Chapther 3 Exercises

Abel Isaías Gutiérrez-Cruz

13/7/2021

Exercise 3.1

A hiking enthusiast has a new app for his smartphone which summarizes his hikes by using a GPS device. Let us look at the distance hiked (in km) and maximum altitude (in m) for the last 10 hikes:

Distance	12.5	29.9	14.8	18.7	7.6	16.2	16.5	27.4	12.1	17.5
Altitude	342	1245	502	555	398	670	796	912	238	466

(a) Calculate the arithmetic mean and median for both distance and altitude.

```
distance <- c(12.5, 29.9, 14.8, 18.7, 7.6, 16.2, 16.5, 27.4, 12.1, 17.5)
distance <- distance[order(distance)]
altitude <- c(342, 1245, 502, 555, 398, 670, 796, 912, 238, 466)
altitude <- altitude[order(altitude)]
data_exercise3_1 <- data.frame(distance = distance, altitude = altitude)
```

Mean and median Distance

```
summarize(data_exercise3_1, mean_distance = mean(distance), median_distance = median(distance))
```

```
##   mean_distance median_distance
## 1         17.32           16.35
```

Mean and median Altitude

```
summarize(data_exercise3_1, mean_altitude = mean(altitude), median_altitude = median(altitude))
```

```
##   mean_altitude median_altitude
## 1         612.4           528.5
```

(b) Determine the first and third quartiles for both the distance and the altitude variables. Discuss the shape of the distribution given the results of (a) and (b).

```
#"Quartiles disntance"
quartile_distance <- quantile(data_exercise3_1$distance, probs = c(0.25, 0.75), type = 2)

#"Quartiles altitude"
quartile_altitude <- quantile(data_exercise3_1$altitude, probs = c(0.25, 0.75), type = 2)
```

```
summary(data_exercise3_1)
```

```
##     distance        altitude
##  Min.   : 7.60   Min.   : 238.0
##  1st Qu.:13.07   1st Qu.: 415.0
##  Median :16.35   Median : 528.5
##  Mean   :17.32   Mean   : 612.4
```

```
##  3rd Qu.:18.40   3rd Qu.: 764.5
##  Max.   :29.90   Max.   :1245.0
```

(c) Calculate the interquartile range, absolute median deviation, and variance for both variables. What is your conclusion about the variability of the data?

To calculate de variance we need to multiply the result by $(n-1)/n$, because R uses $1/(n-1)$ rather than $1/n$ when calculating the variance

```
amd <- function(x){1/length(x)*sum(abs(x-median(x)))}
# Interquartile distance
quartile_distance[[2]] - quartile_distance[[1]]
```

```
## [1] 6.2
```

```
# Absolute median deviation distance
amd(data_exercise3_1$distance)
```

```
## [1] 4.68
```

```
# Variance distance
var(data_exercise3_1$distance)*9/10
```

```
## [1] 41.5036
```

```
# Interquartile altitude
quartile_altitude[[2]] - quartile_altitude[[1]]
```

```
## [1] 398
```

```
# Absolute median deviation altitude
amd(data_exercise3_1$altitude)
```

```
## [1] 223.2
```

```
# variance altitude
var(data_exercise3_1$altitude)*9/10
```

```
## [1] 82314.44
```

(d) One metre corresponds to approximately 3.28 ft. What is the average altitude when measured in feet rather than in metres?

$\bar{y} = a + b\bar{x}$

$\tilde{s}_y^2 = b^2\tilde{s}_x^2$

```
# Mean
3.28*mean(data_exercise3_1$altitude)
```

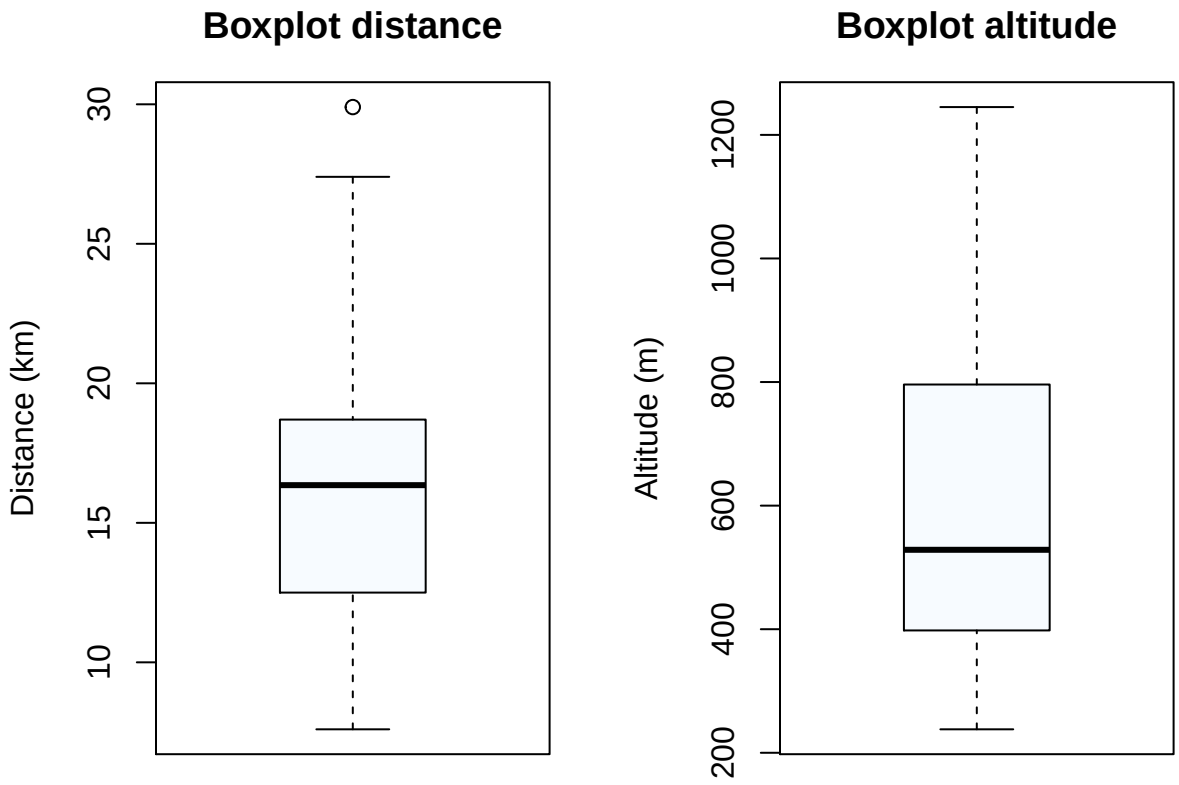
```
## [1] 2008.672
```

```
# Variance
(3.28^2)*(var(data_exercise3_1$altitude)*9/10)
```

```
## [1] 885571.7
```

