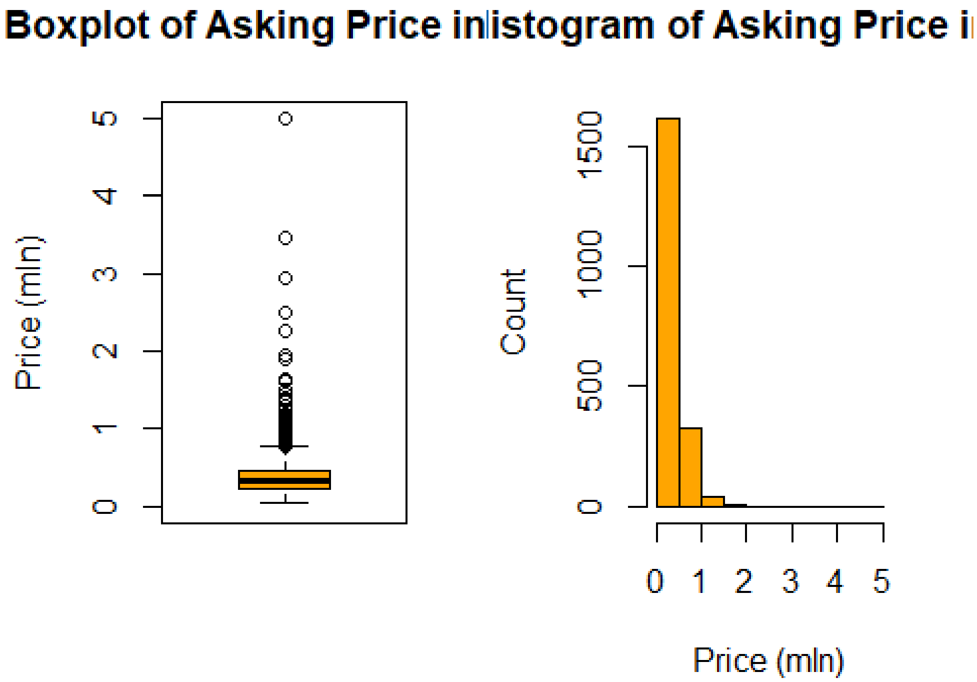
```
## 1  1023 15 lambay court … 15          lambay cou… <NA>      <NA>
<NA>
## 2   536 49 castleknock p… 49          castleknoc… castlekn… <NA>
<NA>
## 3   353 83 barclay court… 83          barclay co… <NA>      <NA>
<NA>
## 4   510 65 ballytore roa… 65          ballytore … rathfarn… <NA>
<NA>
## 5   848 112 sandford roa… 112         sandford r… ranelagh  <NA>
<NA>
## 6   512 85 butterfield p… 85          butterfiel… rathfarn… <NA>
<NA>
## # ℹ 12 more variables: city_town <chr>, county <chr>, daft_sticker <chr>,
## #   ad_info <chr>, date_of_sale <chr>, sold_price_eur <dbl>,
## #   asking_price_eur <dbl>, price_diff <dbl>, bed_no <dbl>, bath_no <dbl>,
## #   house_type <chr>, size <dbl>
```

```
par(mfrow=c(1,2))
boxplot(houses_below1m$sold_price_eur/1000, main="Boxplot of Sold Price below
€1mln", ylab="Price (x €1000)", col="brown2")
hist(houses_below1m$sold_price_eur/1000, main="Histogram of Sold Price below
€1mln", xlab="Price (x €1000)", ylab="Count", col="brown2")
```

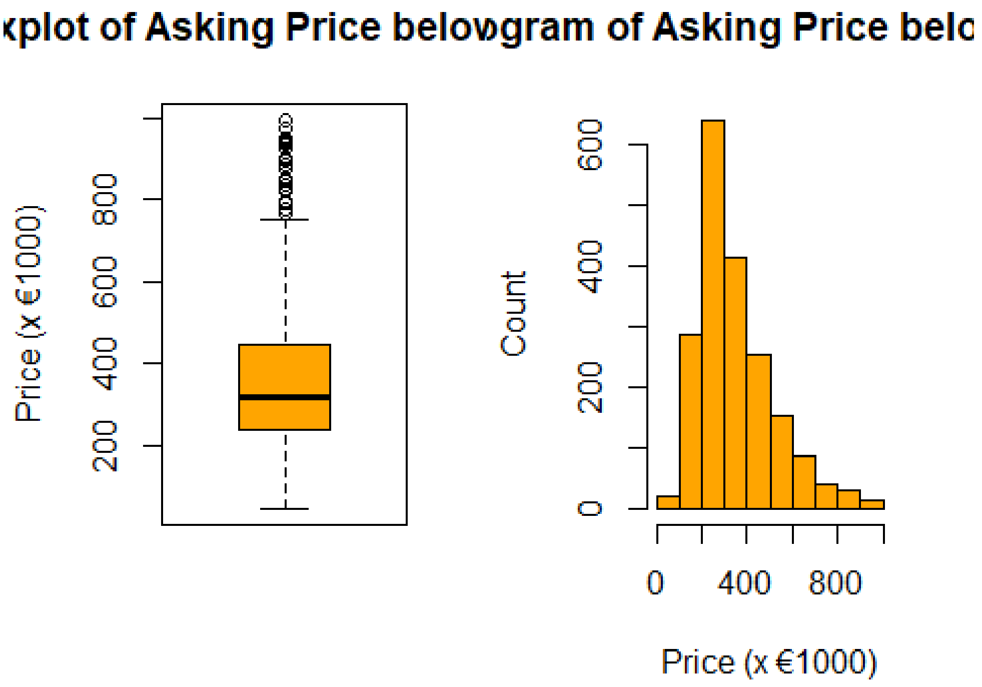


```
par(mfrow=c(1,2))
boxplot(house_price_mln$asking_price_eur, main="Boxplot of Asking Price in
mln", ylab="Price (mln)", ylim=c(0,5), col="orange")
```

```r
hist(house_price_mln$asking_price_eur, main="Histogram of Asking Price in mln", xlab="Price (mln)", ylab="Count", col="orange")
```



```r
par(mfrow=c(1,2))
boxplot(houses_below1m$asking_price_eur/1000, main="Boxplot of Asking Price below €1mln", ylab="Price (x €1000)", col="orange")
hist(houses_below1m$asking_price_eur/1000, main="Histogram of Asking Price below €1mln", xlab="Price (x €1000)", ylab="Count", col="orange")
```

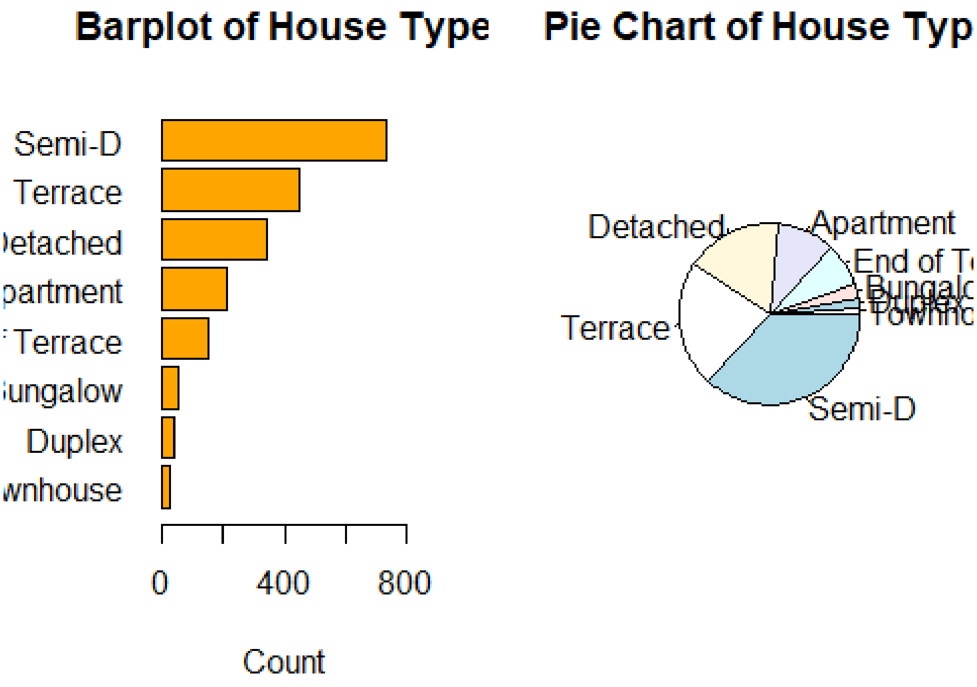


```
# Looking at the different house types
house_types <- table(houses$house_type)
house_types <- house_types[order(house_types, decreasing=FALSE)]
house_types

## 
##      Townhouse         Duplex       Bungalow End of Terrace      Apartment 
##             23             35             52            152            207 
##       Detached        Terrace         Semi-D 
##            344            446            735 

prop.table(house_types)

## 
##      Townhouse         Duplex       Bungalow End of Terrace      Apartment 
##     0.01153460     0.01755266     0.02607823     0.07622869     0.10381143 
##       Detached        Terrace         Semi-D 
##     0.17251755     0.22367101     0.36860582 
```

There are 8 different house types. Most of them are Semi-Detached. For the purposes of the analysis, the house types which are similar need to be merged, i.e. Terrace + Townhouse, Apartment + Duplex, Detached + Bungalow, Semi-D + End of Terrace, which would narrow it down to 4 groups.

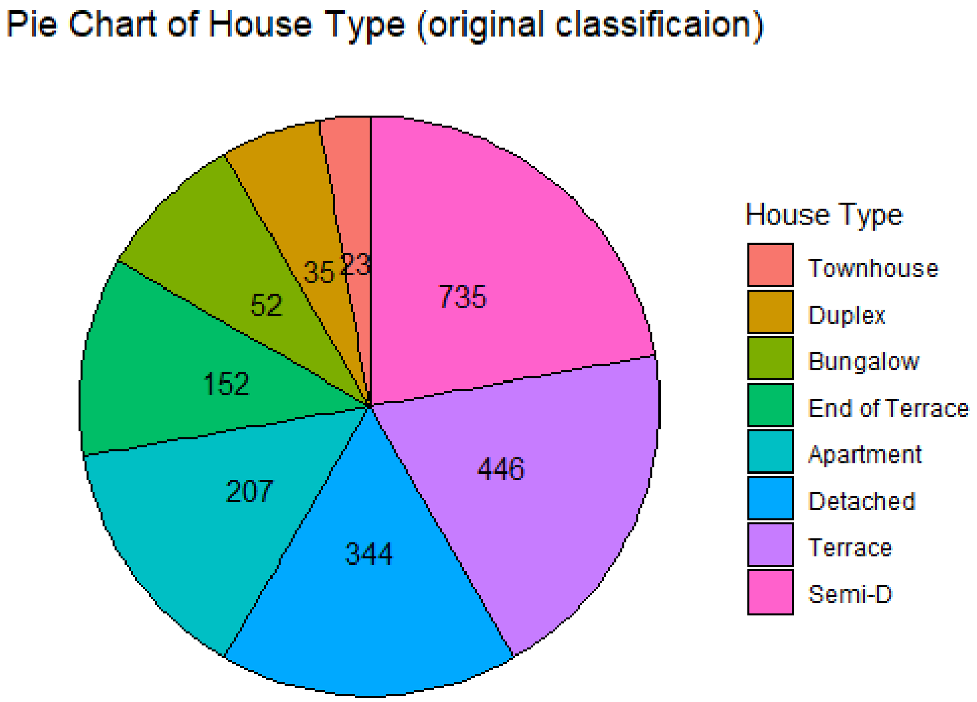
Apartment = c(“Apartment”, “Duplex”), Detached = c(“Detached”, “Bungalow”), Semi-D = c(“Semi-D”, “End of Terrace”), Terrace = c(“Terrace”, “Townhouse”),

```r
par(mfrow=c(1,2))
barplot(house_types, main="Barplot of House Type", xlim=c(0,800),
xlab="Count", col="orange", horiz=T, las=1)
pie(house_types, main="Pie Chart of House Type")
```

