

Statistics for Computer Science

Assignment 1

Kanitha Chim

501453

Field of Study Software System and Service Management

Faculty of Informatics
Masaryk University

April 12, 2020

Exercise 1

1. Create your own R-functions for calculating estimates of the following characteristics: sample mean, sample five number summary, sample skewness, sample kurtosis, sample variance, sample standard deviation, sample range, sample decile range, sample 0.1-trimmed average and sample 0.1-trimmed variance. Use Cramér's estimates for skewness and kurtosis. Do not use in-built functions `min()`, `max()`, `mean()`, `quantile()`, `var()`, `sd()`, `range()` or functions from external libraries.

Implementation in R

```
1  #function to calculate minimum
2  cal_minimum <- function(x){
3    x <- sort(x)
4    min_val <- x[1]
5    return(min_val)
6  }
7
8  #function to calculate maximum
9  cal_maximum <- function(x){
10   x <- sort(x)
11   max_val <- tail(x,1)
12   return(max_val)
13 }
14
15 #median, first quartile, thrid quartile, quartile_type should be, one,
   two(median), or three
16 cal_quartile <- function(x, quartile_type){
17   x <- sort(x)
18   result <- 0
19
20   if(quartile_type == 'two'){
21     if(length(x) %% 2 == 0){
22       return((x[length(x)/2] + x[length(x)+1/2])/2)
23     }else{
24       return(x[length(x)+1/2])
25     }
26   }else if(quartile_type == 'one'){
27     position <- (length(x) + 1)/4
28     if (floor(position) == ceiling(position)){
29       return(x[position])
30     }else{
31       return((x[floor(position)] + x[ceiling(position)])/2)
32     }
33   }else if(quartile_type == 'three'){
34     position <- (3*(length(x) + 1))/4
35     if (floor(position) == ceiling(position)){
36       return(x[position])
```

```
37     }else{
38         return((x[floor(position)] + x[ceiling(position)])/2)
39     }
40 }
41 }
42
43 #Sample mean
44 cal_mean <- function(x){
45     result <- sum(x) / length(x)
46     return(result)
47 }
48
49 #Sample five number summary
50 cal_five_num_summary <- function(x){
51     min <- cal_minimum(x)
52     q1 <- cal_quartile(x, 'one')
53     q2 <- cal_quartile(x, 'two')
54     q3 <- cal_quartile(x, 'three')
55     max <- cal_maximum(x)
56     return(c(min, q1, q2, q3, max))
57 }
58
59 #Sample Skewness
60 cal_coef_skew <- function(x){
61     s <- sqrt(cal_sample_variance(x))
62     mean <- cal_mean(x)
63     total <- 0
64     for(i in x){
65         total = total + (i - mean)^3
66     }
67     return(total/(s^3 * length(x)))
68 }
69
70 #Sample Kurtosis
71 cal_coef_kurt <- function(x){
72     s <- sqrt(cal_sample_variance(x))
73     mean <- cal_mean(x)
74     total <- 0
75     for(i in x){
76         total = total + (i - mean)^4
77     }
78     total = total/(s^4 * length(x))
79     return(total -3)
80 }
81
82 #Sample Variance
83 cal_sample_variance <- function(x){
84     mean <- cal_mean(x)
85     n_minus_one <- length(x) - 1
```

```
86   total <- 0
87   for(i in x){
88     total = total + (i - mean)^2
89   }
90   return(total/n_minus_one)
91 }
92
93 #Sample Standard Deviation
94 cal_sd <- function(x){
95   sample_variance <- cal_sample_variance(x)
96   return(sqrt(sample_variance))
97 }
98
99 #Sample Range
100 cal_sample_range <- function(x){
101   return(cal_maximum(x) - cal_minimum(x))
102 }
103
104 #Sample Decile Range
105 cal_sample_decile_range <- function(x){
106   x_0.9 <- 90/100 * (length(x) +1)
107   x_0.1 <- 10/100 * (length(x) +1)
108
109   if (floor(x_0.9) == ceiling(x_0.9)){
110     x_0.9 = x_0.9
111   }else{
112     x_0.9 = (x[floor(x_0.9)] + x[ceiling(x_0.9)])/2
113   }
114
115   if (floor(x_0.1) == ceiling(x_0.1)){
116     x_0.1 = x_0.1
117   }else{
118     x_0.1 = (x[floor(x_0.1)] + x[ceiling(x_0.1)])/2
119   }
120   return(x_0.9 - x_0.1)
121 }
122
123 #Sample 0.1 trimmed average
124 cal_sample_0.1_trimmed_average <- function(x){
125   g <- abs(0.1 * length(x))
126   n <- length(x)
127   total <- 0
128   for(i in seq(g+1, n-g, 1)){
129     total <- total + x[i]
130   }
131   return(total/(n - 2*g))
132 }
133
134 #Sample 0.1 trimmed variance
```

```

135 cal_sample_0.1_trimmed_variance <- function(x){
136   g <- abs(0.1 * length(x))
137   n <- length(x)
138   x_gt <- cal_sample_0.1_trimmed_average(x)
139   total <- 0
140   for(i in seq(g+1, n-g, 1)){
141     total <- total + (x[i] - x_gt)^2
142   }
143   return(total/(n - 2*g - 1))
144 }
145
146 library(xtable)

```

2. Separately for each population calculate sample size and these characteristics of maximum cranial breadth. Print them in a table with values rounded to 4 decimal places.

