

Exercise 1

- logical, double, integer and character
- While matrices are similar to data frames, matrices are more limited because they can only contain data of the same type.
- The function 'as.integer()' converts the logical argument of True or False to integer values of 1 and 0.
- It depends on the input of the columns. Typically the 'as.matrix()' function will retrieve information from the data frame and provide us with characters.

Exercise 2

```
a <- c(1, 2, 3, 4, 5)
a
```

```
[1] 1 2 3 4 5
```

```
typeof(a)
```

```
[1] "double"
```

a: creates a numeric vector and it's type is "double".

```
b <- 1:5
b
```

```
[1] 1 2 3 4 5
```

```
typeof(b)
```

```
[1] "integer"
```

b: creates an integer sequence so it's type is "integer".

```
c <- c(sqrt(2), 4.7e4, 1/0)
c
```

```
[1] 1.414214 47000.000000 Inf
```

```
typeof(c)
```

```
[1] "double"
```

c: Combines numeric values and it's type is "double".

```
d <- c(T, T, T, T)
d
```

```
[1] TRUE TRUE TRUE TRUE
```

```
typeof(d)
```

```
[1] "logical"
```

d: Creates "logical".

```
e <- c("1", 2, 3)
e
```

```
[1] "1" "2" "3"
```

```
typeof(e)
```

```
[1] "character"
```

e: Creates "character"

```
f <- c(7L, NA, 5L, 3L)
f
```

```
[1] 7 NA NA 5 3
```

```
typeof(f)
```

```
[1] "integer"
```

f: Creates "integer".

```
g <- c(7L, "NA", "NA", 5L, 3L)
g
```

```
[1] "7" "NA" "NA" "5" "3"
```

```
typeof(g)
```

```
[1] "character"
```

g: creates "character".

```
h <- c()
h
```

```
NULL
```

```
typeof(h)
```

```
[1] "NULL"
```

h: Creates a "NULL" vector

Exercise 3

```
head(airquality)
```

```
   Ozone Solar.R Wind Temp Month Day
1    41     190   7.4   67    5    1
2    36     118   8.0   72    5    2
3    12     149  12.6   74    5    3
4    18     313  11.5   62    5    4
5     NA      NA  14.3   56    5    5
6    28      NA  14.9   66    5    6
```

```
Ozone1 <- airquality$Ozone
is.na(Ozone1)
```

```
[1] FALSE FALSE FALSE TRUE FALSE FALSE FALSE TRUE FALSE FALSE
[13] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[25] TRUE TRUE TRUE FALSE FALSE FALSE TRUE TRUE TRUE TRUE TRUE
[37] TRUE FALSE TRUE FALSE FALSE TRUE TRUE FALSE TRUE TRUE FALSE
[49] FALSE FALSE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
[61] TRUE FALSE FALSE TRUE TRUE FALSE FALSE FALSE FALSE FALSE TRUE
[73] FALSE FALSE TRUE FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE
[85] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[97] FALSE FALSE FALSE FALSE TRUE TRUE FALSE FALSE FALSE TRUE FALSE
[109] FALSE FALSE FALSE FALSE TRUE TRUE FALSE FALSE TRUE FALSE
[121] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[133] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[145] FALSE FALSE FALSE FALSE TRUE FALSE FALSE FALSE
```

```
Ozone2 <- na.omit(Ozone1)
na_removed <- sum(is.na(Ozone1))
Ozone2
```

```
[1] 41 36 12 18 28 23 19 8 7 16 11 14 18 14 34 6 30 11
[19] 1 11 4 32 23 45 115 37 29 71 39 23 21 37 20 12 13 135
[37] 49 32 64 40 77 97 85 10 27 7 48 35 61 79 63 16 80
[55] 108 20 52 82 50 64 59 39 9 16 78 35 66 122 89 110 44 28
[73] 65 22 59 23 31 44 21 9 45 168 73 76 118 84 85 96 78 73
[91] 91 47 32 20 23 21 24 44 21 28 9 13 46 18 13 24 16 13
[109] 23 36 7 14 30 14 18 20
attr(,"na.action")
[1] 5 10 25 26 27 32 33 34 35 36 37 39 42 43 45 46 52 53 54
[20] 55 56 57 58 59 60 61 65 72 75 83 84 102 103 107 115 119 150
attr(,"class")
[1] "omit"
```

```
na_removed
```

```
[1] 37
```