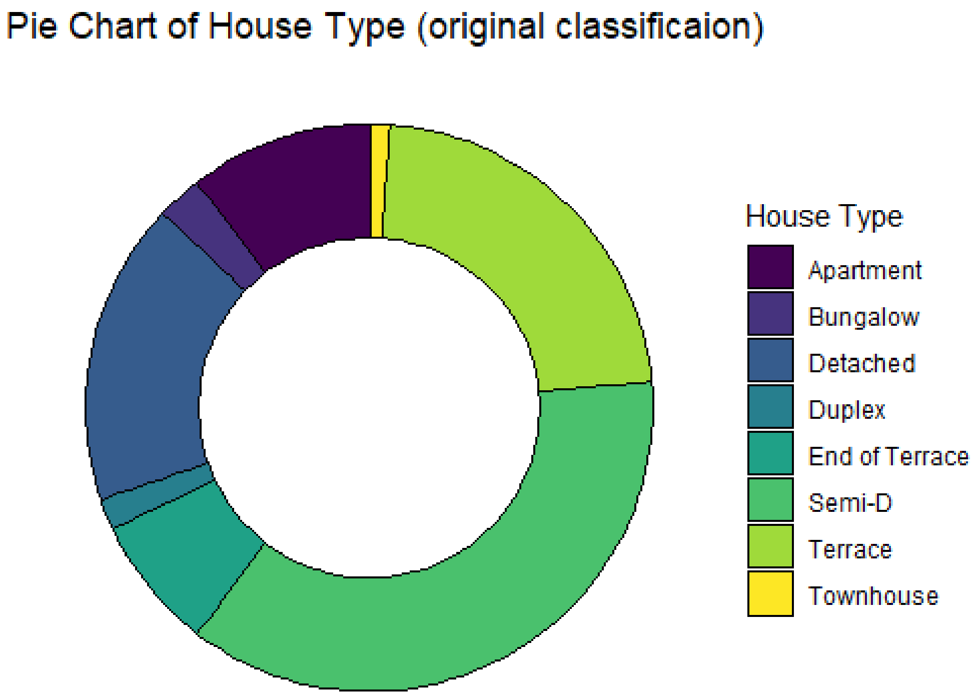


```r
house_types_df <- data.frame(house_types)
house_types_df

##            Var1 Freq
## 1     Townhouse   23
## 2        Duplex   35
## 3      Bungalow   52
## 4 End of Terrace  152
## 5     Apartment  207
## 6      Detached  344
## 7       Terrace  446
## 8        Semi-D  735

h1 <- ggplot(house_types_df, aes(x = "", y=Var1, fill=Var1)) +
  geom_col(color = "black") +
  geom_text(aes(label = Freq),
            position = position_stack(vjust = 0.5)) +
  coord_polar(theta = "y") +
  labs(title ="Pie Chart of House Type (original classificaion)")+
  guides(fill=guide_legend(title="House Type"))+
  theme_void()
```

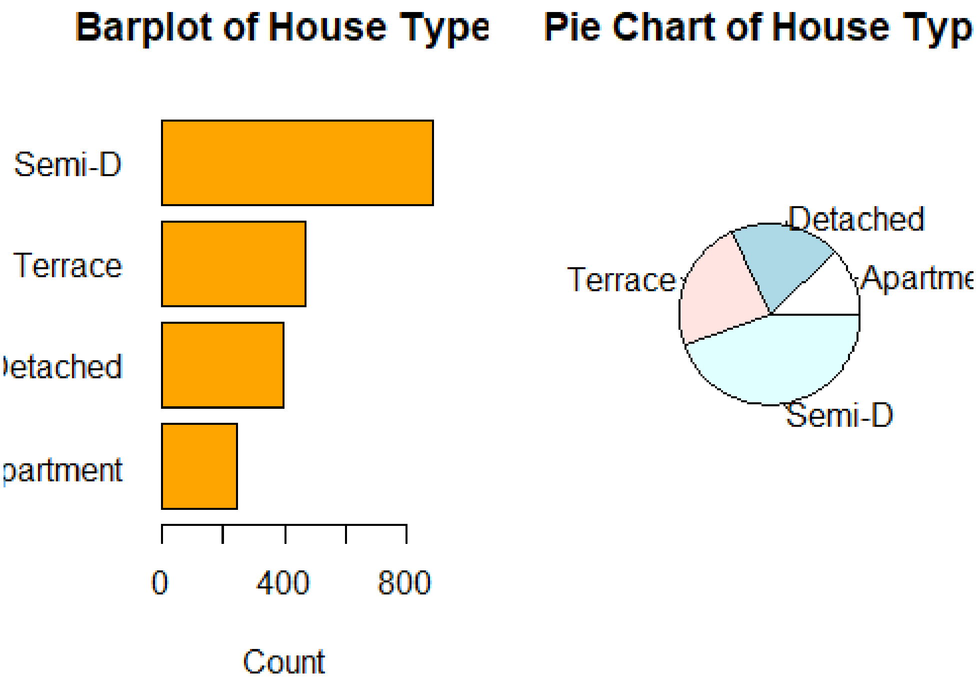
```
h1
```



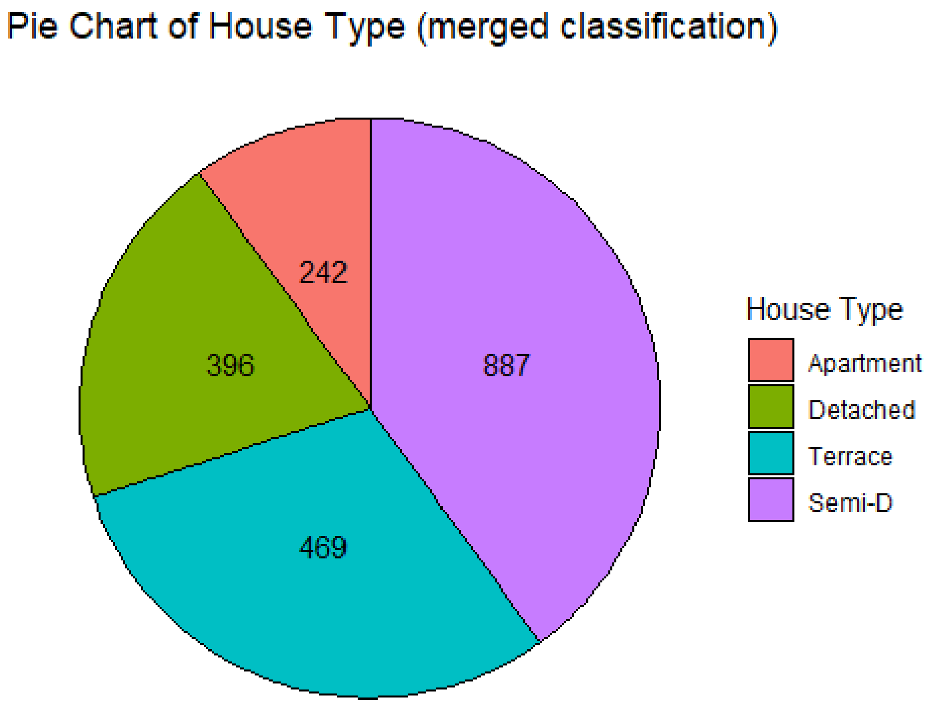
```
houses %>% data.frame(houses) %>%
  ggplot(aes(y = 2, fill=house_type)) +
  geom_bar(color = "black") +
  theme_void()+
  scale_fill_viridis_d() +
  coord_polar(theta = "x") +
  ylim(0.2,2.5)+
  labs(title ="Pie Chart of House Type (original classificaion)")+
  guides(fill=guide_legend(title="House Type"))
```



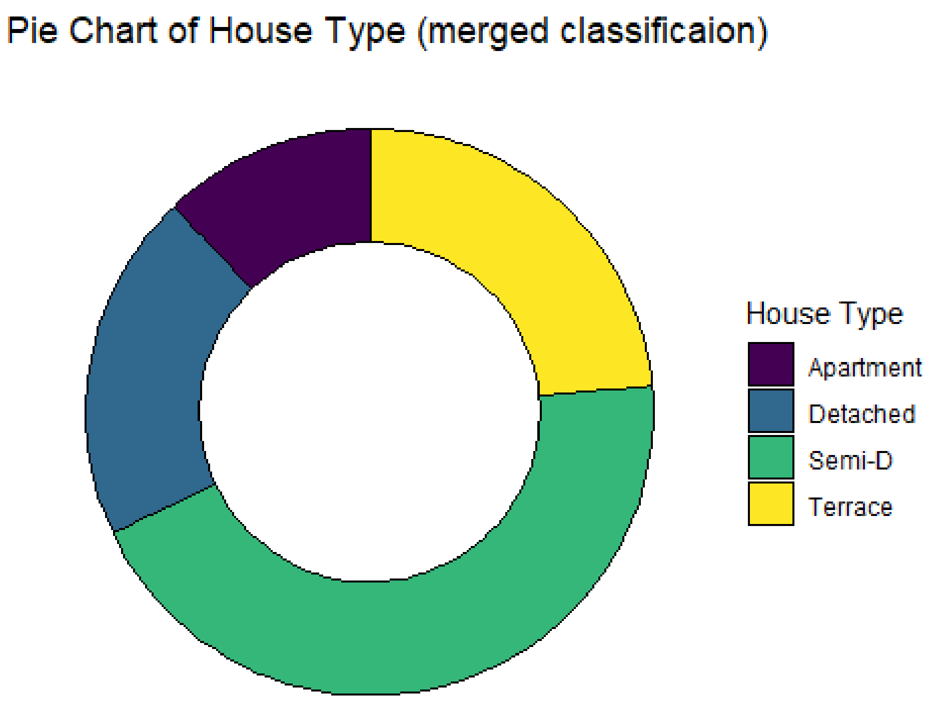
```
# Defining replacement values

replace_house_types <- c("Duplex"="Apartment",
                         "Bungalow"="Detached",
                         "End of Terrace"="Semi-D",
                         "Townhouse"="Terrace")

# Using str_replace_all() to replace the names in the house_type column
house_types_collapsed <- data.frame(houses)
house_types_collapsed$house_type <-
str_replace_all(house_types_collapsed$house_type, replace_house_types)
#view(house_types_collapsed)

house_types_collapsed_tbl <- table(house_types_collapsed$house_type)
house_types_collapsed_tbl <-
house_types_collapsed_tbl[order(house_types_collapsed_tbl, decreasing=FALSE)]
house_types_collapsed_tbl

##
## Apartment  Detached   Terrace    Semi-D
##       242       396       469       887

par(mfrow=c(1,2))
barplot(house_types_collapsed_tbl, main="Barplot of House Type",
xlim=c(0,800), xlab="Count", col="orange", horiz=T, las=1)
pie(house_types_collapsed_tbl, main="Pie Chart of House Type")
```



