Homework Assignment 11 (100 points + 12 points possible bonus)

Armen Mkrtumyan

29/04/23

Write your solutions and comments in this markdown file and submit it with its pdf version to moodle.

WARNING!!! (If not done you will lose points.)
– Make sure to put titles on the plots and texts on axes.
– Descriptions for the graphs, write a few words what you see in the graph. – Your graph should be nice for the eye, according to guidelines - have titles, xlab, ylab, no overlapping texts, no long titles that go out of bounds and so on.

Libraries:

Part 1: Video games dataset

You are given a video games dataset containing information about popular video games, their sales in North America, Europe, Japan and globally in the world.

1.1. Import the *video_games_2.csv* dataset into R. Print the structure of the dataset, by 1-2 sentences write what you see in the structure. (3 point)

```
video_games_df <- read.csv("video_games_2.csv")
str(video_games_df)
```

```
## 'data.frame':    7586 obs. of  15 variables:
##  $ Name        : chr  "Wii Sports" "Mario Kart Wii" "Wii Sports Resort" "New Super Mario Bros." ...
##  $ Platform    : chr  "Wii" "Wii" "Wii" "DS" ...
##  $ Year        : chr  "2006" "2008" "2009" "2006" ...
##  $ Genre       : chr  "Sports" "Racing" "Sports" "Platform" ...
##  $ Publisher   : chr  "Nintendo" "Nintendo" "Nintendo" "Nintendo" ...
##  $ NA_Sales    : num  41.4 15.7 15.6 11.3 14 ...
##  $ EU_Sales    : num  28.96 12.76 10.93 9.14 9.18 ...
##  $ JP_Sales    : num  3.77 3.79 3.28 6.5 2.93 4.7 4.13 3.6 0.24 2.53 ...
##  $ Global_Sales: num  82.5 35.5 32.8 29.8 28.9 ...
##  $ Critic_Score: int  76 82 80 89 58 87 91 80 61 80 ...
##  $ Critic_Count: int  51 73 73 65 41 80 64 63 45 33 ...
##  $ User_Score  : num  8 8.3 8 8.5 6.6 8.4 8.6 7.7 6.3 7.4 ...
##  $ User_Count  : int  322 709 192 431 129 594 464 146 106 52 ...
##  $ Developer   : chr  "Nintendo" "Nintendo" "Nintendo" "Nintendo" ...
##  $ Rating      : chr  "E" "E" "E" "E" ...
```

```
#We see that there are 7568 rows of observations with 15 columns of features
#The types vary -> chr, num, int
#The data structure also shows that we have information about the video game's
#name, year, genre and so on ...
```

1.2. Using dplyr package subset the dataframe by following these instructions:
- remove columns Publisher, JP_Sales (Sales in Japan), Critic_Count, User_Count and Developer,

- multiply the numbers in NA_Sales, EU_Sales and Global_Sales by 1 million as they are given in millions of sales,
- include only those observations for which NA_Sales>=20000, EU_Sales>=20000 and Rating is among "E", "M", "T", "E10+", "AO" (the meanings of abbrevations: Everyone, Mature, Teen, Everyone 10+, Adults Only, they are used in the dataframe by the abbrevations).

(12 points)