(e) Draw and interpret the box plot for both distance and altitude.

```
par(mfrow = c(1, 2), mar = c(2, 5, 3, 1))
boxplot(data_exercise3_1$distance, main = "Boxplot distance", col = blues9, ylab = "Distance (km)")
boxplot(data_exercise3_1$altitude, main = "Boxplot altitude", col = blues9, ylab = "Altitude (m)")
```



(f) Assume distance is measured as only short (5–15 km), moderate (15–20 km), and long (20–30 km). Summarize the grouped data in a frequency table. Calculate the weighted arithmetic mean under the assumption that the raw data is not known. Determine the weighted median under the assumption that the values within each class are equally distributed.

```
groups <- c(5, 15, 20, 30)
labels_distances <- c("1", "2", "3")
grouped_distance <- cut(distance, breaks = groups, labels = labels_distances, right = FALSE)
data_exercise3_1$groups <- grouped_distance

relative_frecuency <- as.data.frame(table(grouped_distance)/nrow(data_exercise3_1))
# get the weithted arithmetic mean
middle_values <- groups[1:3] + (diff(groups)/2)
weighted.mean(middle_values, relative_frecuency$Freq)
```

```
## [1] 16
```

Estimate the weighted median:

```
groups[2] + ((0.5 - relative_frecuency$Freq[1])/relative_frecuency$Freq[1])*(diff(groups)[2])
```

```
## [1] 16.25
```

Exercise 3.2

A gambler notes down his wins and losses (in euros) from playing 10 games of roulette in a casino.

Round	Won/Lost
1	200
2	600
3	200
4	200
5	200
6	100
7	100
8	400
9	0

10

(a) Assume $\bar{x} =$ 90 euros and s = 294.7881 euros. What is the result of round 10?

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

$$-90 = \frac{1}{10}(-400 + X)$$

$$X = (-90 \times 10) + 400$$

```
won_lost = c(200, 600, -200, -200, -200, -100, -100, -400, 0)
data_exercise3_2 <- data.frame(round = 1:10, won_lost = c(won_lost, ((-90*10) - sum(won_lost))))
kable(data_exercise3_2)
```

round	won_lost
1	200
2	600
3	-200
4	-200
5	-200
6	-100
7	-100
8	-400
9	0
10	-500

(b) Determine the mode and the interquartile range.

```
Modes <- function(x) {
  ux <- unique(x)
  tab <- tabulate(match(x, ux))
  ux[tab == max(tab)]
}

Modes(data_exercise3_2$won_lost)
```

```
## [1] -200
```

(c) A different gambler plays 33 rounds of roulette. His results are $\bar{x} = 12$ euros and s = 1000 euros. Is it meaningful to compare the variability of results of the two players by using the coefficient of variation? If yes, determine the coefficients of variation; if no, why is a comparison not possible?

It isn't possible to estimate the coefficient of variation because some of the values are negative

Exercise 3.3

A fashion boutique has summarized its daily sales of designer socks in different groups: men's socks, women's socks, and children's socks. Unfortunately, the data for men's socks was lost. Determine the missing values.

.	n	Arithmetic mean in euros	standard deviation
Women's wear	45	16	$\sqrt{6}$
Men's wear	?	?	?
Children's wear	20	7.5	$\sqrt{3}$
total	100	15	$\sqrt{19.55}$

Men's wear n: $n_T = n_m + n_w + n_c$

$n_m = n_T - n_w - n_c$

```
100 - 45 - 20
```

```
## [1] 35
```

Men's wear $\bar{x}$:

$\bar{x}_T = \frac{1}{n}\sum_{j=1}^{k} n_j m_j$

$\bar{x}_T = \frac{1}{n_T}(n_w\bar{x}_w + n_c\bar{x}_c + n_m\bar{x}_m)$

$\bar{x}_m = \frac{1}{n_m}(n_T\bar{x}_T - n_c\bar{x}_c - n_w\bar{x}_w)$

```
1/35*(100*15 - 20*7.5 - 46*16)
```

```
## [1] 17.54286
```

Men's wear $\tilde{s}^2$

$\tilde{s}^2 = \frac{1}{n}\sum_{i=1}^{k} n_j(\bar{x}_j - \bar{x})^2 + \frac{1}{n}\sum_{i=1}^{k} n_j\tilde{s}_j^2$

$\tilde{s}_T^2 = \frac{1}{n_T}\left(n_w(\bar{x}_w - \bar{x}_T)^2 + n_c(\bar{x}_c - \bar{x}_T)^2 + n_m(\bar{x}_m - \bar{x}_T)^2\right) + \frac{1}{n_T}\left(n_w\tilde{s}_w^2 + n_c\tilde{s}_c^2 + n_m\tilde{s}_m^2\right)$

$\frac{1}{n_m}\left(\tilde{s}_T^2 n_T - n_w\tilde{s}_w^2 - n_c\tilde{s}_c^2 - n_w(\bar{x}_w - \bar{x}_T)^2 - n_c(\bar{x}_c - \bar{x}_T)^2 - n_m(\bar{x}_m - \bar{x}_T)^2\right) = \tilde{s}_m^2$

```
1/35*(19.55*100 -(45*6) -(20*3) -(45*(16 - 15)**2) -(20*(7.5-15)**2) -(35*(18 - 15)**2))
```

```
## [1] 4
```

Exercise 3.4

The number of members of a millionaires' club were as follows:

Year	Members
2011	23
2012	24
2013	27
2014	25
2015	30
2016	28

```
data_exercise3_4 <- data.frame(year = c(2011:2016), members = c(23, 24, 27, 25, 30, 28))
```

(a) What is the average growth rate of the membership?

```
data_exercise3_4 <- data_exercise3_4 %>%
    mutate(growth_factor = c(0, sapply(2:nrow(data_exercise3_4), function(i)
        members[i]/members[i-1])))

(prod(data_exercise3_4$growth_factor[2:nrow(data_exercise3_4)]))**(1/5)
```

```
## [1] 1.040126
```

(b) Based on the results of (a), how many members would one expect in 2018?