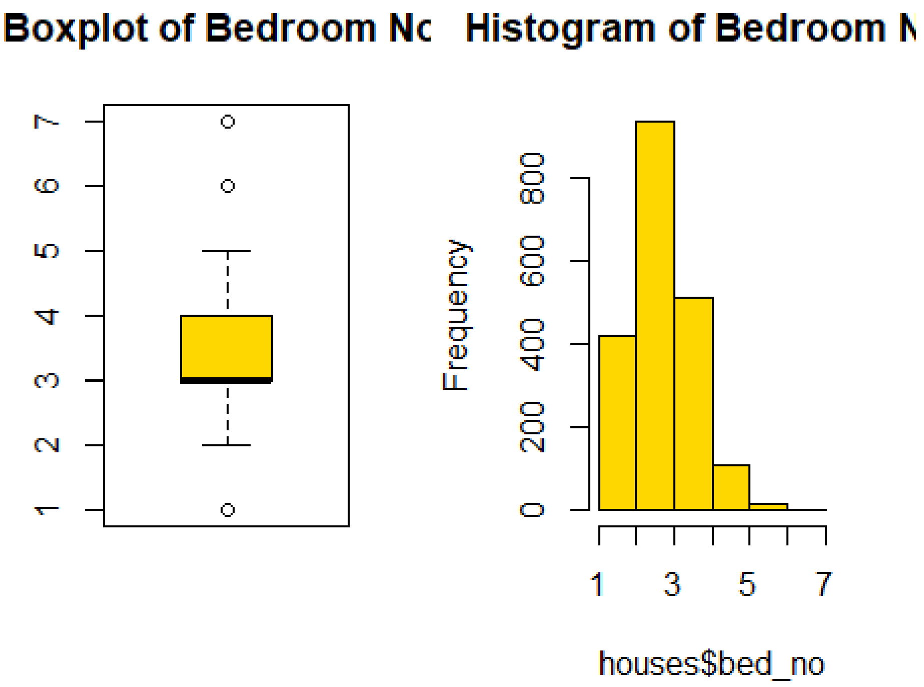
```
house_types_collapsed_df <- data.frame(house_types_collapsed_tbl)
house_types_collapsed_df

##        Var1 Freq
## 1 Apartment  242
## 2  Detached  396
## 3   Terrace  469
## 4    Semi-D  887

prop.table(house_types_collapsed_tbl)

## 
## Apartment  Detached   Terrace    Semi-D 
## 0.1213641 0.1985958 0.2352056 0.4448345

h2 <- ggplot(house_types_collapsed_df, aes(x = "", y=Var1, fill=Var1)) +
  geom_col(color = "black") +
  geom_text(aes(label = Freq),
            position = position_stack(vjust = 0.5)) +
  coord_polar(theta = "y") +
  labs(title ="Pie Chart of House Type (merged classification)")+
  guides(fill=guide_legend(title="House Type"))+
  theme_void()

h2
```



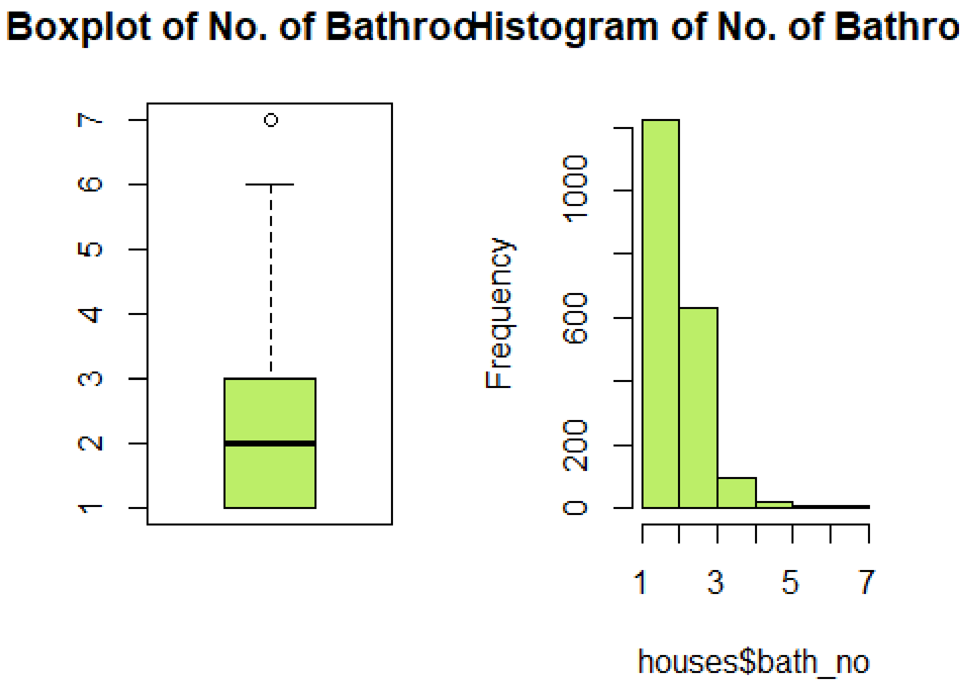
```r
house_types_collapsed %>% data.frame(house_types_collapsed) %>%
  ggplot(aes(y = 2, fill=house_type)) +
  geom_bar(color = "black") +
  theme_void()+
  scale_fill_viridis_d() +
  coord_polar(theta = "x") +
  ylim(0.2,2.5)+
  labs(title ="Pie Chart of House Type (merged classificaion)")+
  guides(fill=guide_legend(title="House Type"))
```



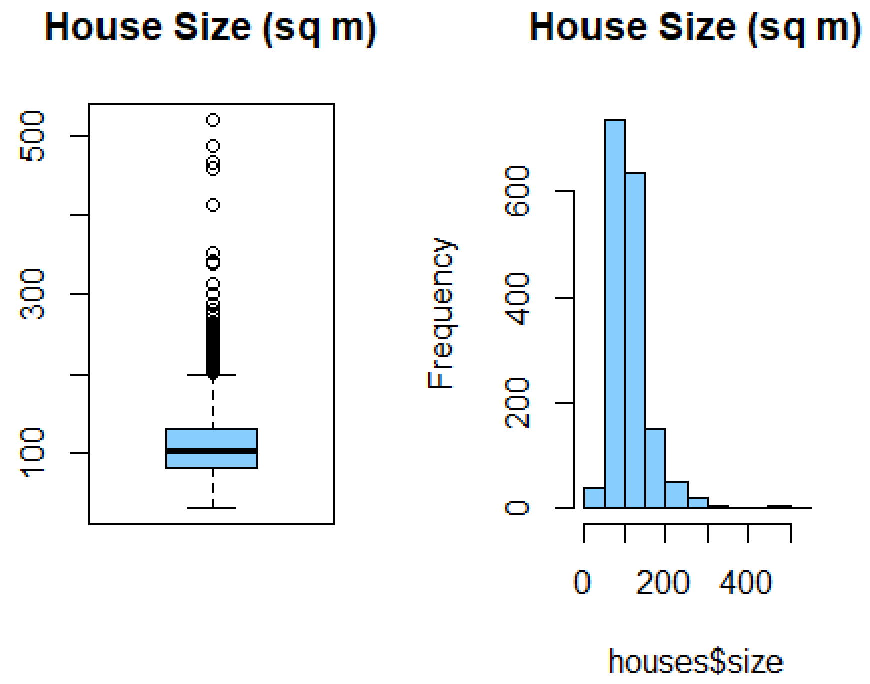
```
par(mfrow=c(1,2))
boxplot(houses$bed_no, main="Boxplot of Bedroom No.", col="gold1")
hist(houses$bed_no, breaks=7, main="Histogram of Bedroom No.", col="gold1")
```



```r
par(mfrow=c(1,2))
boxplot(houses$bath_no, main="Boxplot of No. of Bathrooms",
col="darkolivegreen2")
hist(houses$bath_no, breaks=7, main="Histogram of No. of Bathrooms",
col="darkolivegreen2")
```

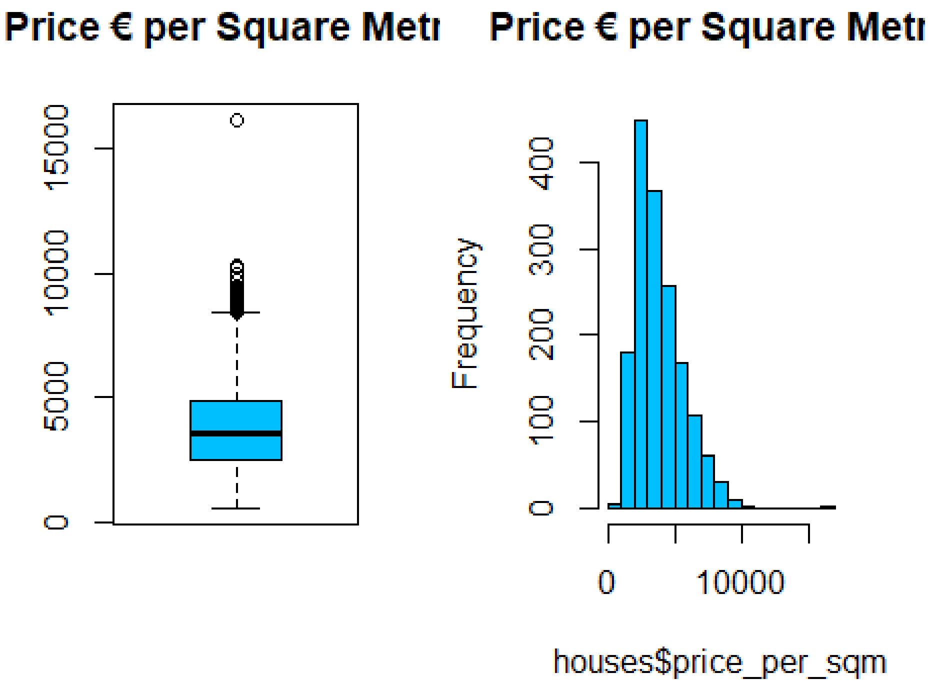


```r
par(mfrow=c(1,2))
boxplot(houses$size, main="House Size (sq m)", col="skyblue1")
hist(houses$size, main="House Size (sq m)", col="skyblue1")
```