b. There were 37 NA's removed

```
summary(Ozone2)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
1.00   18.00   31.50   42.13   63.25  168.00
```

```
sd(Ozone2)
```

```
[1] 32.98788
```

c. The min is 1.00, the median is 31.50, the mean is 42.13, the max is 168.00 and the standard deviation is 32.98788.

```
summary(airquality$Ozone)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's
1.00   18.00   31.50   42.13   63.25  168.00    37
```

```
sd(airquality$Ozone)
```

```
[1] NA
```

```
sd(airquality$Ozone, na.rm = TRUE)
```

```
[1] 32.98788
```

d. Summary: The summary command handles NA's by excluding them in the calculation of the summary statistics and includes a count of how many 'NA' values are present. The summary statistics are calculated only using non-missing values.

sd(airquality\$Ozone): The sd command handles 'NA' by returning NA if there are any present in the Ozone column.

sd(airquality\$Ozone, na.rm = TRUE):The 'na.rm' command instructs the sd function to remove the NA values before calculating the standard deviation.

```
airquality1 <- na.omit(airquality)
dim(airquality1)
```

```
[1] 42 0
```

e. There are 42 rows of the airquality data frame that have one or more missing values.

Exercise 4

a.

```
p <- seq(from = 0, to = 1, by = 0.2)
p
```

```
[1] 0.0 0.2 0.4 0.6 0.8 1.0
```

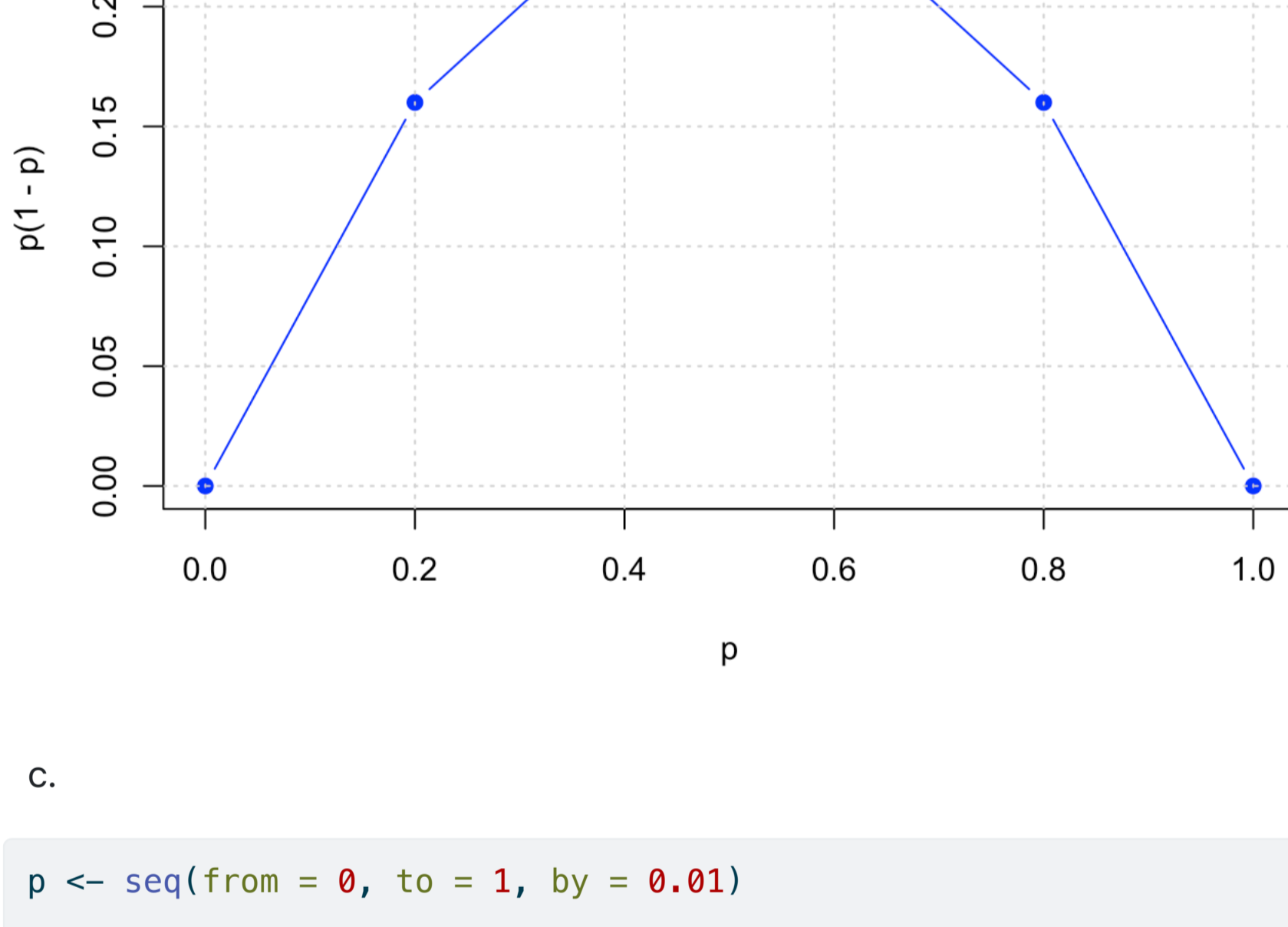
b.

```
f_p <- p * (1 - p)

plot(p, f_p, type = "b", col = "blue", pch = 19,
     xlab = "p", ylab = "p(1 - p)",
     main = "Plot of f(p) = p(1 - p)")

grid()
```