$B_{2018} = B_{2017} \times \bar{x}_G$

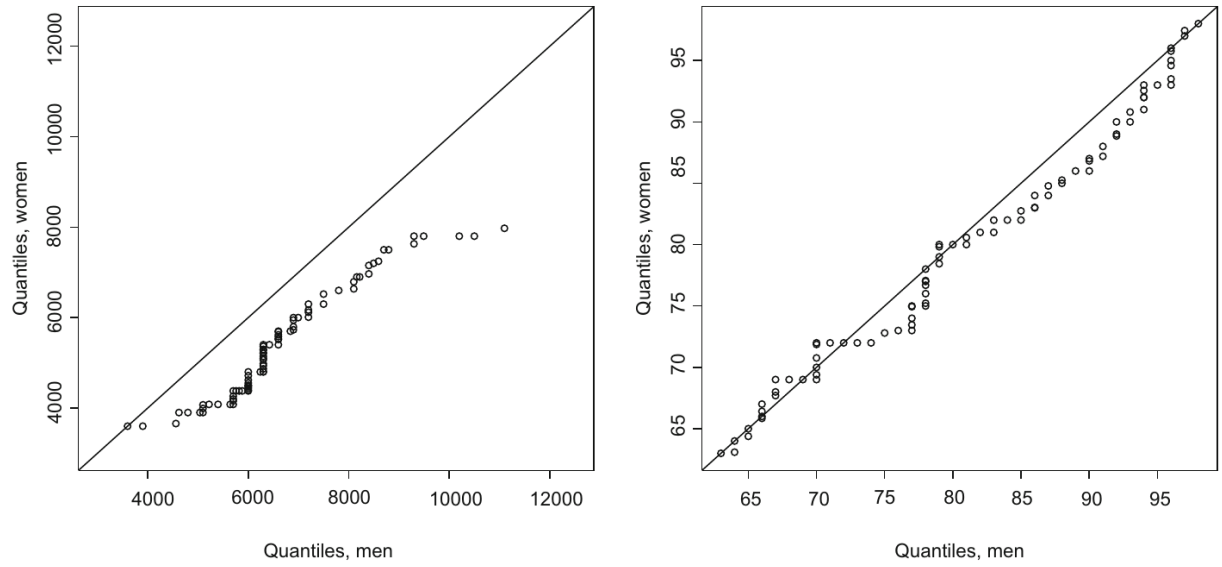
$B_{2017} = B_{2016} \times \bar{x}_G$

$B_{2018} = (B_{2016} \times \bar{x}_G) \times \bar{x}_G$

$B_{2017} = B_{2016} \times \bar{x}_G^2$

```
last(data_exercise3_4$members) * (1.04**2)
```

```
## [1] 30.2848
```



(a) for the salary (b) for length of service

Fig. 3.8 QQ-plots

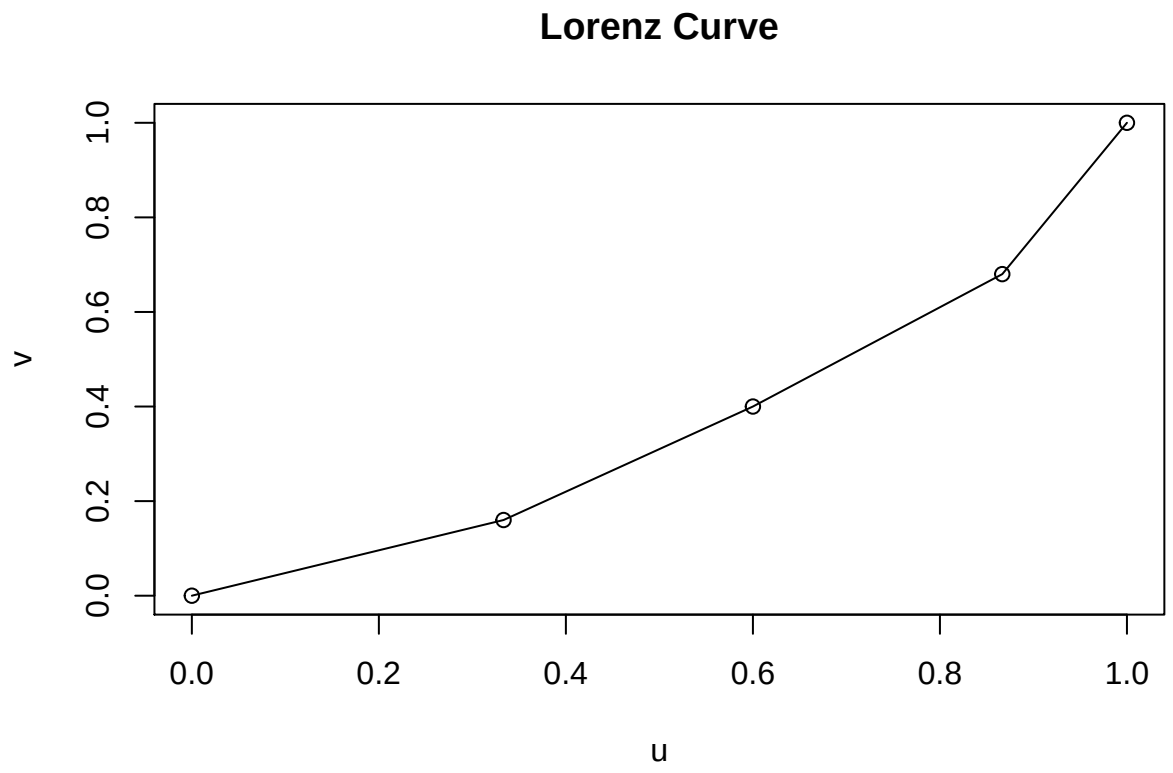
(c) The president of the club is interested in the number of members in 2025, the year when his presidency ends. Would it make sense to predict the number of members for 2025?

It could be used, but is unrealistic that the behavior of the data will be the same during all this time

In 2015, the members invested 250 euros million on the stock market. 10 members contributed 16% of the investment sum, 8 members contributed 60 euros million, 8 members contributed 70 euros million, and another 4 members contributed the remaining amount.