Implementation in R

```

147 setwd('/home/jane/Documents/SSME (Service Development Management)/Term2/
    MV013-Statistics for Computer Science/Assignment1')
148 Howell <- read.csv('Howell.csv')
149 Howell.Male <- Howell[Howell$Sex == 'M', ]
150 Howell.Male.Australi <- Howell.Male[(Howell.Male$Population == 'AUSTRALI
    '), ]
151 Howell.Male.Peru <- Howell.Male[(Howell.Male$Population == 'PERU'), ]
152
153 #Working on population of 'AUSTRALI'
154 Australi.sample_size <- length(Howell.Male.Australi$XCB)
155 Australi.sample_mean <- cal_mean(Howell.Male.Australi$XCB)
156 Australi.sample_five_num_summary <- cal_five_num_summary(Howell.Male.
    Australi$XCB)
157 Australi.sample_skewness <- cal_coef_skew(Howell.Male.Australi$XCB)
158 Australi.sample_kurtosis <- cal_coef_kurt(Howell.Male.Australi$XCB)
159 Australi.sample_variance <- cal_sample_variance(Howell.Male.Australi$XCB
    )
160 Australi.sample_sd <- cal_sd(Howell.Male.Australi$XCB)
161 Australi.sample_range <- cal_sample_range(Howell.Male.Australi$XCB)
162 Australi.sample_decile_range <- cal_sample_decile_range(Howell.Male.
    Australi$XCB)
163 Australi.sample_0.1_trimmed_average <- cal_sample_0.1_trimmed_average(
    Howell.Male.Australi$XCB)
164 Australi.sample_0.1_trimmed_variance <- cal_sample_0.1_trimmed_variance(
    Howell.Male.Australi$XCB)
165
166 #Working on population of 'PERU'
167 Peru.sample_size <- length(Howell.Male.Peru$XCB)
168 Peru.sample_mean <- cal_mean(Howell.Male.Peru$XCB)
169 Peru.sample_five_num_summary <- cal_five_num_summary(Howell.Male.Peru$
    XCB)
170 Peru.sample_skewness <- cal_coef_skew(Howell.Male.Peru$XCB)

```

```

171 Peru.sample_kurtosis <- cal_coef_kurt(Howell.Male.Peru$XCB)
172 Peru.sample_variance <- cal_sample_variance(Howell.Male.Peru$XCB)
173 Peru.sample_sd <- cal_sd(Howell.Male.Peru$XCB)
174 Peru.sample_range <- cal_sample_range(Howell.Male.Peru$XCB)
175 Peru.sample_decile_range <- cal_sample_decile_range(Howell.Male.Peru$XCB
  )
176 Peru.sample_0.1_trimmed_average <- cal_sample_0.1_trimmed_average(Howell
  .Male.Peru$XCB)
177 Peru.sample_0.1_trimmed_variance <- cal_sample_0.1_trimmed_variance(
  Howell.Male.Peru$XCB)
178
179 Howell.characteristics <- data.frame(Australi=c(Australi.sample_size,
  Australi.sample_mean, Australi.sample_five_num_summary, Australi.
  sample_skewness,
180           Australi.sample_kurtosis, Australi.sample_
  variance, Australi.sample_sd, Australi.sample
  _range,
181           Australi.sample_decile_range, Australi.sample_
  0.1_trimmed_average, Australi.sample_0.1_
  trimmed_variance),
182           Peru=c(Peru.sample_size, Peru.sample_mean, Peru.sample
  _five_num_summary, Peru.sample_skewness,
183           Peru.sample_kurtosis, Peru.sample_variance, Peru
  .sample_sd, Peru.sample_range,
184           Peru.sample_decile_range, Peru.sample_0.1_
  trimmed_average, Peru.sample_0.1_trimmed_
  variance),
185           row.names = c('Size', 'Mean', 'Minimum', 'Q1', 'Q2', '
  Q3', 'Maximum', 'Skewness', 'Kurtosis', 'Variance',
186           'Standard Deviation', 'Range', 'Decile Range', '
  Sampe 0.1 trimmed average', 'Sample 0.1
  trimmed variance'))
187
188 library(xtable)

```

3. Create boxplots of maximum cranial breadth of each population side by side in one figure. Set the width of boxes to be proportional to sample sizes, add notches and arithmetic averages for both groups.

Implementation in R

```

189 max.len = max(length(Howell.Male.Australi$XCB), length(Howell.Male.Peru$
  XCB))
190 x=c(Howell.Male.Australi$XCB, rep(NA, max.len - length(Howell.Male.
  Australi$XCB)))
191 y=c(Howell.Male.Peru$XCB, rep(NA, max.len - length(Howell.Male.Peru$XCB)
  ))
192 Populations <- c('Australi', 'Peru')
193 values <- c(x, y)
194 df <- data.frame(Populations, values)

```

```

195
196 library(ggplot2)
197 boxplot <- ggplot(df, aes(x=Populations, y=values, fill=Populations)) +
198   geom_boxplot(notch = TRUE, width=(55/110)) +
199   scale_fill_manual(values=c("lightblue", "lightgreen")) +
200   ylab('Maximum cranial breadth') +
201   xlab('Populations') +
202   annotate("text", x=1, y=Australi.sample_mean, label= round(
203     Australi.sample_mean, 4)) +
204   annotate("text", x=2, y=Peru.sample_mean, label= round(Peru.
205     sample_mean, 4)) +
206   ggtitle('Boxplot of maximum cranial breadth') +
207   theme(plot.title = element_text(hjust = 0.5))

```

4. Separately for each population create histogram of maximum cranial breadth. Make sure that the histograms can be easily compared (without using back-to-back histogram).

Implementation in R

```

206 Australi.hist <- ggplot(data=data.frame(Australi=c(Howell.Male.Australi$
207   XCB)), aes(x=Howell.Male.Australi$XCB)) +
208   geom_histogram(color="darkblue", fill="lightblue",
209     binwidth = 1) +
210   xlab('Maximum cranial breadth of Australi') +
211   ylab('Frequency') +
212   scale_x_continuous(breaks = seq(124, 144, 2), lim = c
213     (122, 146)) +
214   geom_text(stat = 'count', aes(label = ..count.., vjust =
215     -0.5))
216
217 Peru.hist <- ggplot(data=data.frame(Peru=c(Howell.Male.Peru$XCB)), aes(x
218   =Howell.Male.Peru$XCB)) +
219   geom_histogram(color="darkgreen", fill="lightgreen",
220     binwidth = 1) +
221   xlab('Maximum cranial breadth of Peru') +
222   ylab('Frequency') +
223   scale_x_continuous(breaks = seq(129, 149, 2), lim = c(128,
224     150)) +
225   geom_text(stat = 'count', aes(label = ..count.., vjust =
226     -0.5))