Plot of $f(p) = p(1 - p)$



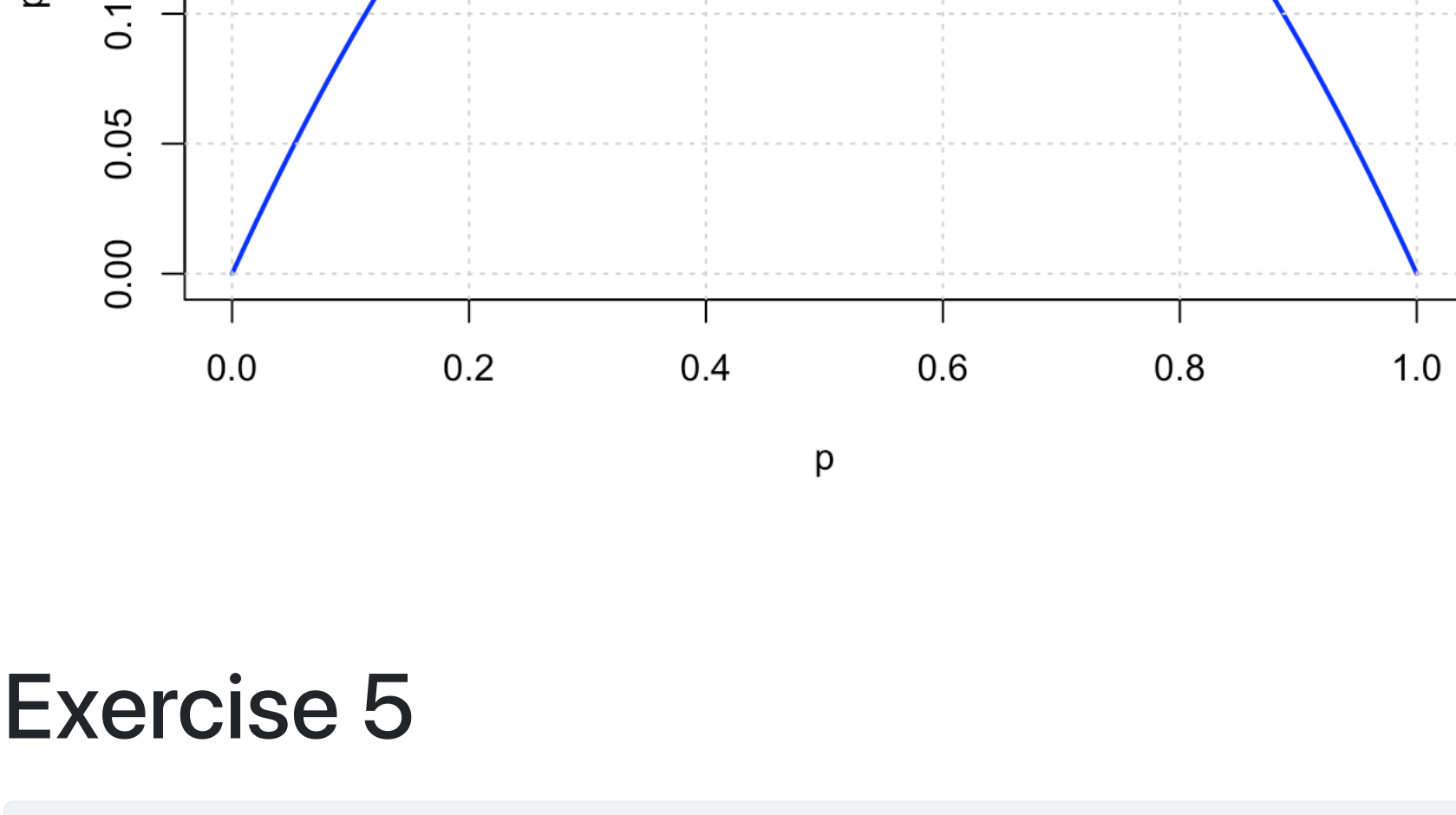
c.

```
p <- seq(from = 0, to = 1, by = 0.01)
f_p <- p * (1 - p)

plot(p, f_p, type = "l", col = "blue", lwd = 2,
     xlab = "p", ylab = "p(1 - p)",
     main = "Plot of f(p) = p(1 - p)")

grid()
```