(d) Draw the Lorenz curve for this data.

```r
data2_exercise3_4 <- data.frame(members = c(0, 10, 8, 8, 4), invested = c(0, .16*250, 60, 70, 80)) %>%
  mutate(members_freq = members / sum(members)) %>% mutate(u = cumsum(members_freq)) %>%
  mutate(invested_freq = invested/sum(invested)) %>% mutate(v = cumsum(invested_freq))

plot(data2_exercise3_4$u, data2_exercise3_4$v, xlab = "u", ylab = "v", main = "Lorenz Curve")
lines(data2_exercise3_4$u, data2_exercise3_4$v)
```



(e) Calculate and interpret the standardized Gini coefficient.

```r
sum_v <- sum(sapply(2:nrow(data2_exercise3_4), function(i) data2_exercise3_4$v[i] + data2_exercise3_4$v
               data2_exercise3_4$members[2:nrow(data2_exercise3_4)])

# Gini coefficient
1 - (1/sum(data2_exercise3_4$members)*sum_v)
```

```
## [1] 0.2853333
```

Exercise 3.5

Consider the monthly salaries Y (in Swiss francs) of a well-reputed software company, as well as the length of service (in months, X), and gender (Z). Figure 3.8 shows the QQ-plots for both Y and X given Z . Interpret both graphs.

In the graph for the salary, we would seem that men have a higher salary than women. For this reason, the dots tends to the bottom. Furthermore, in the graph for the length of service, the data would seem more equitation for both, men and women.

Exercise 3.6

There is no built-in function in R to calculate the mode of a variable. Program such a function yourself. Hint: type ?table and ?names to recall the functionality of these functions. Combine them in an intelligent way.

```
Modes <- function(x) {
  ux <- unique(x)
  tab <- tabulate(match(x, ux))
  ux[tab == max(tab)]
}
```

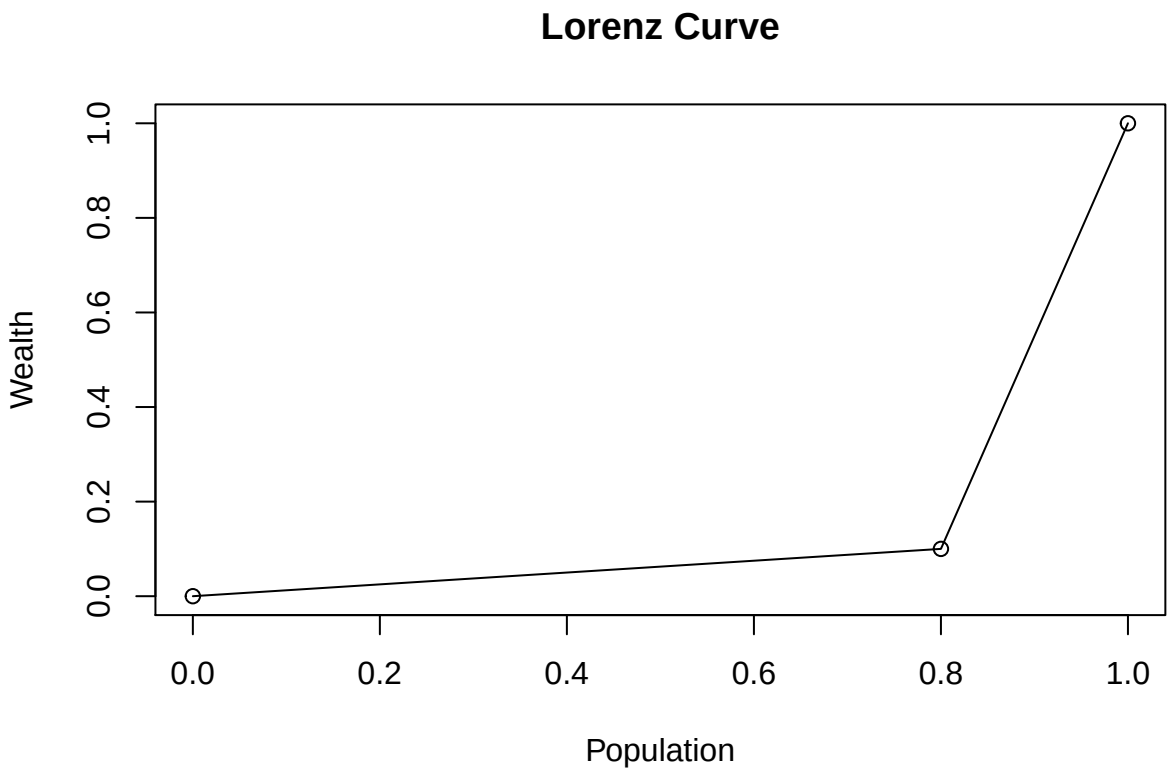
Exercise 3.7

Consider a country in which 90 % of the wealth is owned by 20 % of the population, the so-called upper class. For simplicity, let us assume that the wealth is distributed equally within this class.

(a) Draw the Lorenz curve for this country.

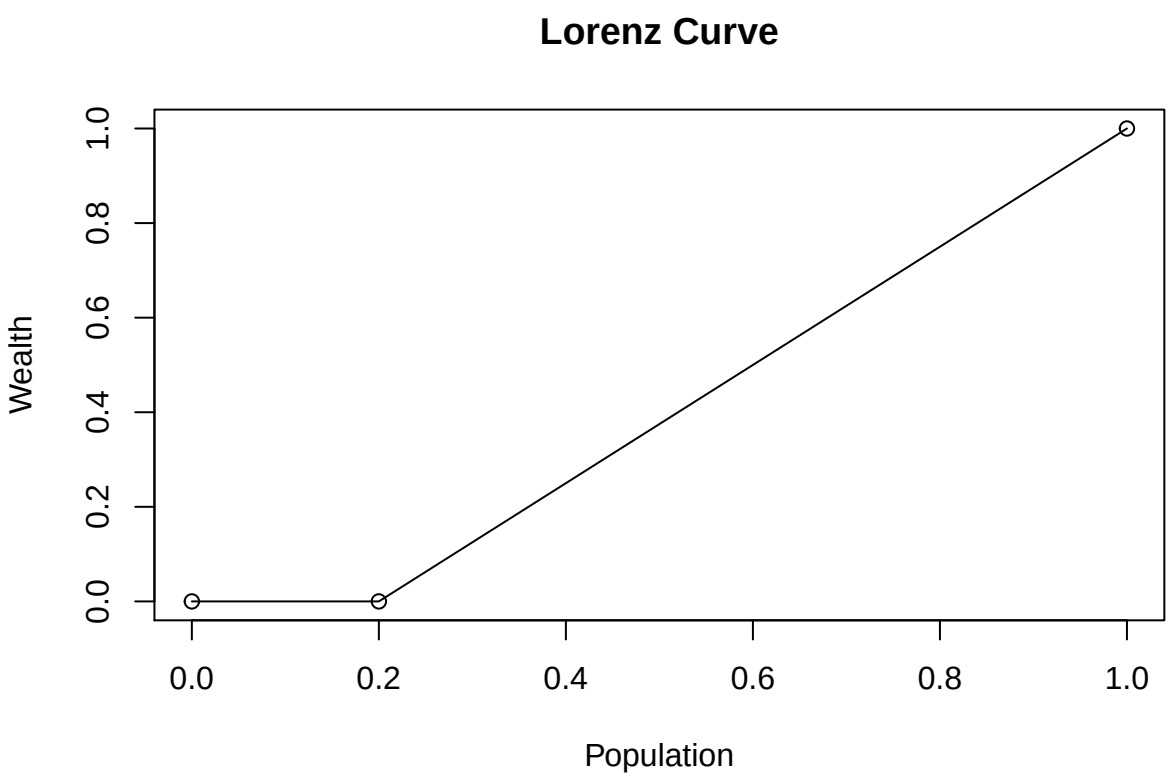
In this case we only have two dots of the graph

```
data_exercise3_7 <- data.frame(population = c(0, 0.8, 1), wealth = c(0, 0.1, 1))

plot(data_exercise3_7$population, data_exercise3_7$wealth, xlab = "Population", ylab = "Wealth", main =
lines(data_exercise3_7$population, data_exercise3_7$wealth)
```