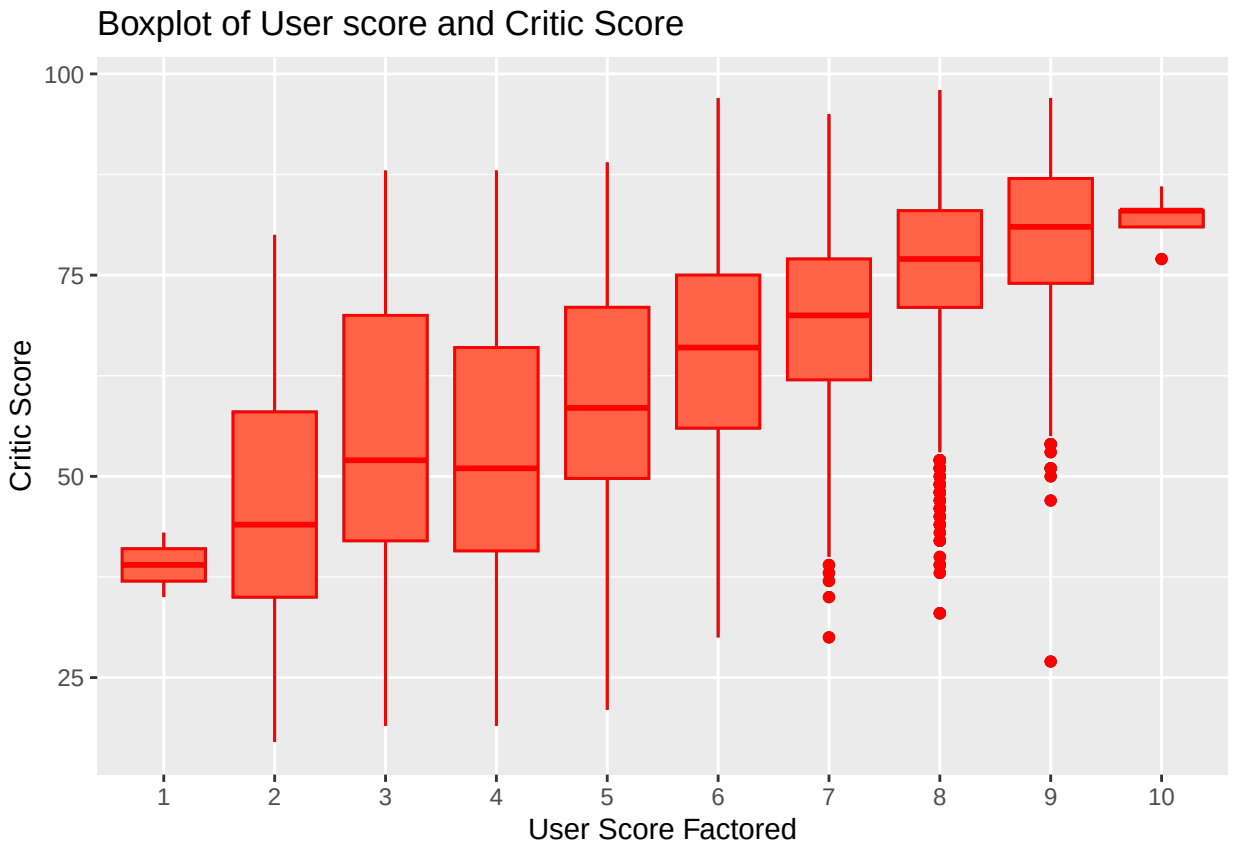
```
video_games <- video_games_df %>%
  select(-c("Publisher", "JP_Sales", "Critic_Count", "User_Count", "Developer"))
video_games <- video_games %>%
  mutate(NA_Sales = NA_Sales * 1000000,
         EU_Sales = EU_Sales * 1000000,
         Global_Sales = Global_Sales * 1000000) %>%
  filter(NA_Sales >= 20000,
         EU_Sales >= 20000,
         Rating %in% c("E", "M", "T", "E10+", "AO"))
```

1.3. Imagine that there are validation rules for 2 columns of our dataframe: "Critic_Score"s can be integer values from 0 to 100 and "User_Score"s can be integers from 1 to 10. To follow these rules, using dplyr, round any non integers values to nearest integer (?round) for these two columns and then filter the dataframe so that only the rows satisfying to validation remain to the new dataframe named *video_games_valid*. Use the obtained dataframe from *exercise 1.2.* (*video_games*) to do this exericise. (7 points)

```
video_games_valid <- video_games %>%
  mutate(Critic_Score = round(Critic_Score),
         User_Score = round(User_Score)) %>%
  filter(Critic_Score >= 0 & Critic_Score <= 100,
         User_Score >=1 & User_Score <= 10)
```

1.4. Use subsetted data *video_games_valid* and create a boxplot displaying how *User_Score*s and *Critic_Score*s are interconnected. Use *User_Score* for the x-axis and as it is a numerical type but has discrete value range (1, 2, . . . , 10), convert it to Factor datatype, then create the boxplot to have better results. Make the boxplot's edges *red* and boxplot's background *tomato*. Explain what you see in the graph in a few words. Then calculate the correlation between these two features to prove your observations from the graph. (10 points)

```
User_Score_Factor <- factor(video_games_valid$User_Score)
ggplot(video_games_valid, aes(x=User_Score_Factor, y=Critic_Score)) +
  geom_boxplot() +
  geom_boxplot(fill = "tomato", color = "red") +
  labs(y= "Critic Score", x = "User Score Factored") +
  ggtitle("Boxplot of User score and Critic Score")
```




```
paste("Correlation score between User_Score and Critic_Score is :", cor(video_games_valid$User_Score, v
```

```
## [1] "Correlation score between User_Score and Critic_Score is : 0.537713400177788"
```

```
#In a graph, we see the boxplots for 10 factors of User Scores (1 to 10) against
#Critic scores. Specifically, the data for 1 through 6 is concentrated closer
#together compared to 7,8,9, where we have outliers far away from where the
#main data is. But still, the data is relatively spread out, which is proved
#by the moderate correlation score 0.538
```