```

5. Create normal qq-plot of maximum cranial breadth for each population.

Implementation in R

```

219 Australi.qq <- ggplot(data=data.frame(Australi=c(Howell.Male.Australi$
220   XCB))) +
221   aes(sample= Australi) +
222   geom_qq(distribution = qnorm) +
223   geom_qq_line(line.p = c(0.25, 0.75), col = "lightblue")
224   +

```

```

223         ylab('Maximum cranial breadth for Australi') +
224         ggtitle('QQ-Plot of maximum cranial breadth of Australi'
225                ) +
226         theme(plot.title = element_text(hjust = 0.5))
227 Peru.qq <- ggplot(data=data.frame(Peru=c(Howell.Male.Peru$XCB))) +
228         aes(sample= Peru) +
229         geom_qq(distribution = qnorm) +
230         geom_qq_line(line.p = c(0.25, 0.75), col = "lightgreen") +
231         ylab('Maximum cranial breadth for Peru') +
232         ggtitle('QQ-Plot of maximum cranial breadth of Peru') +
233         theme(plot.title = element_text(hjust = 0.5))

```

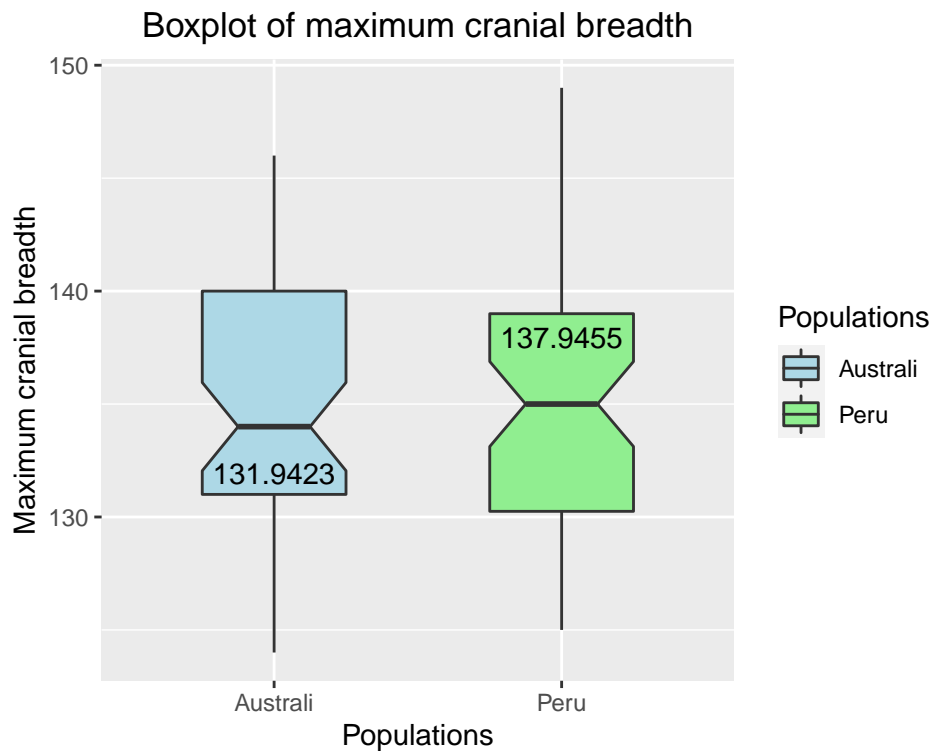
6. Interpretation of results and graphics.

Characteristics of maximum cranial breadth

Table represents the characteristics of maximum cranial breadth. The selected populations are male people from Austali and Peru.

	Australi	Peru
Size	52.0000	55.0000
Mean	131.9423	137.9455
Minimum	124.0000	129.0000
Q1	128.0000	135.0000
Q2	131.0000	138.0000
Q3	134.0000	141.0000
Maximum	144.0000	149.0000
Skewness	0.6436	-0.0037
Kurtosis	-0.3390	0.0616
Variance	26.0554	15.8673
Standard Deviation	5.1045	3.9834
Range	20.0000	20.0000
Decile Range	15.0000	-4.0000
Sampe 0.1 trimmed average	129.4712	138.1364
Sample 0.1 trimmed variance	26.3226	15.1438

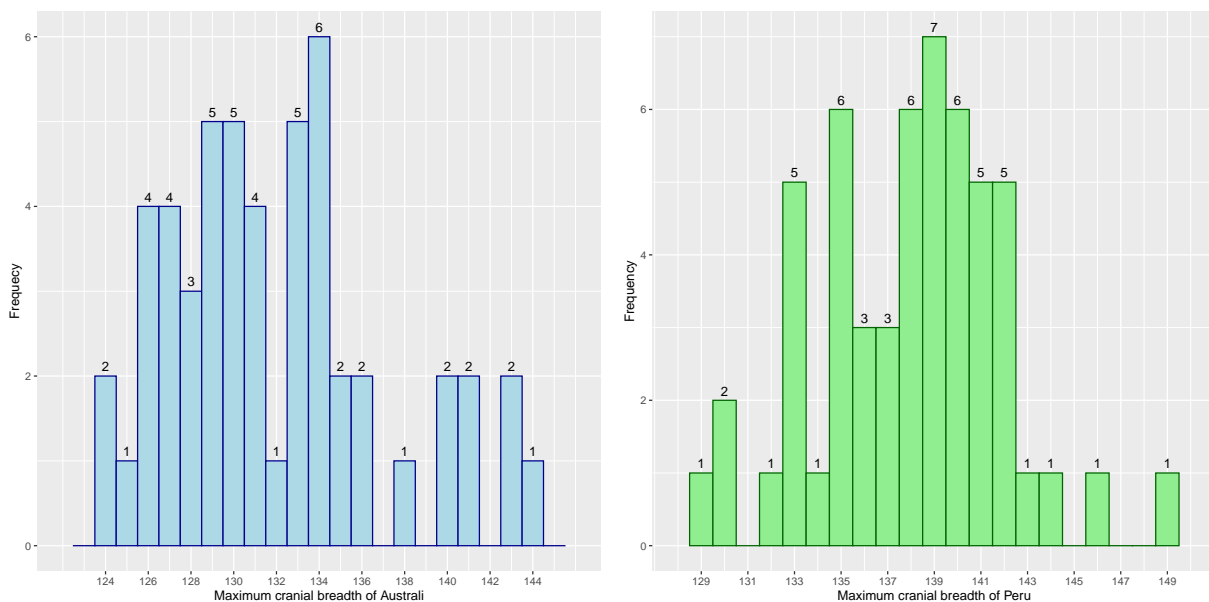
Table 1: Characteristics of maximum cranial breadth