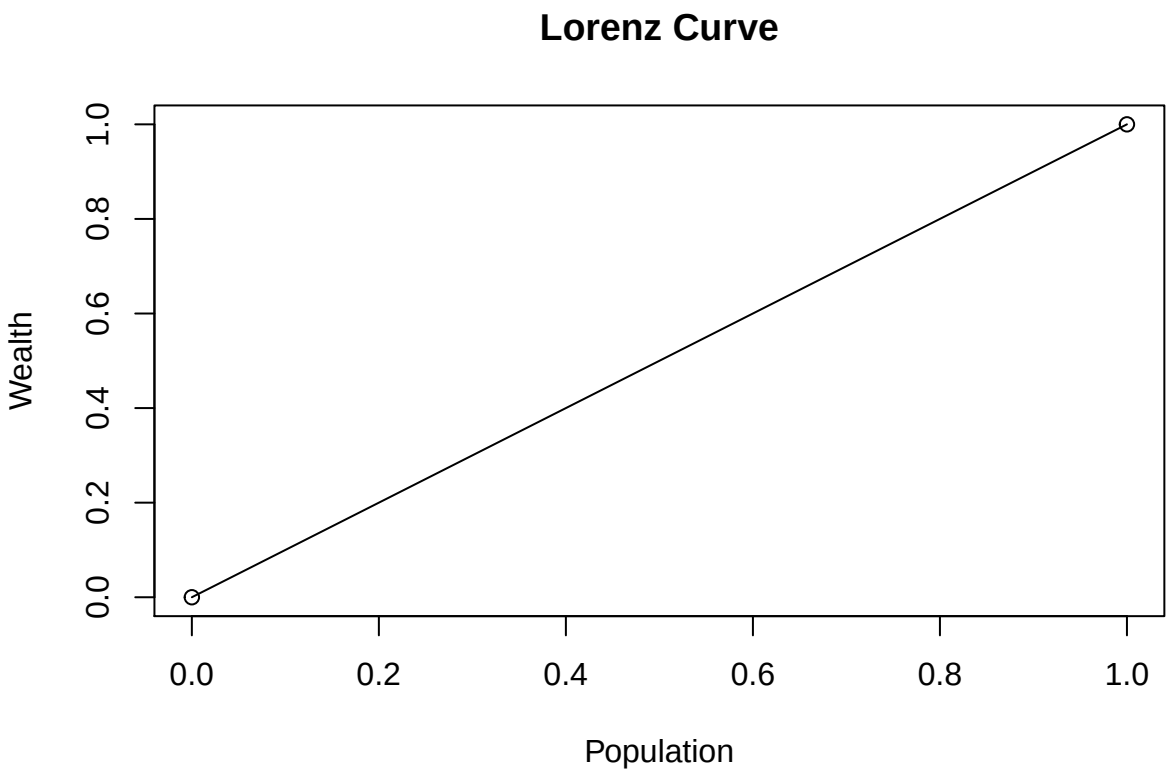
(b) Now assume a revolution takes place in the country and all members of the upper class have to give away their wealth which is then distributed equally across the remaining population. Draw the Lorenz curve for this scenario.

```r
data_exercise3_7 <- data.frame(population = c(0, 0.2, 1), wealth = c(0, 0, 1))

plot(data_exercise3_7$population, data_exercise3_7$wealth, xlab = "Population", ylab = "Wealth", main =
lines(data_exercise3_7$population, data_exercise3_7$wealth)
```



(c) What would the curve from (b) look like if the entire upper class left the country?

```r
data_exercise3_7 <- data.frame(population = c(0, 1), wealth = c(0, 1))

plot(data_exercise3_7$population, data_exercise3_7$wealth, xlab = "Population", ylab = "Wealth", main =
lines(data_exercise3_7$population, data_exercise3_7$wealth)
```



Exercise 3.8

A bus route in the mountainous regions of Romania has a length of 418 km. The manager of the bus company serving the route wants his buses to finish a trip within 8 h. The bus travels the first 180 km with an average speed of 48 km/h, the next 117 km with an average speed of 37 km/h, and the last section with an average speed of 52 km/h. (a) What is the average speed with which the bus travels?

```
data_exercise3_8 <- data.frame(distance = c(180, 117, 418 - 180 - 117), av_speed = c(48, 37, 52))

av_speed <- mean(data_exercise3_8$av_speed)
av_speed
```

```
## [1] 45.66667
```

(b) Will the bus finish the trip in time?

```
418 * 1/av_speed <= 8
```

```
## [1] FALSE
```