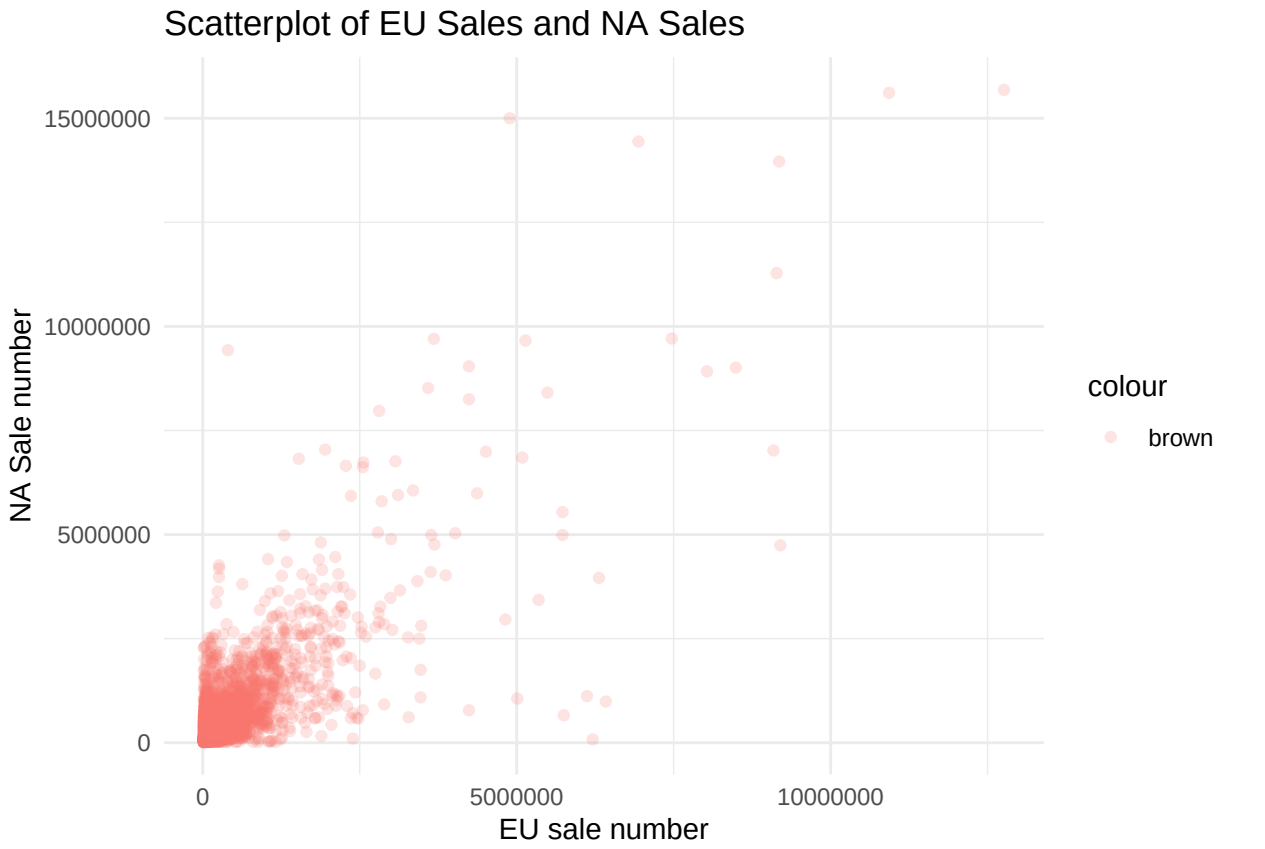
1.5. Construct a scatterplot showing how the *EU_Sales* (Europe Sales) of the game is dependent of *NA_Sales* (North American Sales). Use subsetted dataframe *video_games_valid* from *exercise 1.3.*.
In your graph:
- make the points 20% transperent, and color them with your favorite color,
- zoom in the graph, so the outliers do not prevent you seeing the real picture of the data.