Boxplot represent maximum cranial breadth of Australi and Peru with the arithmetic average of each population placed on each box. X-axis shows the population and y-axis show the number of maximum cranial breadth.

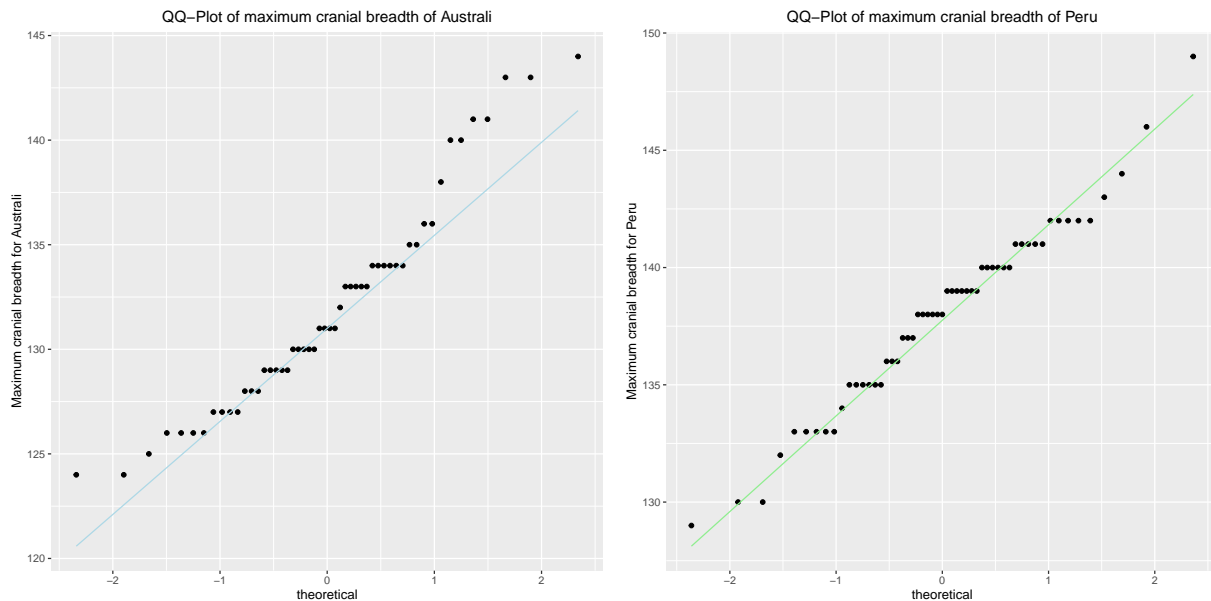
Histograms represent maximum cranial breadth of each population

Histograms show frequency of maximum cranial breadth of Australi(on the left) and Peru(on the right) with the number of frequency on top of each bar. X-axis shows the number of maximum cranial breadth and y-axis shows the frequency.



QQ-plot represent maximum cranial breadth of each population

The left figure shows qq-plot represent maximum cranial breadth of Australi while the right one is for Peru.



Exercise 2

1. Calculate the number of men and women in Spain for each year and print them in a table together with the total population.

Implementation in R

```

234 setwd('/home/jane/Documents/SSME (Service Development Management)/Term2/
    MV013-Statistics for Computer Science/Assignment1')
235 Spanish.province <- read.csv('area_spanish_provinces.csv')
236 Spanish.population <- read.csv('population-spain-1998-2018.csv', sep=";"
    )
237
238 population.2018 <- sum(Spanish.population$males.2018, Spanish.population
    $females.2018)
239 population.2013 <- sum(Spanish.population$males.2013, Spanish.population
    $females.2013)
240 population.2008 <- sum(Spanish.population$males.2008, Spanish.population
    $females.2008)
241 population.2003 <- sum(Spanish.population$males.2003, Spanish.population
    $females.2003)
242 population.1998 <- sum(Spanish.population$males.1998, Spanish.population
    $females.1998)
243
244 total.population <- sum(colSums(Spanish.population[, -1]))

```

```

245
246 population.df <- data.frame(Male=c(sum(Spanish.population$males.1998),
247                                   sum(Spanish.population$males.2003),
248                                   sum(Spanish.population$males.2008), sum(Spanish.
249                                   population$males.2013),
250                                   sum(Spanish.population$males.2018), 0),
251                                   Female=c(sum(Spanish.population$females.1998), sum(
252                                   Spanish.population$females.2003),
253                                   sum(Spanish.population$females.2008), sum(
254                                   Spanish.population$females.2013),
255                                   sum(Spanish.population$females.2018), 0),
256                                   Total=c(population.1998, population.2003, population
257                                   .2008, population.2013, population.2018, total.
258                                   population))
259 row.names(population.df) <- c('1998', '2003', '2008', '2013', '2018', '
260   Total')
261 population.df[6,1] <- sum(population.df$Male)
262 population.df[6,2] <- sum(population.df$Female)