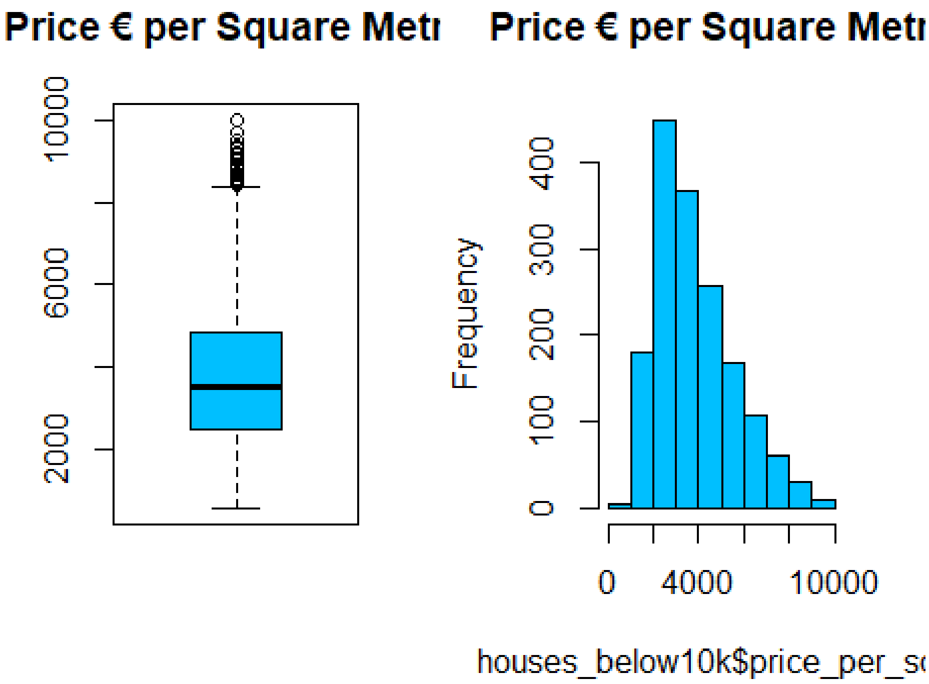


```
houses <- houses %>%
  mutate(price_per_sqm = round(sold_price_eur / size, 2))

head(houses)

## # A tibble: 6 × 20
##      id full_address      house_number street_name locality1 locality2
locality3
##   <dbl> <chr>             <chr>        <chr>       <chr>     <chr>
<chr>
## 1   780 26 herbert park … 26           herbert pa… ballsbri… <NA>
<NA>
## 2   763 60 ailesbury roa… 60           ailesbury … ballsbri… <NA>
<NA>
## 3  1017 35 abbotts hill … 35           abbotts hi… <NA>      <NA>
<NA>
## 4   764 1 argyle road do… 1            argyle road donnybro… <NA>
<NA>
## 5  1036 4 willow bank mo… 4            willow bank <NA>      <NA>
<NA>
## 6   772 135 strand road … 135          strand road sandymou… <NA>
<NA>
## # ℹ 13 more variables: city_town <chr>, county <chr>, daft_sticker <chr>,
## #   ad_info <chr>, date_of_sale <chr>, sold_price_eur <dbl>,
## #   asking_price_eur <dbl>, price_diff <dbl>, bed_no <dbl>, bath_no <dbl>,
## #   house_type <chr>, size <dbl>, price_per_sqm <dbl>
```

```
tail(houses)

## # A tibble: 6 × 20
##      id full_address      house_number street_name locality1 locality2
locality3
##   <dbl> <chr>             <chr>        <chr>       <chr>     <chr>
<chr>
## 1  1573 5 lord edward st… 5            lord edwar… <NA>      <NA>
<NA>
## 2  1764 81 james connoll… 81           james conn… <NA>      <NA>
<NA>
## 3  1900 32 waterside new… 32           waterside   <NA>      <NA>
<NA>
## 4   181 71 gerald griffi… 71           gerald gri… <NA>      <NA>
blackpool
## 5  1385 7 chapel street … 7            chapel str… <NA>      <NA>
<NA>
## 6  1697 7 saint ronans p… 7            saint rona… <NA>      <NA>
<NA>
## # ℹ 13 more variables: city_town <chr>, county <chr>, daft_sticker <chr>,
## #   ad_info <chr>, date_of_sale <chr>, sold_price_eur <dbl>,
## #   asking_price_eur <dbl>, price_diff <dbl>, bed_no <dbl>, bath_no <dbl>,
## #   house_type <chr>, size <dbl>, price_per_sqm <dbl>

par(mfrow=c(1,2))
boxplot(houses$price_per_sqm, main="Price € per Square Metre",
col="deepskyblue")
hist(houses$price_per_sqm, main="Price € per Square Metre",
col="deepskyblue")
```



```
houses_below10k <- subset(houses, price_per_sqm <=10000)

par(mfrow=c(1,2))
boxplot(houses_below10k$price_per_sqm, main="Price € per Square Metre",
col="deepskyblue")
hist(houses_below10k$price_per_sqm, main="Price € per Square Metre",
col="deepskyblue")
```



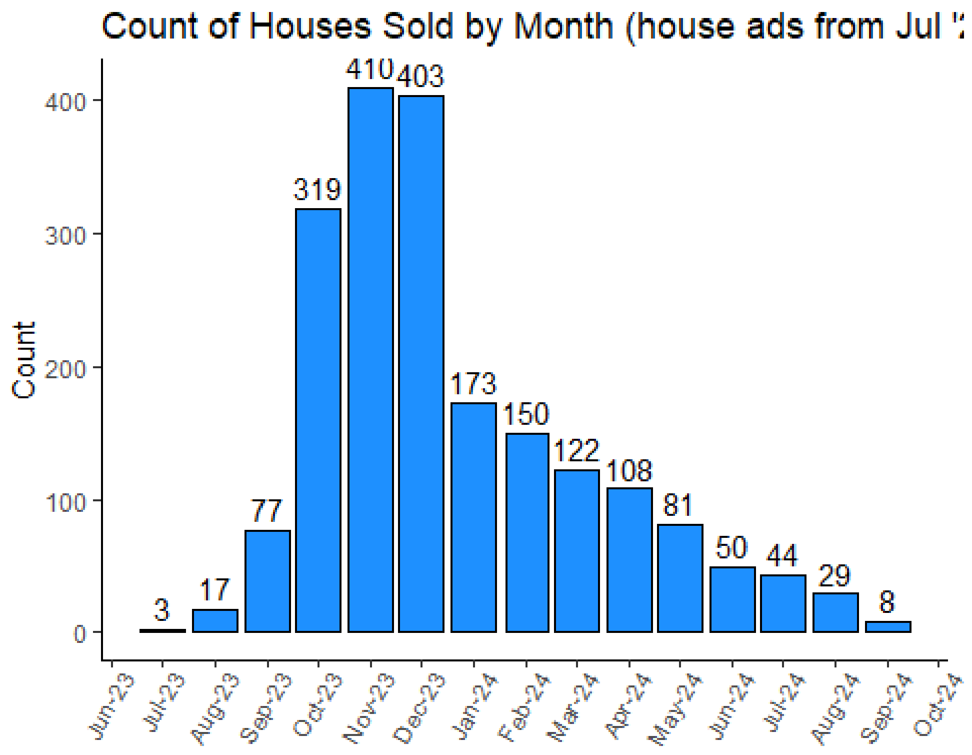
```
# Dealing with date, converting it from character
#library(lubridate)
houses$date_of_sale <- as.Date(houses$date_of_sale, format="%d/%m/%Y")
#houses$date_of_sale <- as.Date(dmy(houses$date_of_sale))
print(houses)

## # A tibble: 1,994 × 20
##        id full_address      house_number street_name locality1 locality2
locality3
##     <dbl> <chr>             <chr>        <chr>       <chr>     <chr>
<chr>
##  1   780 26 herbert park… 26           herbert pa… ballsbri… <NA>
<NA>
##  2   763 60 ailesbury ro… 60           ailesbury … ballsbri… <NA>
<NA>
##  3  1017 35 abbotts hill… 35           abbotts hi… <NA>      <NA>
<NA>
##  4   764 1 argyle road d… 1            argyle road donnybro… <NA>
<NA>
##  5  1036 4 willow bank m… 4            willow bank <NA>      <NA>
<NA>
##  6   772 135 strand road… 135          strand road sandymou… <NA>
<NA>
##  7   957 24 corrig avenu… 24           corrig ave… <NA>      <NA>
<NA>
##  8   859 159 templeogue … 159          templeogue… terenure  <NA>
<NA>
```

```
## 9   1969 54 eagle valley… 54           eagle vall… <NA>      <NA>
<NA>
## 10   683 17 lad lane upp… 17           lad lane u… <NA>      <NA>
<NA>
## # ℹ 1,984 more rows
## # ℹ 13 more variables: city_town <chr>, county <chr>, daft_sticker <chr>,
## #   ad_info <chr>, date_of_sale <date>, sold_price_eur <dbl>,
## #   asking_price_eur <dbl>, price_diff <dbl>, bed_no <dbl>, bath_no <dbl>,
## #   house_type <chr>, size <dbl>, price_per_sqm <dbl>
```

```
count_by_month <- houses %>%
    group_by(month = lubridate::floor_date(date_of_sale, 'month')) %>%
    count() %>%
    arrange(month)



print(count_by_month)
```

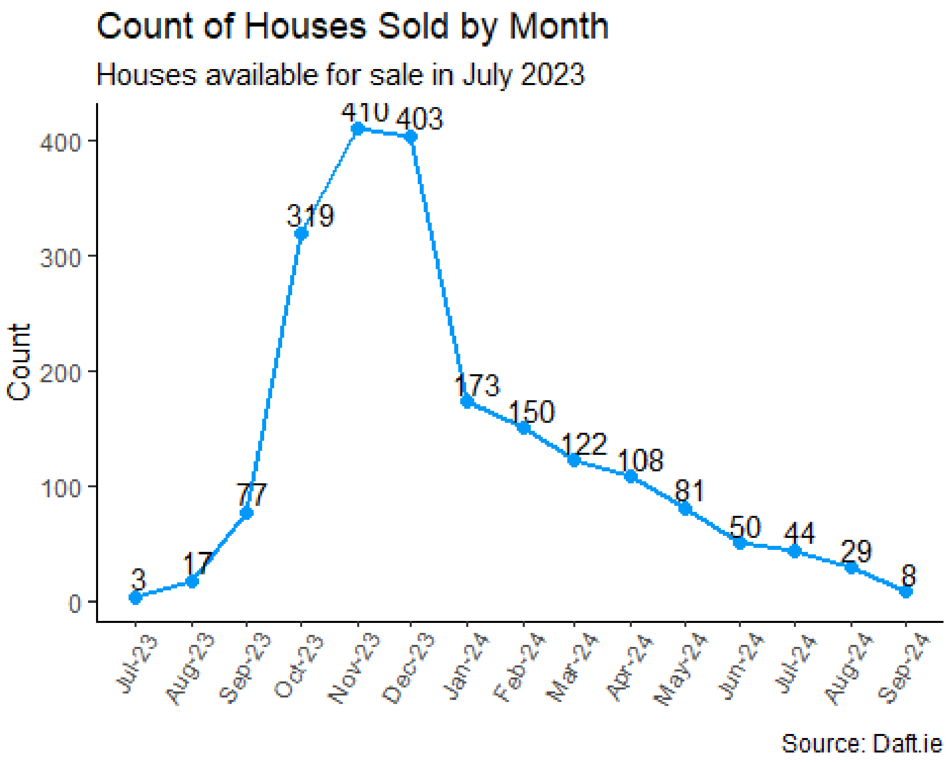
```
## # A tibble: 15 × 2
## # Groups:   month [15]
##    month          n
##    <date>     <int>
##  1 2023-07-01     3
##  2 2023-08-01    17
##  3 2023-09-01    77
##  4 2023-10-01   319
##  5 2023-11-01   410
##  6 2023-12-01   403
##  7 2024-01-01   173
##  8 2024-02-01   150
##  9 2024-03-01   122
## 10 2024-04-01   108
## 11 2024-05-01    81
## 12 2024-06-01    50
## 13 2024-07-01    44
## 14 2024-08-01    29
## 15 2024-09-01     8
```

```
# PLOT MONTHS
months_df <- data.frame(count_by_month)

ggplot(months_df, aes(x=month, y=n))+
  geom_col(color = "black", fill="dodgerblue")+
  labs(title="Count of Houses Sold by Month (house ads from Jul '23)",
x=NULL, y="Count")+
  theme_classic()+
  theme(axis.text.x = element_text(angle = 60, hjust = 0.9))+
  scale_x_date(date_labels="%b-%y", breaks="1 month")+
  geom_text(aes(label = n), hjust = 0.5, vjust = -0.4)
```



```r
ggplot(months_df, aes(x=month, y=n))+
  geom_path(color = "#0099f9", size = 1)+
  geom_point(color = "#0099f9", size = 2)+
  labs(title="Count of Houses Sold by Month", x=NULL, y="Count",
       subtitle = "Houses available for sale in July 2023",
       caption = "Source: Daft.ie")+
  theme_classic()+
  theme(axis.text.x = element_text(angle = 60, hjust = 0.9))+
  scale_x_date(date_labels="%b-%y", breaks="1 month")+
  geom_text(aes(label = n), hjust = 0.3, vjust = -0.3)
```