Exercise 3.9

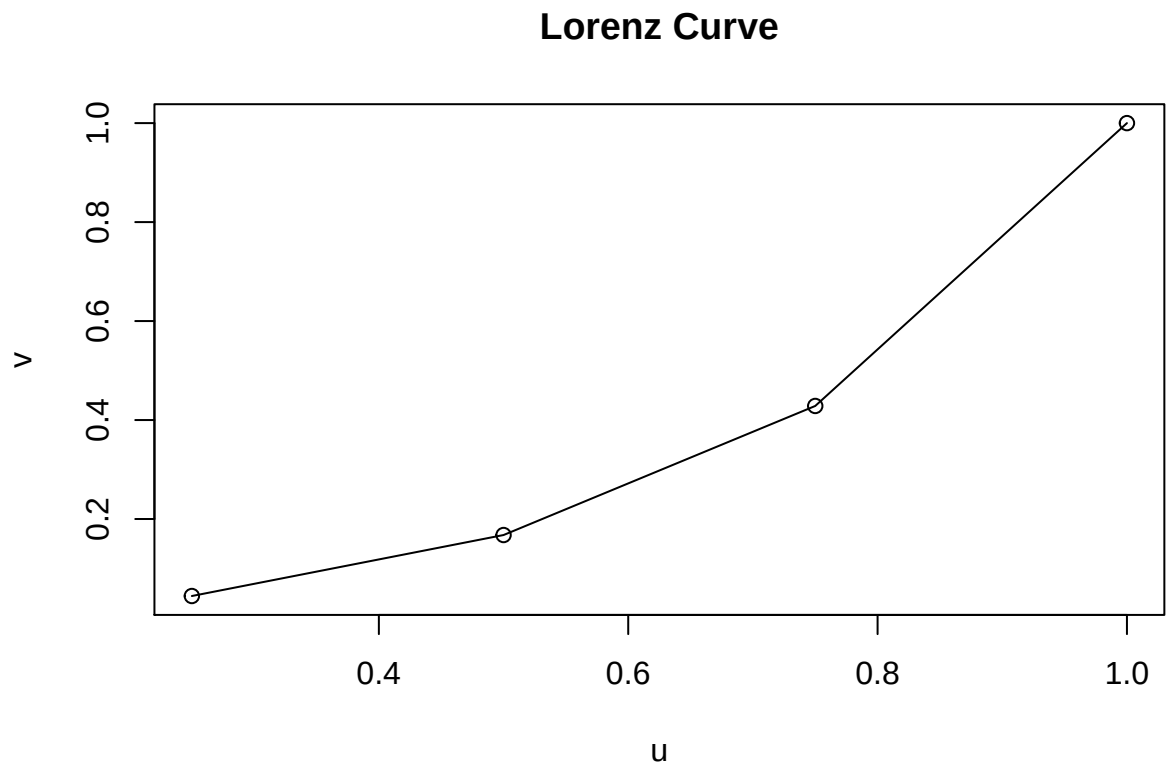
Four friends have a start-up company which sells vegan ice cream. Their initial financial contributions are as follows:

Person	Contribution (in euros)
1	800
2	10300

Person	Contribution (in euros)
3	4700
4	2220

(a) Calculate and draw the Lorenz curve.

```
data_exercise3_9 <- data.frame(person = c(1:4), contribution = c(800, 10300, 4700, 2220)) %>% arrange(c
  mutate(person_freq = c(0.25, 0.25, 0.25, 0.25)) %>% mutate(u = cumsum(person_freq)) %>%
  mutate(contribution_freq = contribution/sum(contribution)) %>% mutate(v = cumsum(contribution_freq))

plot(data_exercise3_9$u, data_exercise3_9$v, xlab = "u", ylab = "v", main = "Lorenz Curve")
lines(data_exercise3_9$u, data_exercise3_9$v)
```



(b) Determine and interpret the standardized Gini coefficient.

```
sum_v <- sum(c(data_exercise3_9$v[1], sapply(2:nrow(data_exercise3_9), function(i) data_exercise3_9$v[i
                                                data_exercise3_9$v[i-1])))

# Gini coefficient
gini_coeff <- 1 - (1/nrow(data_exercise3_9)*sum_v)
```

G^+:

```
(nrow(data_exercise3_9)/(nrow(data_exercise3_9) - 1))*gini_coeff
```

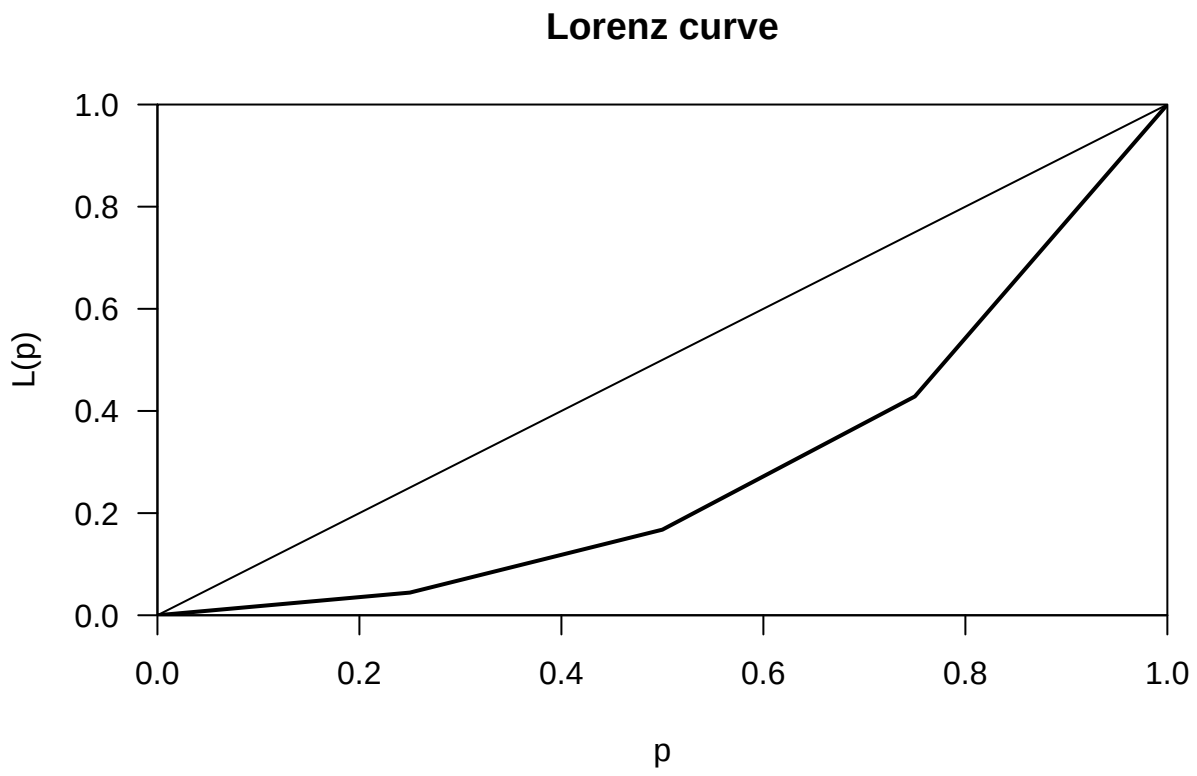
```
## [1] 0.573067
```

(c) Does G^+ change if each of the friends contributes only half the amount of money? If yes, how much? If

no, why not? The Gini coefficient remains the same

(d) Use R to draw the above Lorenz curve and to calculate the Gini coefficient.

```
library(ineq)
plot(Lc(data_exercise3_9$contribution))
```




```
ineq(data_exercise3_9$contribution)
```

```
## [1] 0.4298002
```