```

2. Display barplot plot of total population of Spain in each of the years, with each bar divided between men and women.

Implementation in R

```

266 library(ggplot2)
267 year <- c(1998, 1998, 2003, 2003, 2008, 2008, 2013, 2013, 2018, 2018)
268 population <- c(sum(Spanish.population$males.1998),
269                 sum(Spanish.population$females.1998),
270                 sum(Spanish.population$males.2003),
271                 sum(Spanish.population$females.2003),
272                 sum(Spanish.population$males.2008),
273                 sum(Spanish.population$females.2008),
274                 sum(Spanish.population$males.2013),
275                 sum(Spanish.population$females.2013),
276                 sum(Spanish.population$males.2018),
277                 sum(Spanish.population$females.2018))
278 options(scipen = 999)
279 gender <- c('Male', 'Female', 'Male', 'Female', 'Male', 'Female', 'Male'
280             , 'Female', 'Male', 'Female' )
281 df.population <- data.frame(year, population, gender)
282 population.barplot<-ggplot(data=df.population, aes(x=year, y=population,
283             fill=gender )) +
284   geom_col() +
285   scale_fill_manual(values=c("lightpink", "lightblue")) +
286   scale_x_continuous(breaks = seq(1998, 2018, 5)) +
287   scale_y_continuous(breaks = seq(0, 50000000, 5000000)) +
288   geom_text(aes(label = sprintf("%.0fk", population/1000)), position =
289             position_stack(0.5)) +
290   theme(axis.text.y=element_text(angle = 45))

```

3. Display barplot of relative proportions of men and women within each province in 2018

Implementation in R

```

278 proportion.province <-c(as.vector(Spanish.population$province), as.
    vector(Spanish.population$province))
279 proportion.gender <- c(rep('Male', 52), rep('Female', 52))
280 proportion.male <- round(Spanish.population$males.2018 / (Spanish.
    population$males.2018 + Spanish.population$females.2018), 4)
281 proportion.female <- round(Spanish.population$females.2018 / (Spanish.
    population$males.2018 + Spanish.population$females.2018), 4)
282
283 proportion.province.gender <- c(proportion.male, proportion.female)
284
285 df.proportion <- data.frame(Provinces=proportion.province, Proportion=
    proportion.province.gender, Gender=proportion.gender)
286 proportion.barplot <- ggplot(data=df.proportion, aes(x=Provinces, y=
    Proportion, fill=Gender)) +
287     geom_col() +
288     theme(axis.text.x=element_text(angle = 90, size=6), plot.title
        = element_text(vjust=0.5, hjust=1)) +
289     scale_fill_manual(values=c("lightpink", "lightblue")) +
290     scale_x_discrete(name= "Provinces") +
291     scale_y_continuous(name= "Population proportion")

```

4. Calculate population density in each province in 1998 and in 2018.

Implementation in R

```

292 province.population.2018 <- Spanish.population$males.2018 + Spanish.
    population$females.2018
293 province.population.1998 <- Spanish.population$males.1998 + Spanish.
    population$females.1998
294 provinces <- sapply(strsplit(as.vector(Spanish.population$province), ",
    "), '[', 1)
295 df <- data.frame(province = provinces, population.1998 = province.
    population.1998, population.2018 =province.population.2018)
296
297 area.provinces <- sapply(strsplit(as.vector(Spanish.province$Province), "
    "), tail, 1)
298 df1 <- data.frame(province = area.provinces, area = Spanish.province$
    Area)
299
300 for(i in 1:52){
301     for(j in 1:52){
302         if(grepl(df1$province[j], df$province[i])){
303             df$area[i] <- df1$area[j]
304         }
305     }
306 }
307
308 df$province <- Spanish.population$province