Plot of $f(p) = p(1 - p)$



Exercise 5

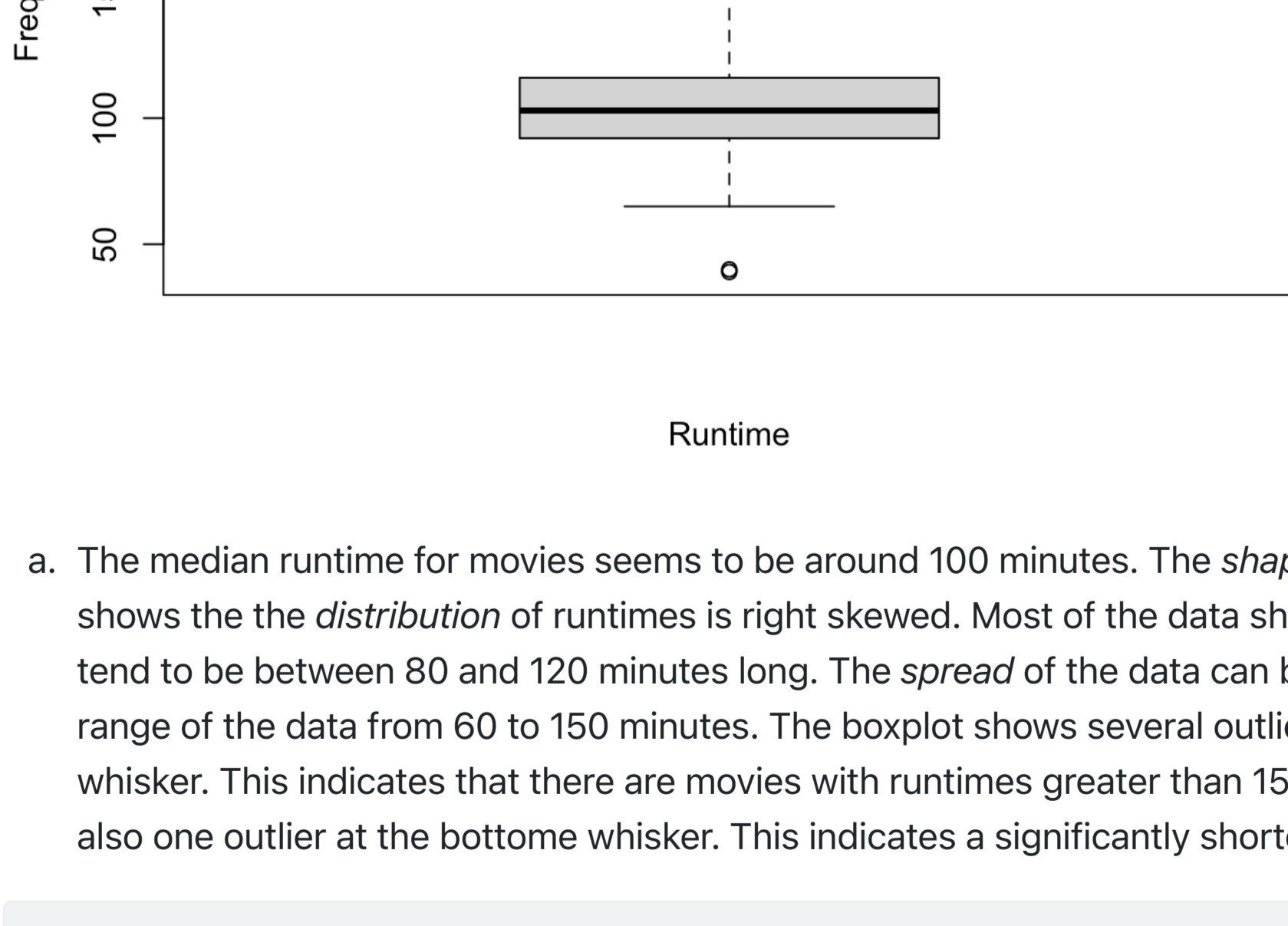
```
movies <- read.csv("https://ericwfox.github.io/data/movies.csv")
hist(movies$runtime, main = "Histogram of Movie Runtimes",
     xlab = "Runtime", ylab = "Frequency")
```

Histogram of Movie Runtimes



```
boxplot(movies$runtime, main = "Boxplot of Movie Runtimes",
        xlab = "Runtime", ylab = "Frequency")
```