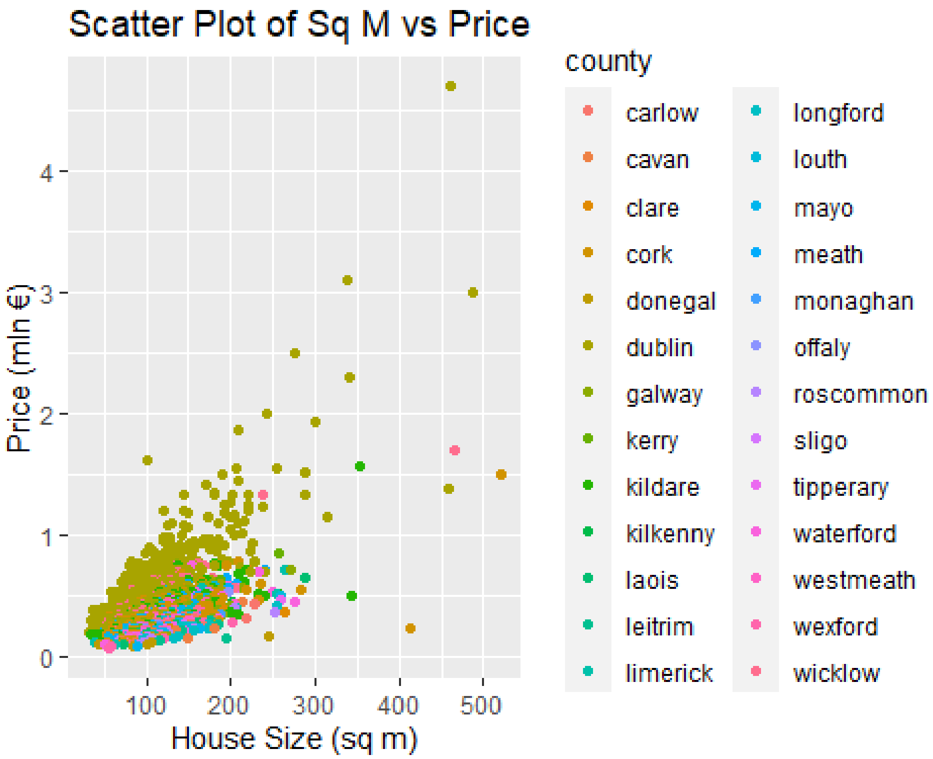
```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## ℹ Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



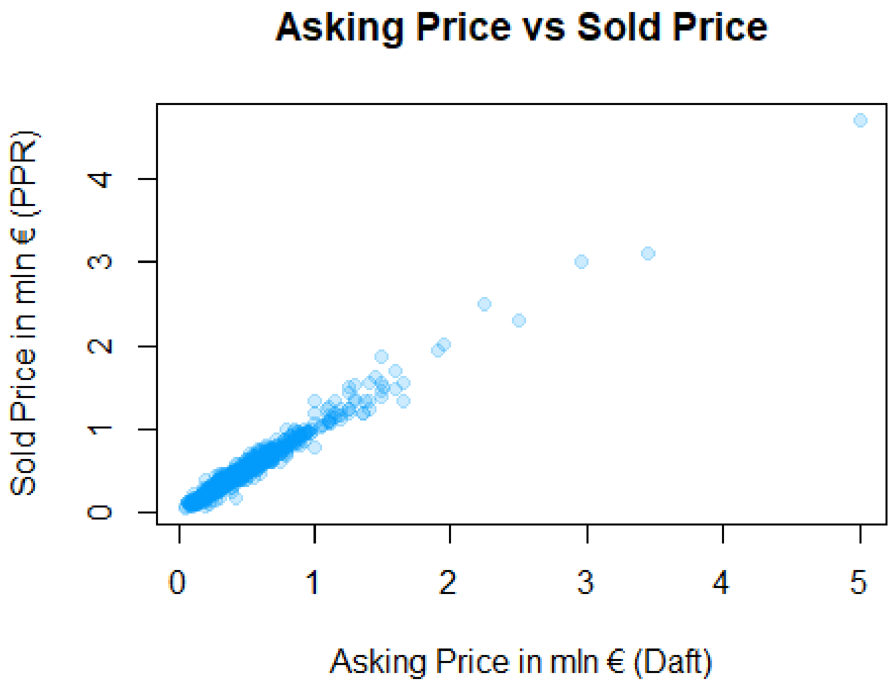
```
table_daft_sticker <- table(houses$daft_sticker)
table_daft_sticker

## 
## ENERGY EFFICIENT    REDUCED PRICE    SCHOOL NEARBY     SOUTH FACING 
##               14                2               15               10 
##  SPACIOUS GARDEN  VIEWING ADVISED 
##               12              111

table_ad_info <- table(houses$ad_info)
table_ad_info

## 
## ADVANTAGE   PREMIUM 
##       231         1
```

————- BIVARIATE ANALYSIS ————-

```
#plot(sold_price_eur/1000000 ~ size, data=houses, main = "Scatter Plot of Sq
M vs Price", ylab="Price (mln)", xlab="House Size (Sq M)", pch=19,
col=as.factor(county))

#Changed the extreme outlier that was at 850 sq m to 85 sq m, as the price
was very low and it was a regular semi-D, so it must have been a mistake

ggplot(houses, aes(size, sold_price_eur/1000000, color=county)) +
  geom_point()+
    labs(x = "House Size (sq m)", y = "Price (mln €)",
      title ="Scatter Plot of Sq M vs Price")

## Warning: Removed 357 rows containing missing values (`geom_point()`).
```

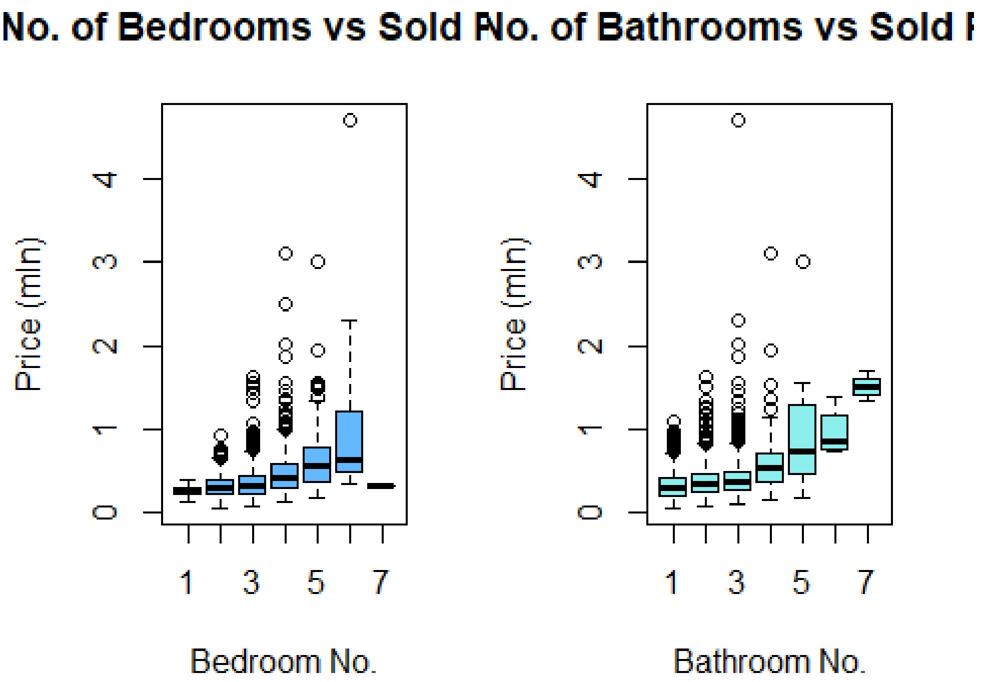


```
plot(I(sold_price_eur/1000000) ~ I(asking_price_eur/1000000), data=houses,
main = "Asking Price vs Sold Price", ylab="Sold Price in mln € (PPR)",
xlab="Asking Price in mln € (Daft)", pch=19, col=alpha("#0099f9", 0.2))
```

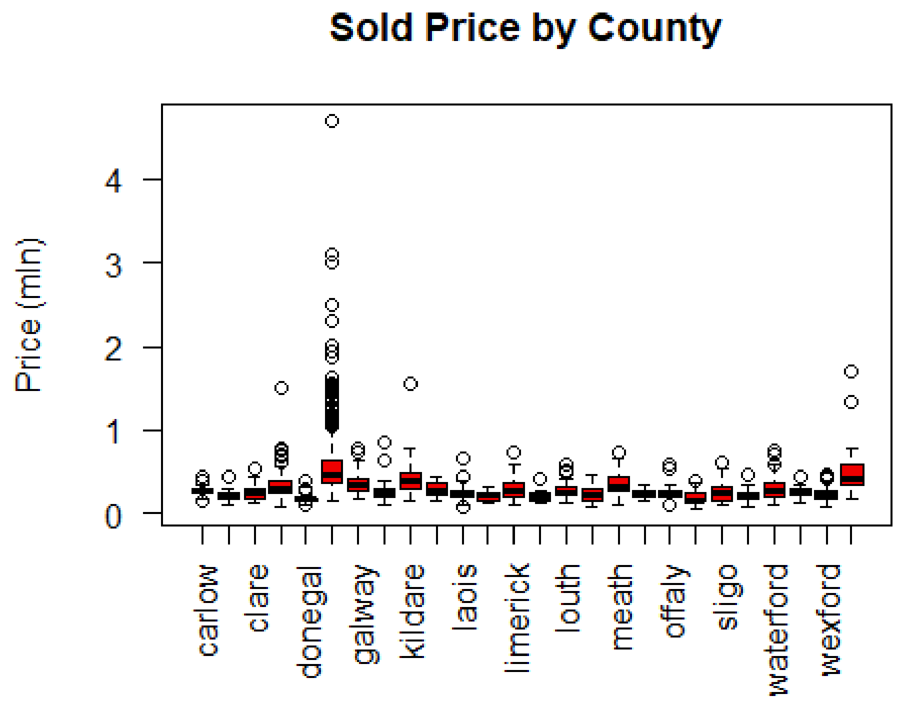


There is an extreme outlier at €5,000,000, which is a verified listing at 26 Herbert Park, Ballsbridge, Dublin 4 https://www.irishtimes.com/property/residential/2023/04/27/crampton-built-home-at-herbert-park-a-rare-offering-for-5m/

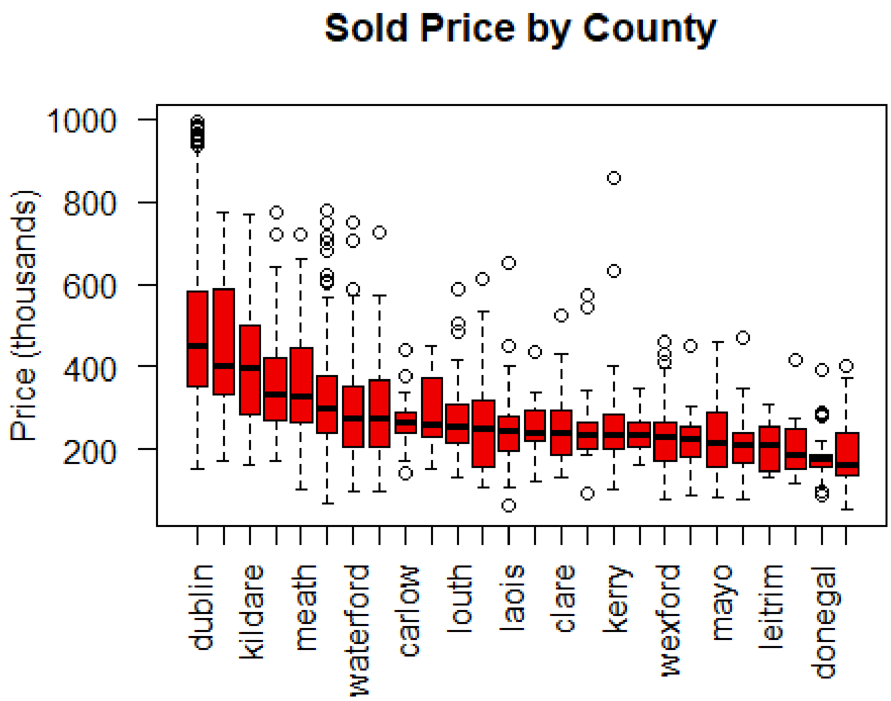
```
par(mfrow=c(1,2))
boxplot(I(sold_price_eur/1000000) ~ bed_no, data=houses, main = "No. of
Bedrooms vs Sold Price", xlab="Bedroom No.", ylab="Price (mln)",
col="steelblue1")
boxplot(I(sold_price_eur/1000000) ~ bath_no, data=houses, main = "No. of
Bathrooms vs Sold Price", xlab="Bathroom No.", ylab="Price (mln)",
col="darkslategray2")
```