```

```

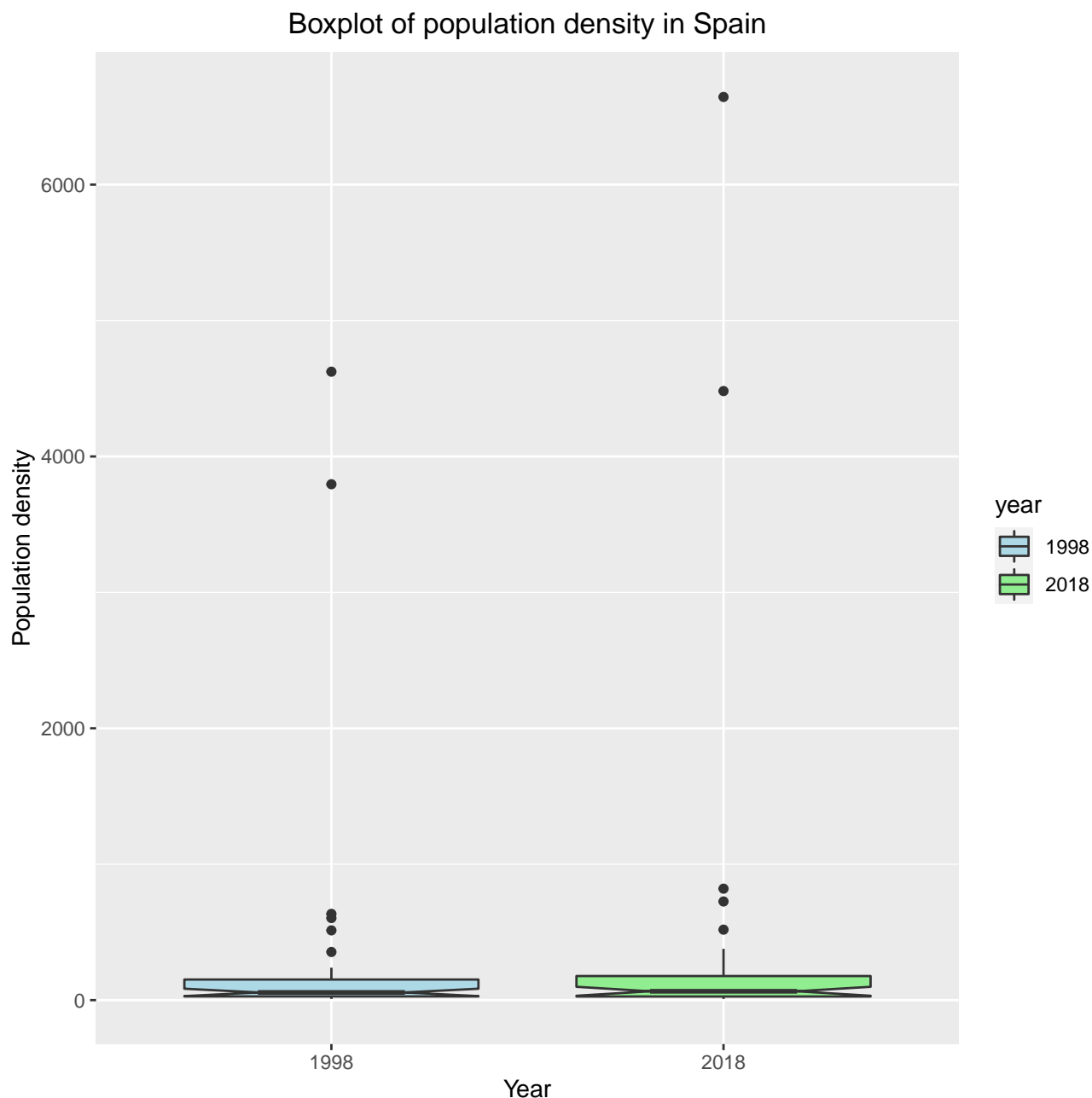
309 df$density.1998 <- round(df$population.1998 / df$area, 4)
310 df$density.2018 <- round(df$population.2018 / df$area, 4)
311
312 density.2018.sample_size <- length(df$density.2018)
313 density.2018.sample_mean <- cal_mean(df$density.2018)
314 density.2018.sample_five_num_summary <- cal_five_num_summary(df$density
    .2018)
315 density.2018.sample_skewness <- cal_coef_skew(df$density.2018)
316 density.2018.kurtosis <- cal_coef_kurt(df$density.2018)
317 density.2018.sample_sd <- cal_sd(df$density.2018)
318
319 density.1998.sample_size <- length(df$density.1998)
320 density.1998.sample_mean <- cal_mean(df$density.1998)
321 density.1998.sample_five_num_summary <- cal_five_num_summary(df$density
    .1998)
322 density.1998.sample_skewness <- cal_coef_skew(df$density.1998)
323 density.1998.kurtosis <- cal_coef_kurt(df$density.1998)
324 density.1998.sample_sd <- cal_sd(df$density.1998)
325
326 density.characteristics <- data.frame("1998"=c(density.1998.sample_size,
    density.1998.sample_mean, density.1998.sample_five_num_summary,
327                                     density.1998.sample_
                                                skewness, density.1998.
                                                kurtosis, density.1998.
                                                sample_sd),
328                                     "2018"=c(density.2018.sample_size,
                                                density.2018.sample_mean,
                                                density.2018.sample_five_num_
                                                summary,
329                                     density.2018.sample_
                                                skewness, density.2018.
                                                kurtosis, density.2018.
                                                sample_sd),
330                                     row.names = c('Size', 'Mean', '
                                                Minimum', 'Q1', 'Q2', 'Q3', '
                                                Maximum', 'Skewness', 'Kurtosis
                                                ', 'Standard Deviation'))
331
332 colnames(density.characteristics) <- c("Year 1998", "Year 2018")
333
334 year.1998 <- rep('1998', 52)
335 year.2018 <- rep('2018', 52)
336 year <- c(year.1998, year.2018)
337 values <- c(df$density.1998, df$density.2018)
338 density.df <- data.frame(year, values)
339
340 density.boxplot <- ggplot(density.df, aes(x=year, y=values, fill=year))
    +
341     geom_boxplot(notch = TRUE) +

```

```

342     scale_fill_manual(values=c("lightblue", "lightgreen")) +
343     ylab('Population density') +
344     xlab('Year') +
345     ggtitle('Boxplot of population density in Spain') +
346     theme(plot.title = element_text(hjust = 0.5))
347 density.boxplot

```



```

348 hist.1998 <- ggplot(data=data.frame('1998'=c(df$density.1998)), aes(x=df
349     $density.1998)) +
350     geom_histogram(color="darkblue", fill="lightblue",
351     binwidth = 50) +
352     xlab('Population density in 1998') +
353     ylab('Frequency') +

```

```

352         scale_x_continuous(breaks = seq(0, 5000, 500))
353
354 hist.2018 <- ggplot(data=data.frame('2018'=c(df$density.2018)), aes(x=df
    $density.2018)) +
355         geom_histogram(color="darkgreen", fill="lightgreen",
            binwidth = 40) +
356         xlab('Population density in 2018') +
357         ylab('Frequency') +
358         scale_x_continuous(breaks = seq(0, 7000, 1000))