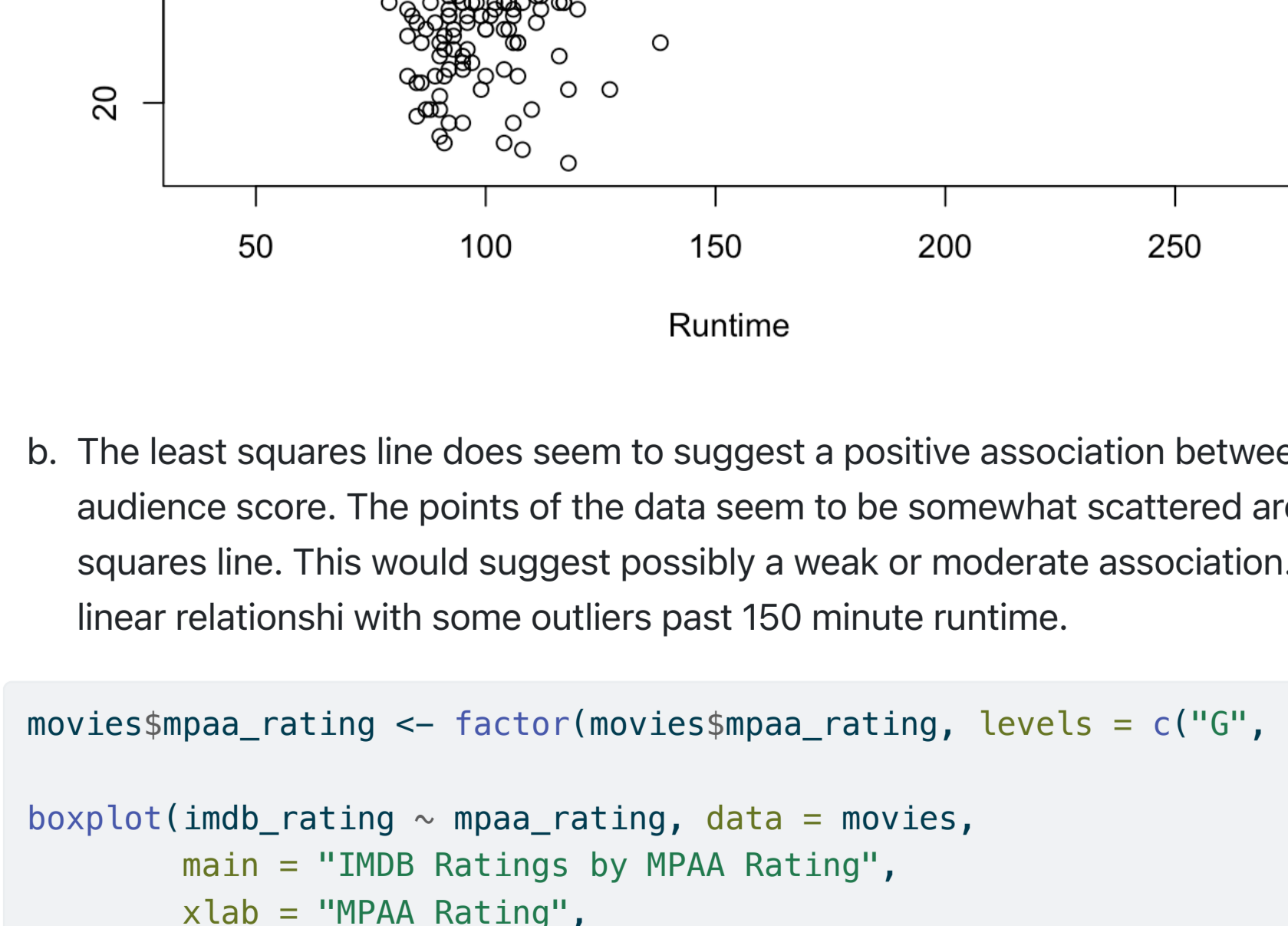
Boxplot of Movie Runtimes



a. Shows the median runtime of movies is around 100 minutes. The *shape* of the histogram shows the *distribution* of runtimes is right skewed. Most of the data shows that movies tend to be between 80 and 120 minutes long. The spread of the data can be observed in the range of the data from 60 to 150 minutes. The boxplot shows several outliers above the whisker. This indicates that there are movies with runtimes greater than 150 minutes. There is also one outlier at the bottom whisker. This indicates a significantly shorter runtime.

```
plot(movies$runtime, movies$audience_score,
     main = "Scatterplot of Runtime vs Audience Score",
     xlab = "Runtime", ylab = "Audience Score")
modell <- lm(movies$audience_score ~ movies$runtime)
abline(modell, col = "red")
```

Scatterplot of Runtime vs Audience Score



b. The least squares line does seem to suggest a positive association between runtime and the audience score. The points of the data seem to be somewhat scattered around the least squares line. This would suggest possibly a weak or moderate association. It does seem a linear relationship with some outliers past 150 minute runtime.

```
movies$mpaa_rating <- factor(movies$mpaa_rating, levels = c("G", "PG", "PG-13", "R", "NC-17", "Unrated"))
boxplot(imdb_rating ~ mpaa_rating, data = movies,
        main = "IMDB Ratings by MPAA Rating",
        xlab = "MPAA Rating", ylab = "IMDB Rating",
        col = c("lightblue", "royalblue", "lightgreen", "green", "yellow", "orange"))
```

IMDB Ratings by MPAA Rating