```
boxplot(data=houses, sold_price_eur/1000000 ~ county, col="red2", las=2,
ylab="Price (mln)", xlab=NULL, main="Sold Price by County")
```

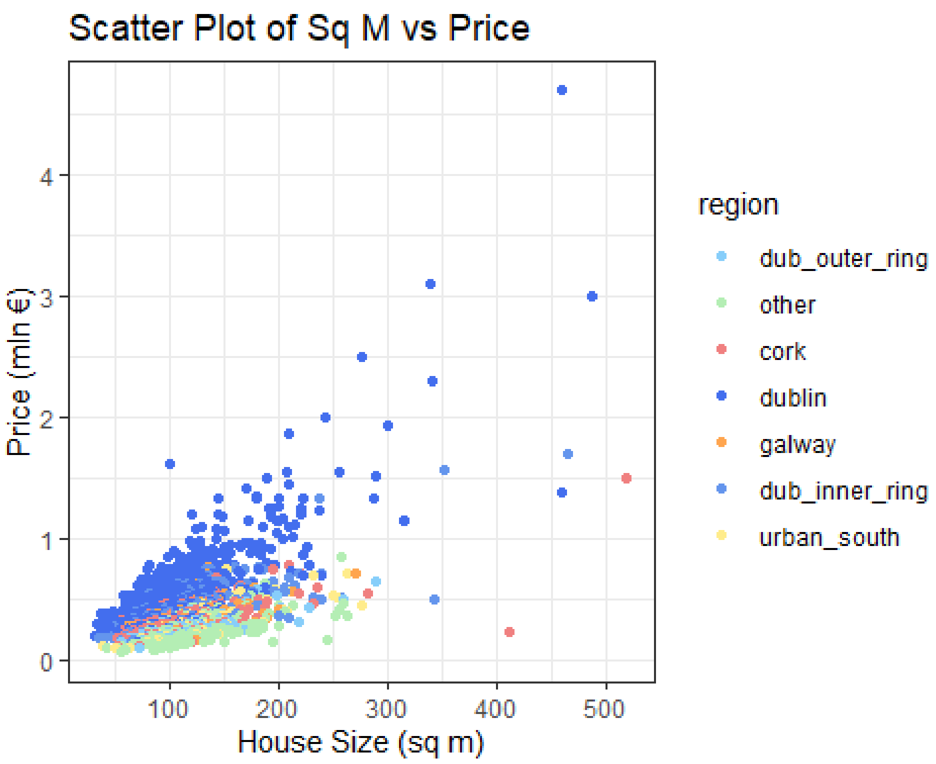


```
# Zooming in on the houses below €1mln and reordering boxplots from the
highest median to the lowest
houses_below1m_ordered <- with(houses_below1m, reorder(county,
sold_price_eur, median, decreasing=TRUE, na.rm=T))
boxplot((houses_below1m$sold_price_eur/1000) ~ houses_below1m_ordered,
col="red2", las=2, ylab="Price (thousands)", xlab=NA, main="Sold Price by
County")
```



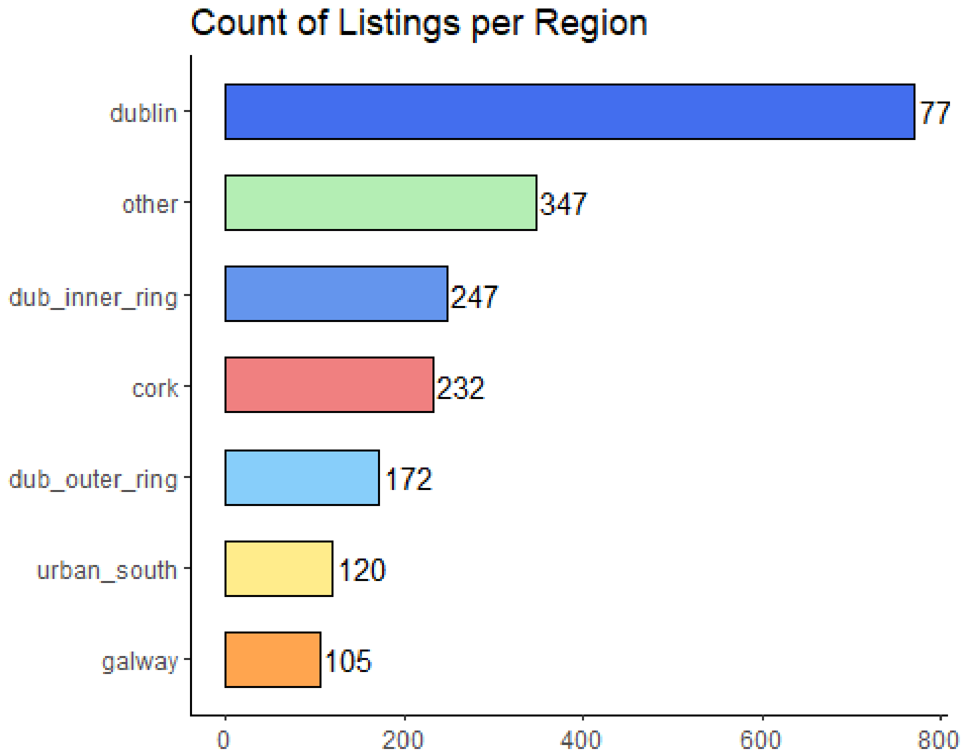
```
houses_regions <- houses %>%
  mutate(region = county) %>%
   group_by(region = fct_collapse(county,
      "dublin" = c("dublin"),
      "cork" = c("cork"),
      "galway" = c("galway"),
      "east_coast" = c("wicklow", "kildare", "meath"),
      "south_coast" = c("waterford", "kerry", "wexford"),
      "west_coast" = c("limerick", "clare", "mayo"),
      "north_coast" = c("louth", "sligo", "leitrim", "donegal"),
      "midlands" = c("carlow", "kilkenny", "laois", "westmeath", "offaly",
"monaghan", "cavan", "tipperary", "longford", "roscommon"))) %>%
  relocate(region, .after=county)

print(houses_regions)

## # A tibble: 1,994 × 21
## # Groups:   region [8]
##       id full_address     house_number street_name locality1 locality2
```

```
locality3
##    <dbl> <chr>            <chr>        <chr>       <chr>     <chr>
<chr>
##  1   780 26 herbert park… 26           herbert pa… ballsbri… <NA>
<NA>
##  2   763 60 ailesbury ro… 60           ailesbury … ballsbri… <NA>
<NA>
##  3  1017 35 abbotts hill… 35           abbotts hi… <NA>      <NA>
<NA>
##  4   764 1 argyle road d… 1            argyle road donnybro… <NA>
<NA>
##  5  1036 4 willow bank m… 4            willow bank <NA>      <NA>
<NA>
##  6   772 135 strand road… 135          strand road sandymou… <NA>
<NA>
##  7   957 24 corrig avenu… 24           corrig ave… <NA>      <NA>
<NA>
##  8   859 159 templeogue … 159          templeogue… terenure  <NA>
<NA>
##  9  1969 54 eagle valley… 54           eagle vall… <NA>      <NA>
<NA>
## 10   683 17 lad lane upp… 17           lad lane u… <NA>      <NA>
<NA>
## # ℹ 1,984 more rows
## # ℹ 14 more variables: city_town <chr>, county <chr>, region <fct>,
## #   daft_sticker <chr>, ad_info <chr>, date_of_sale <date>,
## #   sold_price_eur <dbl>, asking_price_eur <dbl>, price_diff <dbl>,
## #   bed_no <dbl>, bath_no <dbl>, house_type <chr>, size <dbl>,
## #   price_per_sqm <dbl>

houses_regions_new <- houses %>%
  mutate(region = county) %>%
   group_by(region = fct_collapse(county,
      "dublin" = c("dublin"),
      "cork" = c("cork"),
      "galway" = c("galway"),
      "dub_inner_ring" = c("wicklow", "kildare", "meath"),
      "dub_outer_ring" = c("louth", "westmeath", "offaly", "laois",
"carlow"),
      "urban_south" = c("limerick", "waterford"),
      "other" = c("cavan", "clare", "donegal", "kerry", "kilkenny",
"leitrim", "longford", "mayo", "monaghan", "roscommon", "sligo", "tipperary",
"wexford"))) %>%
  relocate(region, .after=county)

cols_region <- c("dublin"="royalblue2",
                "cork"="lightcoral",
                "galway"="tan1",
                "dub_inner_ring"="cornflowerblue",
                "dub_outer_ring"="lightskyblue",
                "urban_south"="lightgoldenrod1",
                "other"="darkseagreen2")

ggplot(houses_regions_new, aes(size, sold_price_eur/1000000, color=region)) +
  geom_point()+
    labs(x = "House Size (sq m)", y = "Price (mln €)",
      title ="Scatter Plot of Sq M vs Price")+
  scale_color_manual(values=cols_region)+
 theme_bw()

## Warning: Removed 357 rows containing missing values (`geom_point()`).
```

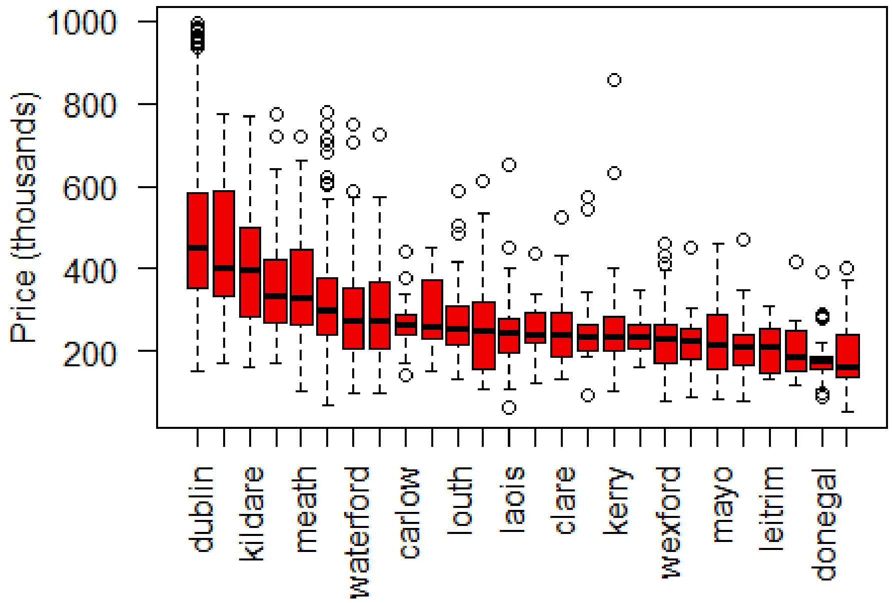


```
table_region <- table(houses_regions$region)
table_region <- table_region[order(table_region, decreasing=FALSE)]
table_region