```

5. Interpretation of results and graphics.

Table of population in Spain

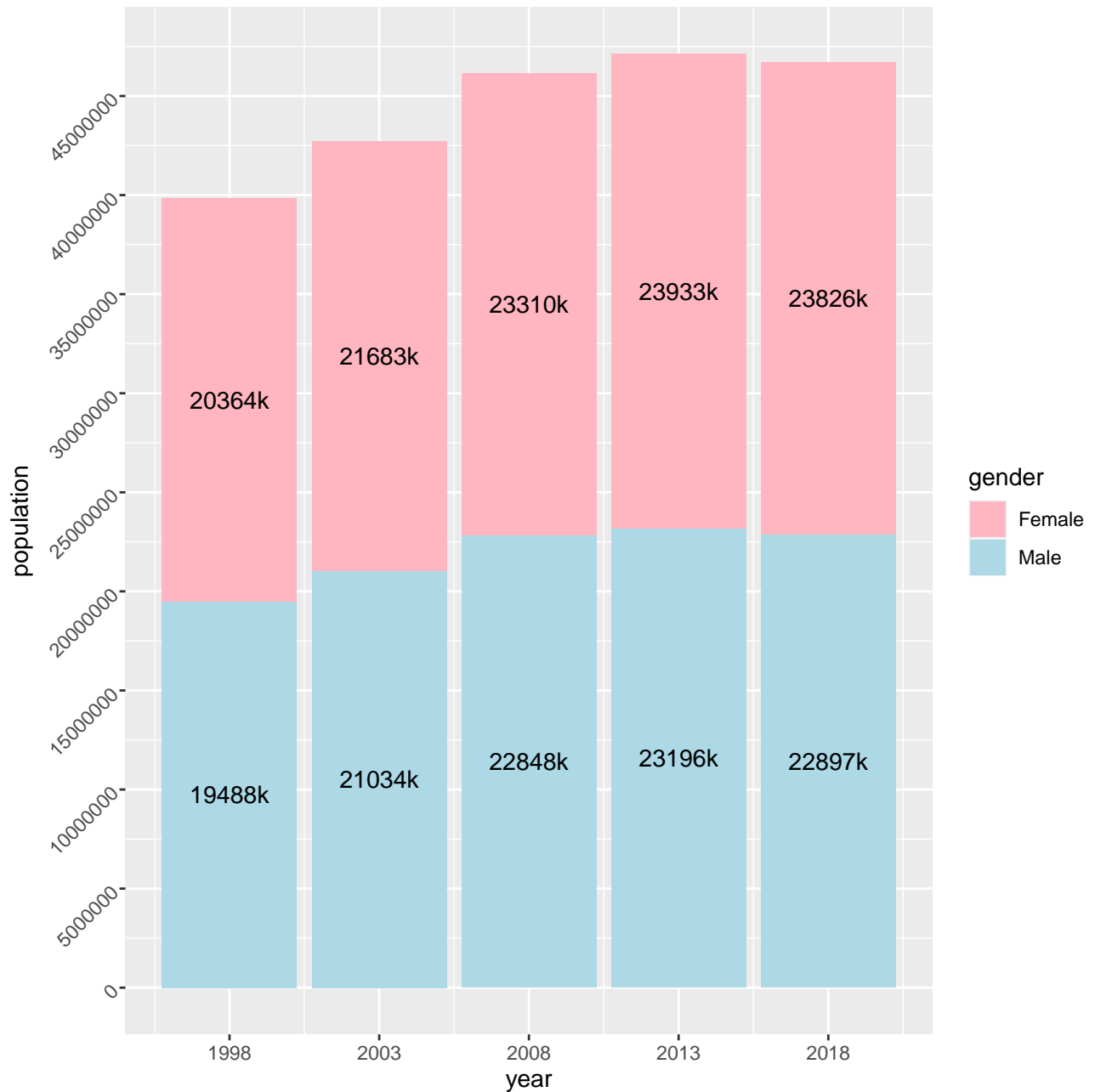
Table represents the population of Spain in 1998, 2003, 2008, 2013 and 2018. It shows the total population based on gender between each year, the total population between each year, and the total population of the 5 selected year.

	Male	Female	Total
1998	19488465	20364186	39852651
2003	21034326	21682738	42717064
2008	22847737	23310085	46157822
2013	23196386	23933397	47129783
2018	22896602	23826378	46722980
Total	109463516	113116784	222580300

Table 2: Population of Spain

Barplot of population in Spain

Barplot of total population in Spain in 1998, 2003, 2008, 2013 and 2018 with bar separated by gender. Male in blue and Female in pink. Each bar has two numbers represents total population based on gender in each year. X-axis shows the years and y-axis shows population. The number of population in Spain slightly increased till 2013 and slightly dropped in 2018.



Barplot of proportions of men and women in Spain (2018)

Barplot represents population proportion of each province in Spain in 2018. X-axis shows provinces in Spain while y-axis show the relative proportion. Each bar is divided between men and women of each province. The blue part is for men and the pink part is for women. The proportion of men and women in each province is almost equal with men are slightly less than women.