Exercise 3.10

Recall the pizza delivery data which is described in Appendix A.4. Use R to read in and analyse the data.

```
# load the data
data_pizza <- read.csv("../../Data/pizza_delivery.csv")
```

(a) Calculate the mean, median, minimum, maximum, first quartile, and third quartile for all quantitative variables.

```
summary(data_pizza)
```

```
##      day                date                time         operator
##  Length:1266        Length:1266        Min.   :12.27   Length:1266
##  Class :character   Class :character   1st Qu.:30.06   Class :character
##  Mode  :character   Mode  :character   Median :34.38   Mode  :character
##                                        Mean   :34.23
##                                        3rd Qu.:38.58
```

```
##                                       Max.   :53.10
##     branch             driver           temperature         bill
##  Length:1266        Length:1266        Min.   :41.76   Min.   : 9.10
##  Class :character   Class :character   1st Qu.:58.24   1st Qu.:35.50
##  Mode  :character   Mode  :character   Median :62.93   Median :42.90
##                                        Mean   :62.86   Mean   :42.76
##                                        3rd Qu.:67.23   3rd Qu.:50.50
##                                        Max.   :87.58   Max.   :75.00
##      pizzas         free_wine         got_wine      discount_customer
##  Min.   : 1.000   Min.   :0.0000   Min.   :0.0000   Min.   :0.000
##  1st Qu.: 2.000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.000
##  Median : 3.000   Median :0.0000   Median :0.0000   Median :0.000
##  Mean   : 3.013   Mean   :0.1809   Mean   :0.1485   Mean   :0.218
##  3rd Qu.: 4.000   3rd Qu.:0.0000   3rd Qu.:0.0000   3rd Qu.:0.000
##  Max.   :11.000   Max.   :1.0000   Max.   :1.0000   Max.   :1.000
```

(b) Determine and interpret the 99 % quantile for delivery time and temperature.

```
#quantile(data_pizza$time, probs = c(.99))

summarize(data_pizza, quantile_time = quantile(time, probs = c(0.99)),
                        quantie_temperature = quantile(temperature, probs = c(0.99)))
```

```
##   quantile_time quantie_temperature
## 1      48.61677               79.87
```

The 99% of the time data is less than or equal to 48.6 min and the 99% of the temperature data is less than or equal to 79.87 °C

(c) Write a function which calculates the absolute mean deviation. Use the function to calculate the absolute mean deviation of temperature.

```
amd <- function(x){1/length(x)*sum(abs(x-mean(x)))}
amd(data_pizza$temperature)
```

```
## [1] 5.473862
```

(d) Scale the delivery time and calculate the mean and variance for this variable.

```
scaled_time <- scale(data_pizza$time)
mean(scaled_time)
```

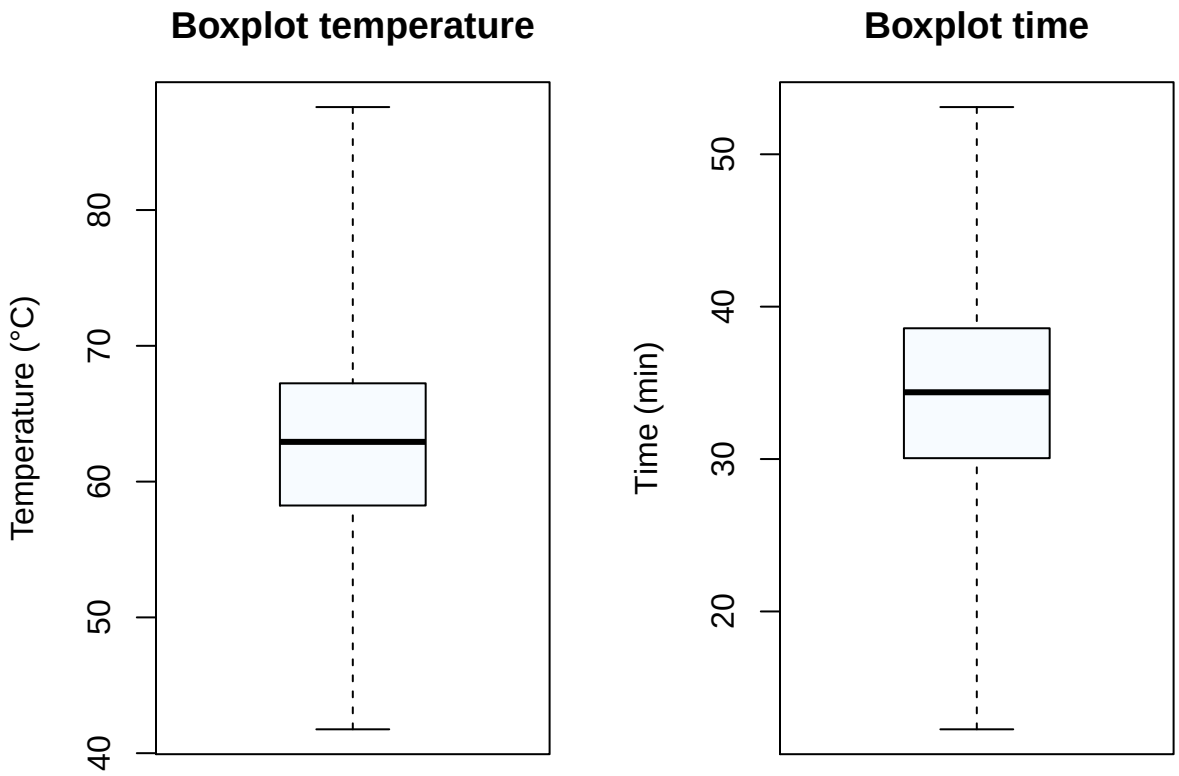
```
## [1] 5.222066e-16
```

```
var(scaled_time)
```

```
##      [,1]
## [1,]    1
```