## 
##      galway north_coast  west_coast south_coast        cork    midlands 
##         105         117         128         157         232         237 
##  east_coast      dublin 
##         247         771 

table_region_new <- table(houses_regions_new$region)
table_region_new <- sort(table_region_new)
table_region_new

## 
##         galway    urban_south dub_outer_ring           cork dub_inner_ring 
##            105            120            172            232            247
##          other         dublin
##            347            771
```

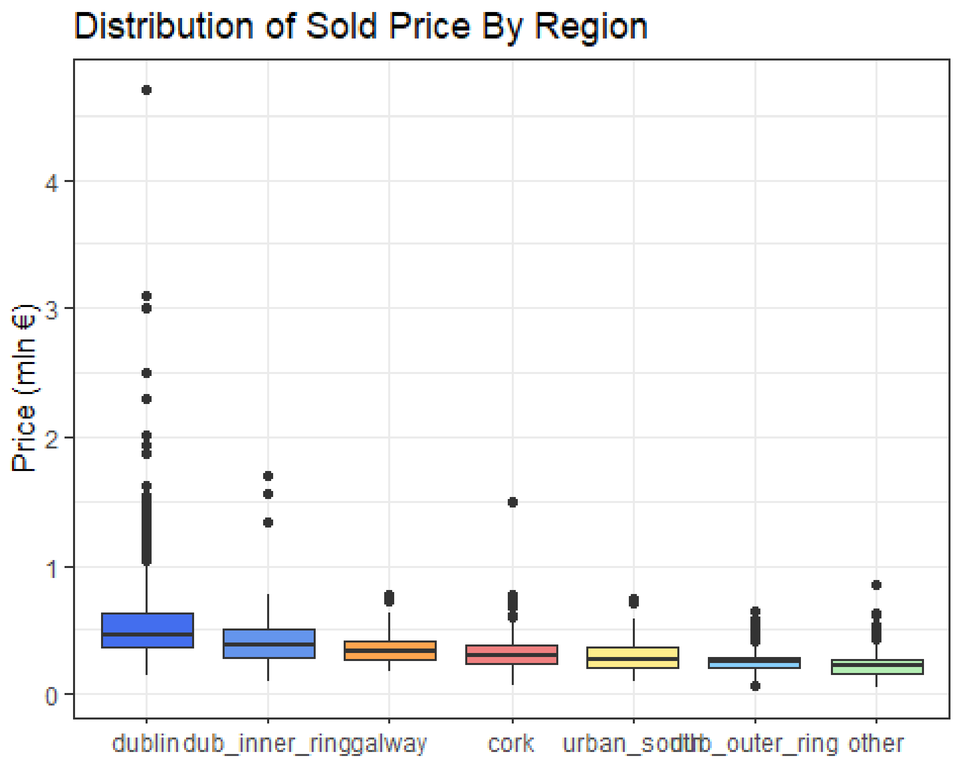
```
regions_df <- data.frame(table_region_new)
#regions_df

ggplot(regions_df, aes(x=Freq, y=Var1, fill=Var1)) +
  geom_col(color = "black", width = 0.6) +
  labs(title="Count of Listings per Region", x=NULL, y=NULL) +
  geom_text(aes(label = Freq), hjust = -0.1)+
  scale_fill_manual(values=cols_region)+
  theme_classic()+
  theme(legend.position="none")
```



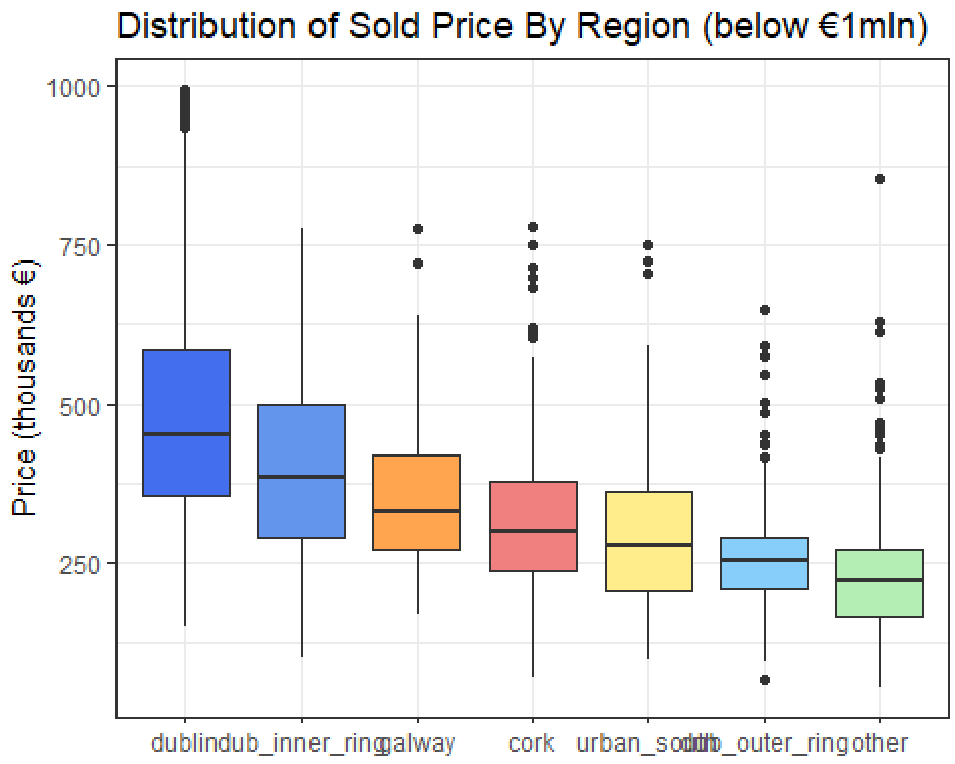
```
#houses_regions_ordered <- with(houses_regions_new, reorder(region,
sold_price_eur, median, decreasing=TRUE, na.rm=T))
#boxplot(data=houses_regions_new, sold_price_eur/1000 ~ region,
col=cols_region, las=2, ylab="Price (thousands)", xlab=NA)

houses_below1m_ordered <- with(houses_below1m, reorder(county,
sold_price_eur, median, decreasing=TRUE, na.rm=T))
boxplot((houses_below1m$sold_price_eur/1000) ~ houses_below1m_ordered,
col="red2", las=2, ylab="Price (thousands)", xlab=NA)
```



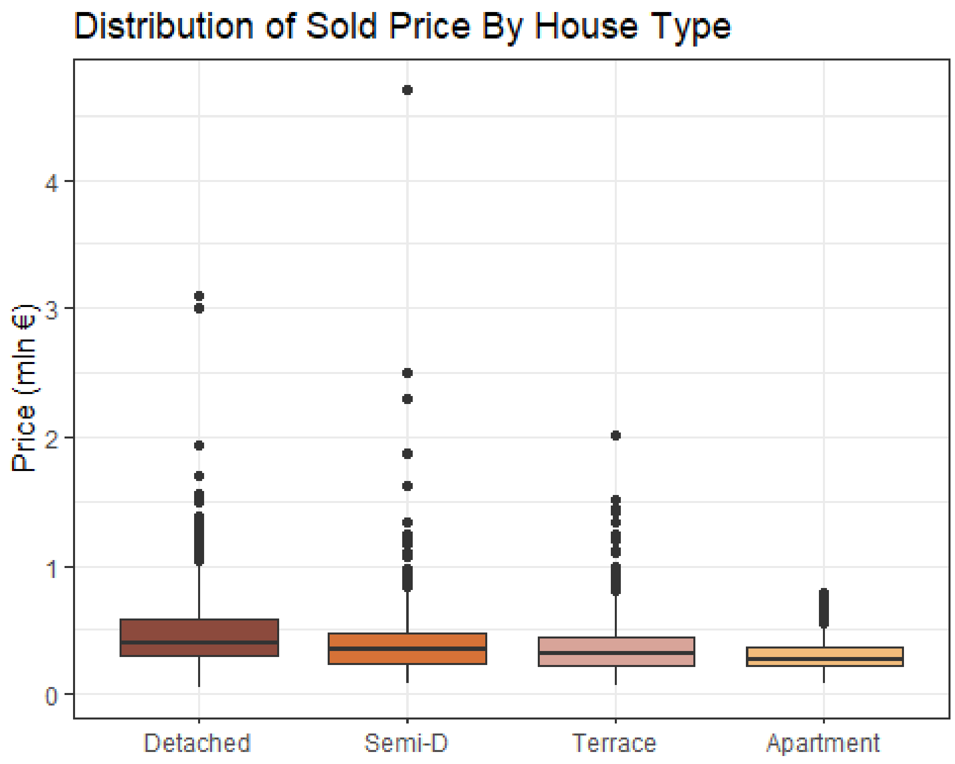
```r
ggplot(houses_regions_new, aes(x=reorder(region, -sold_price_eur),
y=sold_price_eur/1000000))+
  geom_boxplot(fill= c("dublin"="royalblue2",
                       "dub_inner_ring"="cornflowerblue",
                        "galway"="tan1",
                        "cork"="lightcoral",
                        "urban_south"="lightgoldenrod1",
                        "dub_outer_ring"="lightskyblue",
                        "other"="darkseagreen2"))+
  labs(x = NULL, y = "Price (mln €)",
      title ="Distribution of Sold Price By Region")+
  theme_bw()
```



```
houses_regions_below1m <-  subset(houses_regions_new, sold_price_eur<1000000)

ggplot(houses_regions_below1m, aes(x=reorder(region, -sold_price_eur),
y=sold_price_eur/1000, fill=region))+
  geom_boxplot()+
  scale_fill_manual(values= c("dublin"="royalblue2",
                       "dub_inner_ring"="cornflowerblue",
                        "galway"="tan1",
                        "cork"="lightcoral",
                        "urban_south"="lightgoldenrod1",
                        "dub_outer_ring"="lightskyblue",
                        "other"="darkseagreen2"), breaks=c("dublin",
                       "dub_inner_ring",
                        "galway",
                        "cork",
                        "urban_south",
                        "dub_outer_ring",
                        "other"))+
  labs(x = NULL, y = "Price (thousands €)",
      title ="Distribution of Sold Price By Region (below €1mln)")+
  theme_bw()+
  theme(legend.position="none")
```

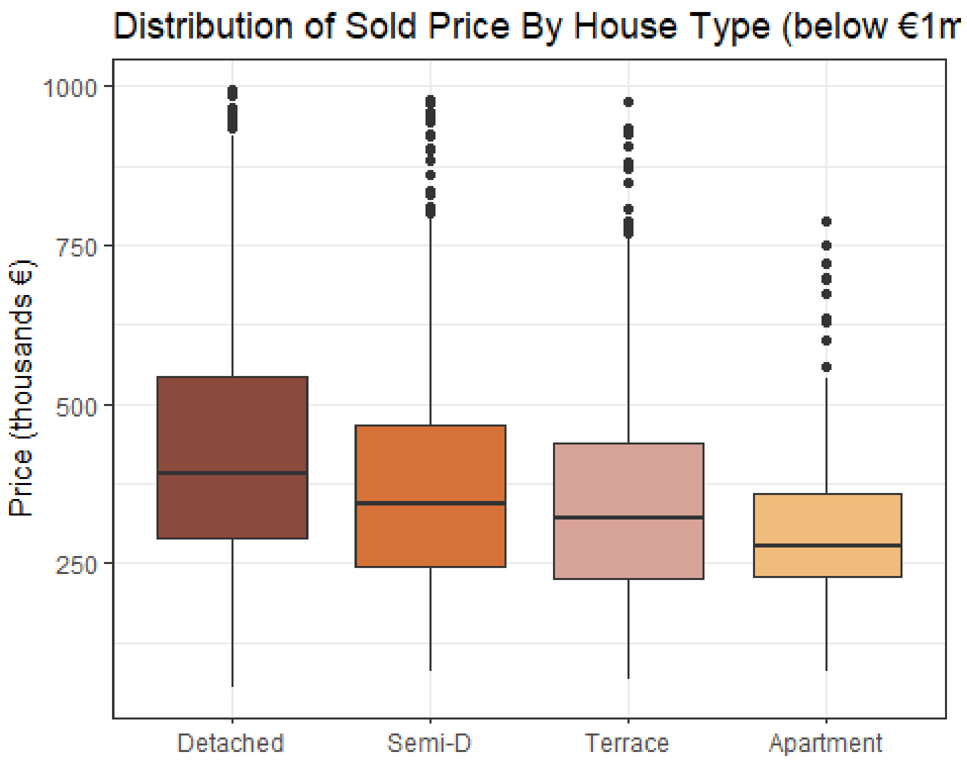