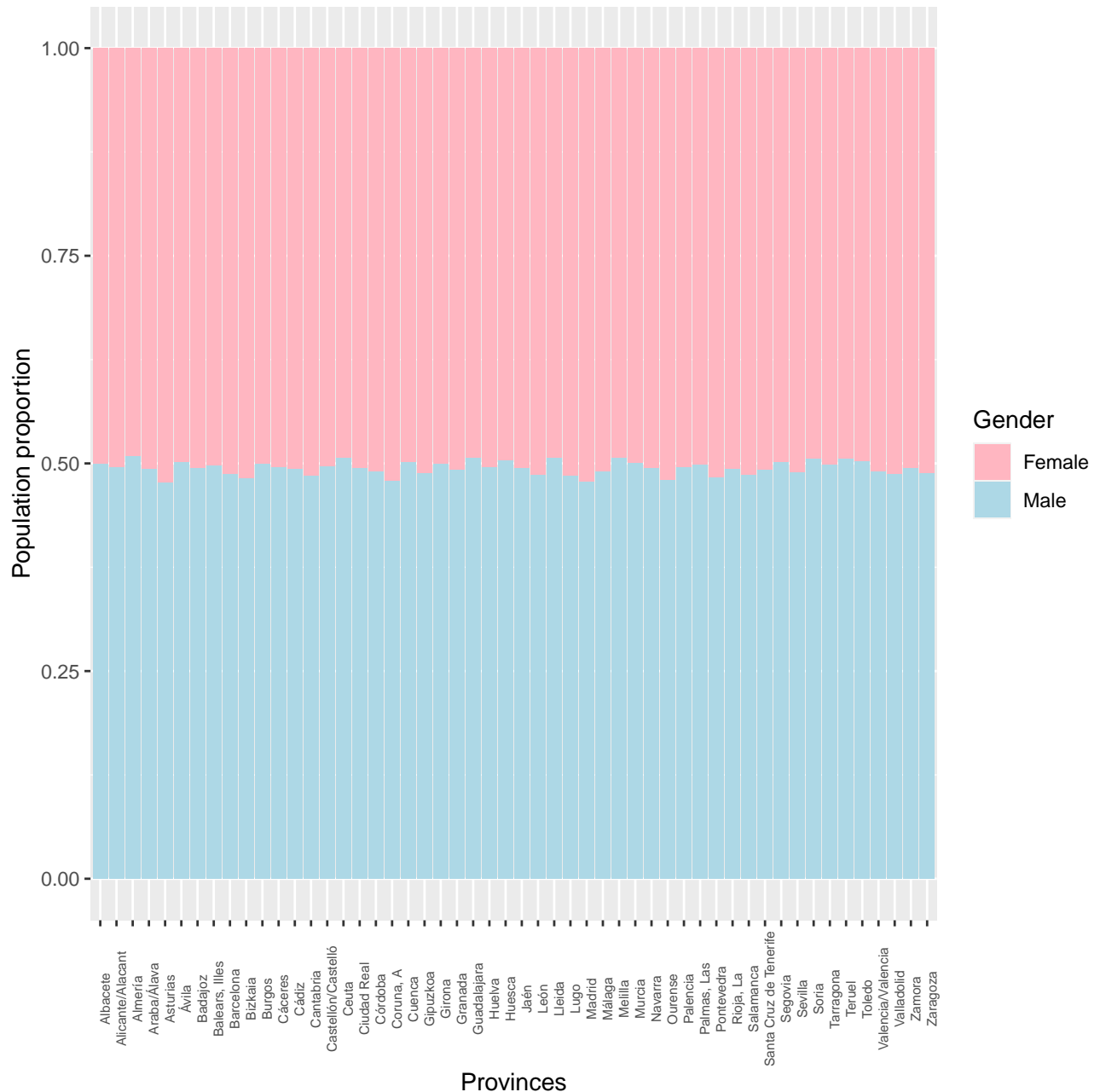
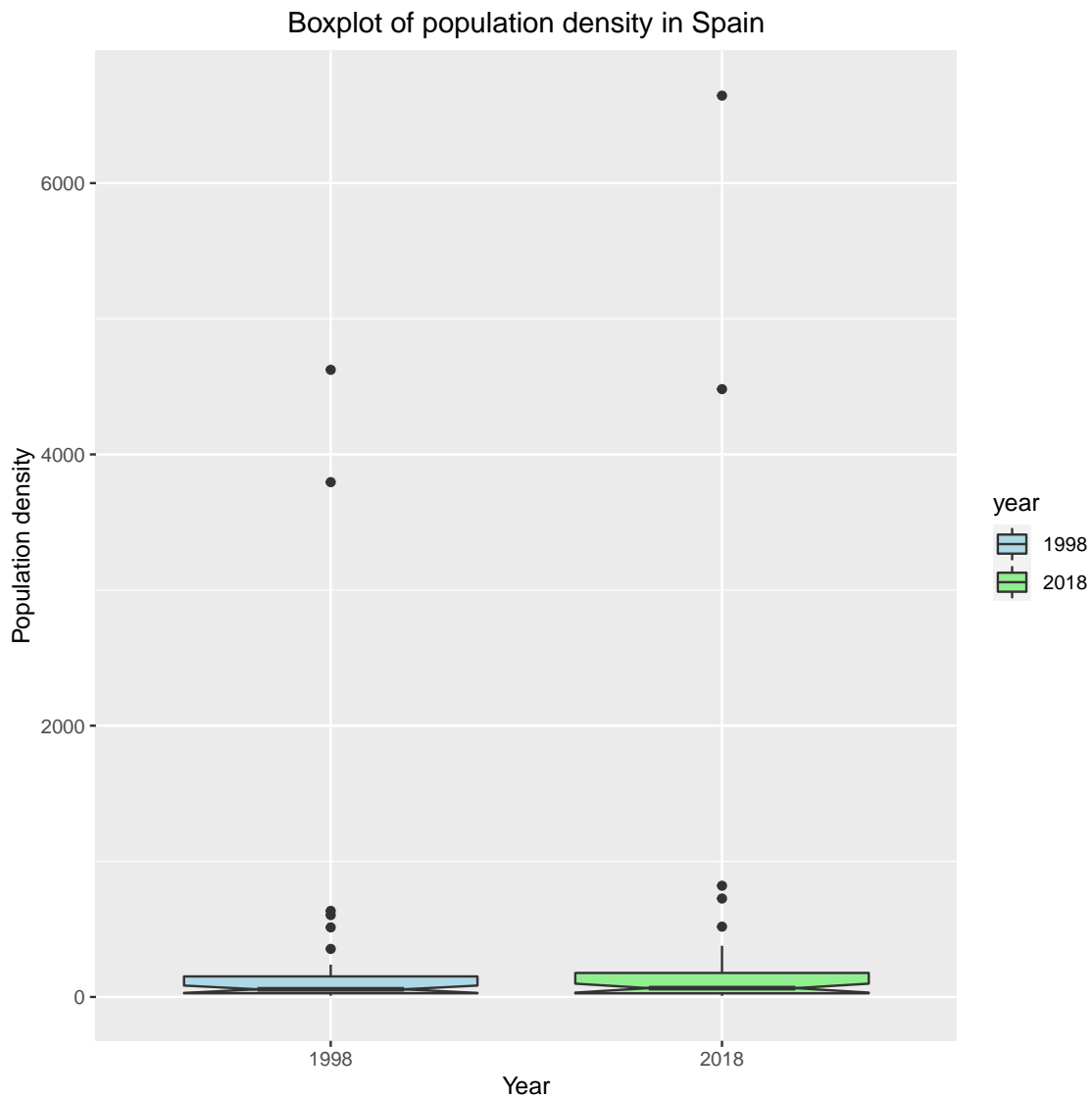


Table of characteristics of population density in Spain (1998 and 2018)

	Year 1998	Year 2018
Size	52.0000	52.0000
Mean	268.3568	339.0729
Minimum	8.8899	8.5994
Q1	26.5034	26.4518
Q2	55.7095	64.1000
Q3	154.2437	188.0495
Maximum	4623.6923	6644.9231
Skewness	4.5286	4.7737
Kurtosis	19.7477	22.5997
Standard Deviation	812.2020	1089.5388

Table 3: Characteristics of Population density of Spain



Histograms represent population density in Spain

Histograms show population density in Spain in 1998 and 2018. The x-axis shows population density while the y-axis shows frequency.