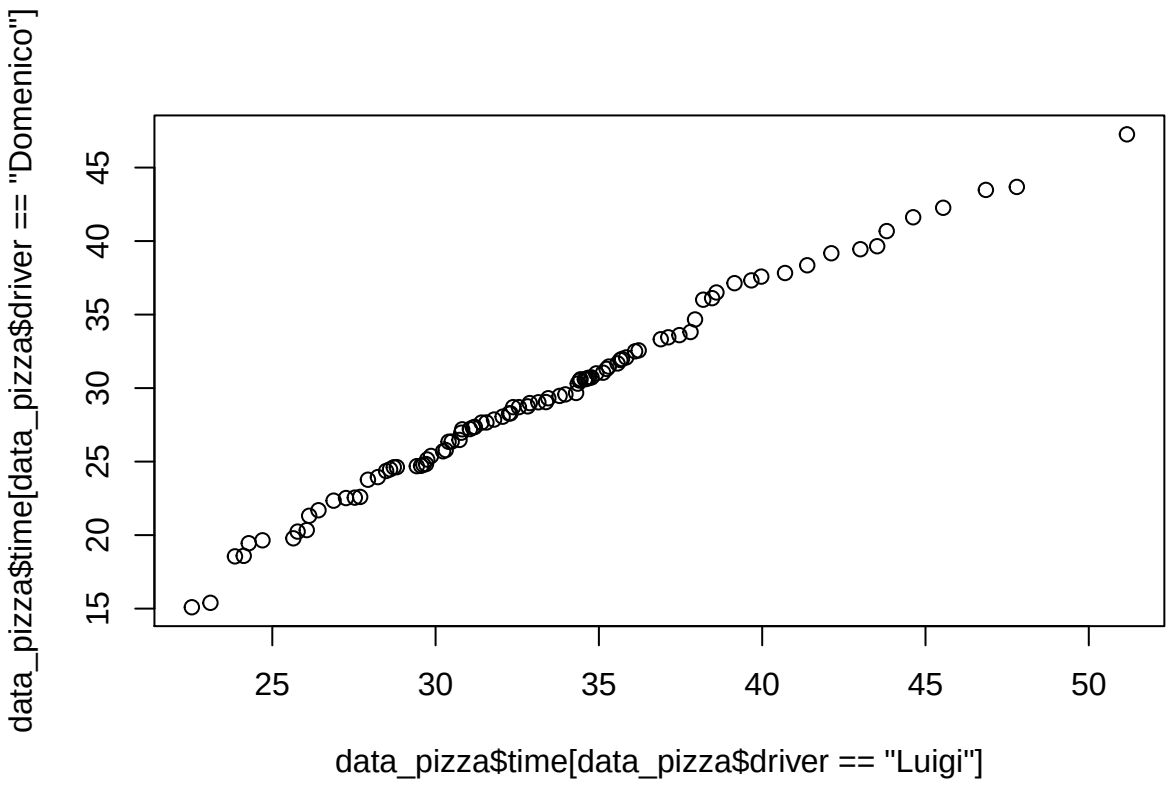
(e) Draw a box plot for delivery time and temperature. The box plots should not highlight extreme values.

```
par(mfrow = c(1, 2), mar = c(2, 5, 3, 1))
boxplot(data_pizza$temperature, main = "Boxplot temperature", col = blues9, ylab = "Temperature (°C)", r
boxplot(data_pizza$time, main = "Boxplot time", col = blues9, ylab = "Time (min)", range = 0)
```

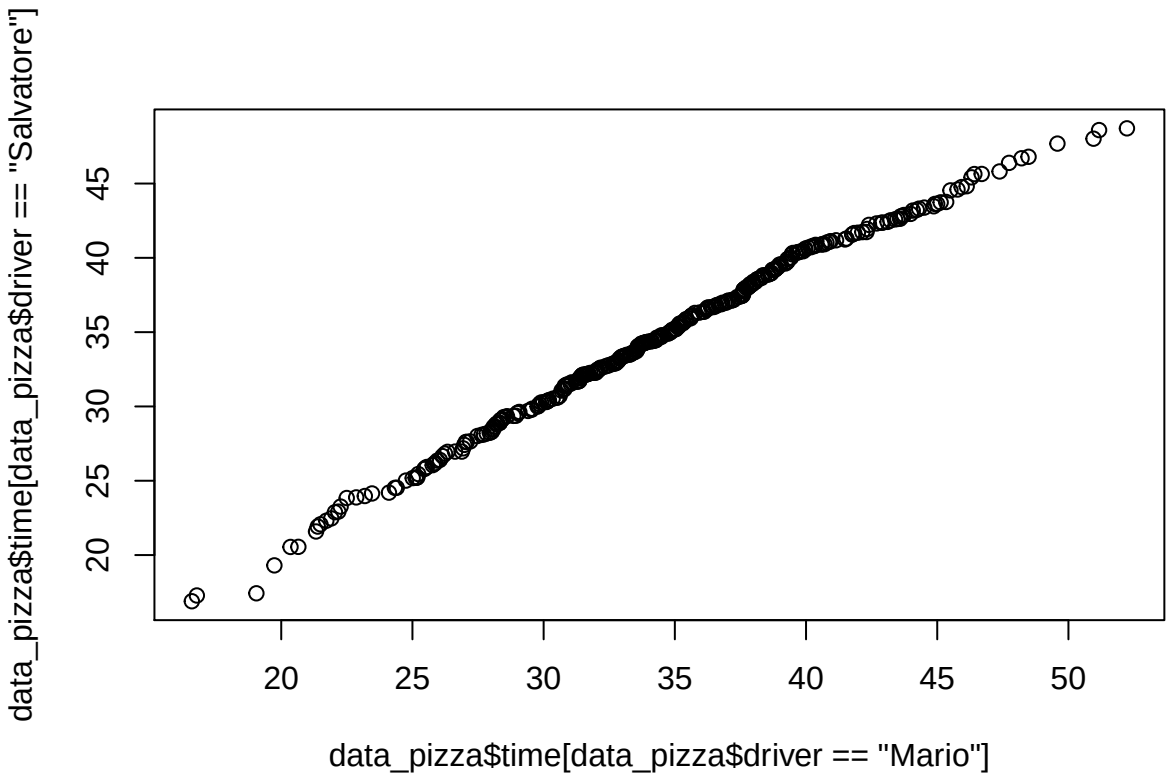


(f) Reproduce the QQ-plots shown in Example 3.1.6.

```
qqplot(data_pizza$time[data_pizza$driver == 'Luigi'], data_pizza$time[data_pizza$driver == 'Domenico'])
```




```
qqplot(data_pizza$time[data_pizza$driver == 'Mario'], data_pizza$time[data_pizza$driver == 'Salvatore'])
```


data_pizza$time[data_pizza$driver == "Salvatore"]
20
25
30
35
40
45
data_pizza$time[data_pizza$driver == "Mario"]
20
25
30
35
40
45
50