```
ggplot(house_types_collapsed, aes(x=reorder(house_type, -sold_price_eur),
y=sold_price_eur/1000000))+
  geom_boxplot(fill= c("#8c4a3d", "#d67237", "#d8a499", "#f1bb7b"))+
  labs(x = NULL, y = "Price (mln €)",
      title ="Distribution of Sold Price By House Type")+
  theme_bw()
```

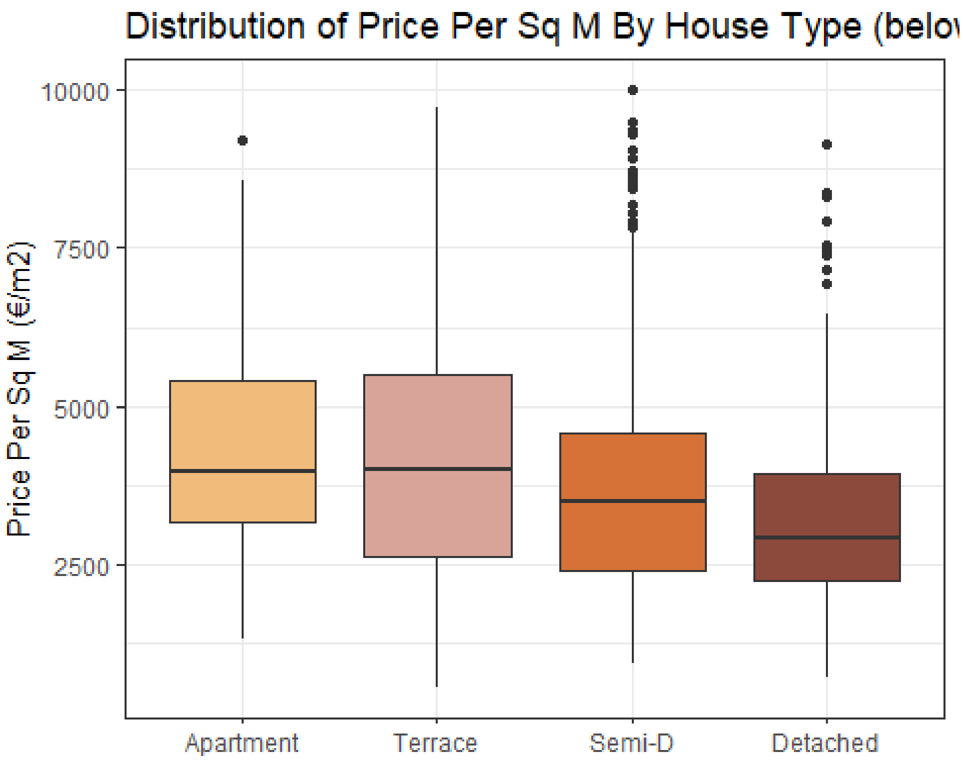


c(“#7294d4”, “#c6cdf7”, “#d8a499”, “#e6a0c4”)

```
house_types_below1m <-  subset(house_types_collapsed, sold_price_eur<1000000)

ggplot(house_types_below1m, aes(x=reorder(house_type, -sold_price_eur),
y=sold_price_eur/1000))+
  geom_boxplot(fill= c("#8c4a3d", "#d67237", "#d8a499", "#f1bb7b"))+
  labs(x = NULL, y = "Price (thousands €)",
      title ="Distribution of Sold Price By House Type (below €1mln)")+
  theme_bw()
```



```
# Defining replacement values

replace_house_types <- c("Duplex"="Apartment",
                         "Bungalow"="Detached",
                         "End of Terrace"="Semi-D",
                         "Townhouse"="Terrace")

# Using str_replace_all() to replace the names in the house_type column
house_types_collapsed <- data.frame(houses)
house_types_collapsed$house_type <-
str_replace_all(house_types_collapsed$house_type, replace_house_types)
#view(house_types_collapsed)

house_types_collapsed %>% subset(price_per_sqm <=10000) %>%
  ggplot(aes(x=reorder(house_type, -price_per_sqm), y=price_per_sqm))+
  geom_boxplot(fill= c("#f1bb7b", "#d8a499","#d67237", "#8c4a3d"))+
  labs(x = NULL, y = "Price Per Sq M (€/m2)",
      title ="Distribution of Price Per Sq M By House Type (below €1mln)")+
  theme_bw()
```

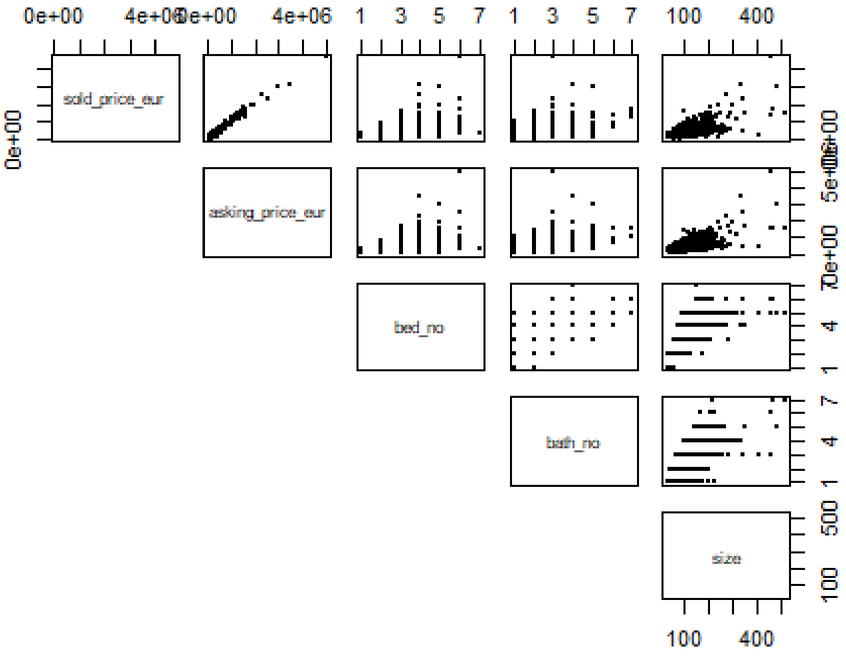


```
# Investigating the median price of houses by county
df_houses <- data.frame(houses)
price_by_county <- df_houses %>%
  group_by(county) %>%
  summarize(county_median = median(sold_price_eur)) %>%
  arrange(desc(county_median))


price_by_county

## # A tibble: 26 × 2
##    county    county_median
##    <chr>             <dbl>
##  1 dublin           467000
##  2 wicklow          420000
##  3 kildare          400000
##  4 galway           332000
##  5 meath            330000
##  6 cork             300000
##  7 waterford        277000
##  8 limerick         275000
##  9 carlow           266250
## 10 kilkenny         260000
## # ℹ 16 more rows

houses_pairs <- houses %>%
 select(sold_price_eur, asking_price_eur, bed_no, bath_no, size)
```

```
  # pairs(houses_pairs)
```

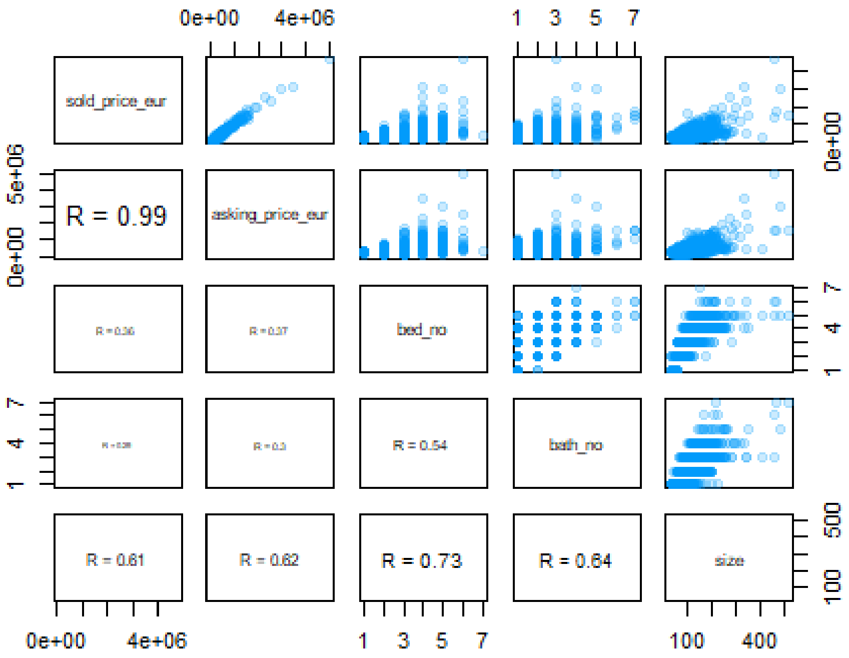
correlations on pairs graph: https://www.sthda.com/english/wiki/scatter-plot-matrices-r-base-graphs

```
pairs(houses_pairs[,1:5], pch = 19,  cex = 0.5,
      lower.panel=NULL)
```



```
# Correlation panel
panel_cor <- function(x, y){
    usr <- par("usr"); on.exit(par(usr))
    par(usr = c(0, 1, 0, 1))
    r <- round(cor(x, y, use="pairwise"), digits=2) # added use="pairwise" to omit the NA values in Size and Bath No.
    txt <- paste0("R = ", r)
    cex_cor <- 0.8/strwidth(txt)
    text(0.5, 0.5, txt, cex = cex_cor * r)
}
# Customize upper panel
upper_panel<-function(x, y){
  points(x,y, pch = 19, col = alpha("#0099f9", 0.2))
}
# Create the plots
pairs(houses_pairs[,1:5],
      lower.panel = panel_cor,
      upper.panel = upper_panel)
```

```
## Warning in par(usr): argument 1 does not name a graphical parameter

## Warning in par(usr): argument 1 does not name a graphical parameter

## Warning in par(usr): argument 1 does not name a graphical parameter

## Warning in par(usr): argument 1 does not name a graphical parameter

## Warning in par(usr): argument 1 does not name a graphical parameter

## Warning in par(usr): argument 1 does not name a graphical parameter

## Warning in par(usr): argument 1 does not name a graphical parameter

## Warning in par(usr): argument 1 does not name a graphical parameter

## Warning in par(usr): argument 1 does not name a graphical parameter

## Warning in par(usr): argument 1 does not name a graphical parameter
```


0e+00
4e+06
sold_price_eur
0e+00
R = 0.99
asking_price_eur
0e+00
5e+06
R = 0.36
R = 0.37
bed_no
R = 0.28
R = 0.3
R = 0.54
bath_no
R = 0.61
R = 0.62
R = 0.73
R = 0.64
size
0e+00
4e+06