(10 points)

Note: You can explore ?options function to display values without "e" short notation.
Note: Try to "add to your graph" the following line "+ theme_minimal()". (There are a few built-in R themes like this one, that you can use to quickly make your graph more pleasant.)

```
options(scipen = 100)
video_games_valid_outlier <- video_games_valid %>%
  filter(EU_Sales != max(video_games_valid$EU_Sales))
ggplot(data=video_games_valid_outlier,
       aes(x=EU_Sales, y=NA_Sales, color = "brown")) +
  geom_point(alpha = 0.2) +
  labs(y= "NA Sale number", x = "EU sale number") +
```

```
  ggtitle("Scatterplot of EU Sales and NA Sales") +
  theme_minimal()
```

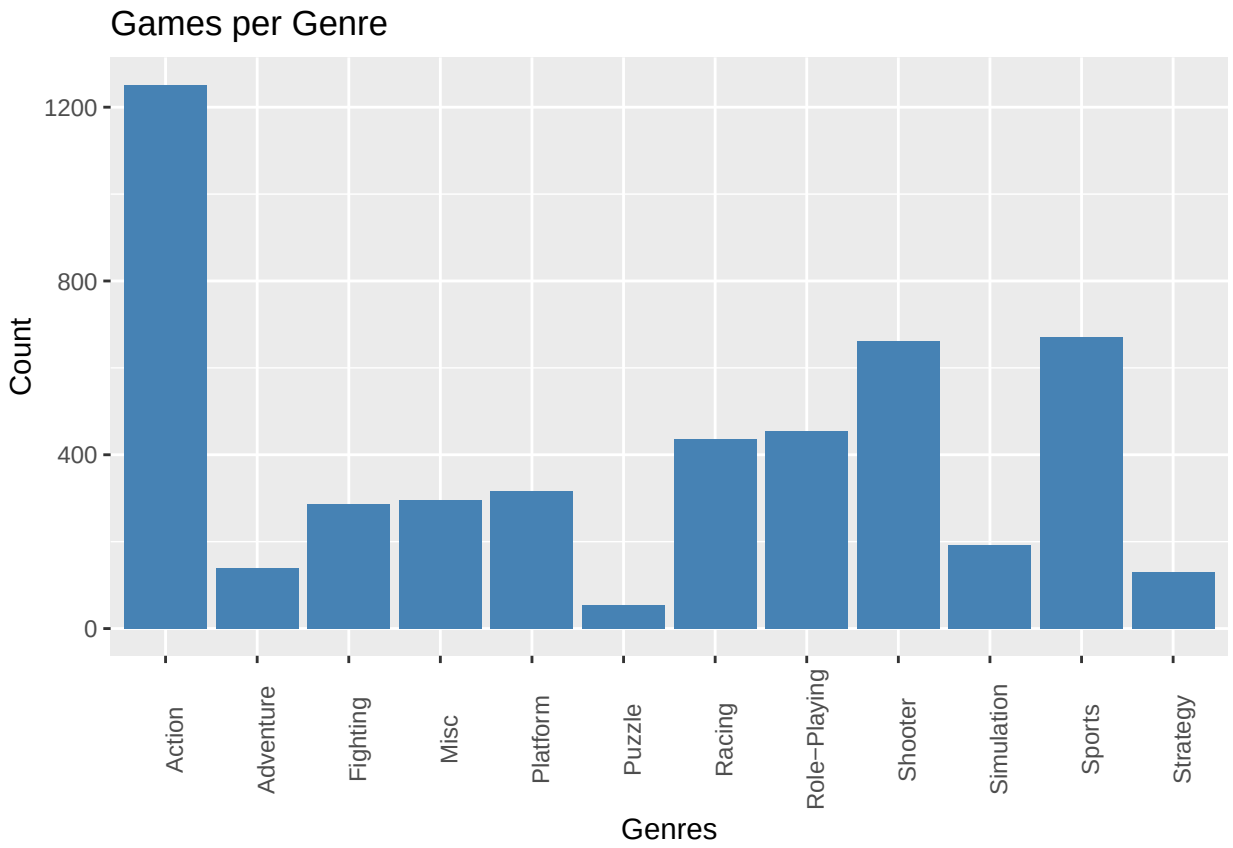



```
#In the scatterplot we see that most of the data is concentrated when both
#EU sale numbers and NA sale numbers are low, correlation is relatively good
```

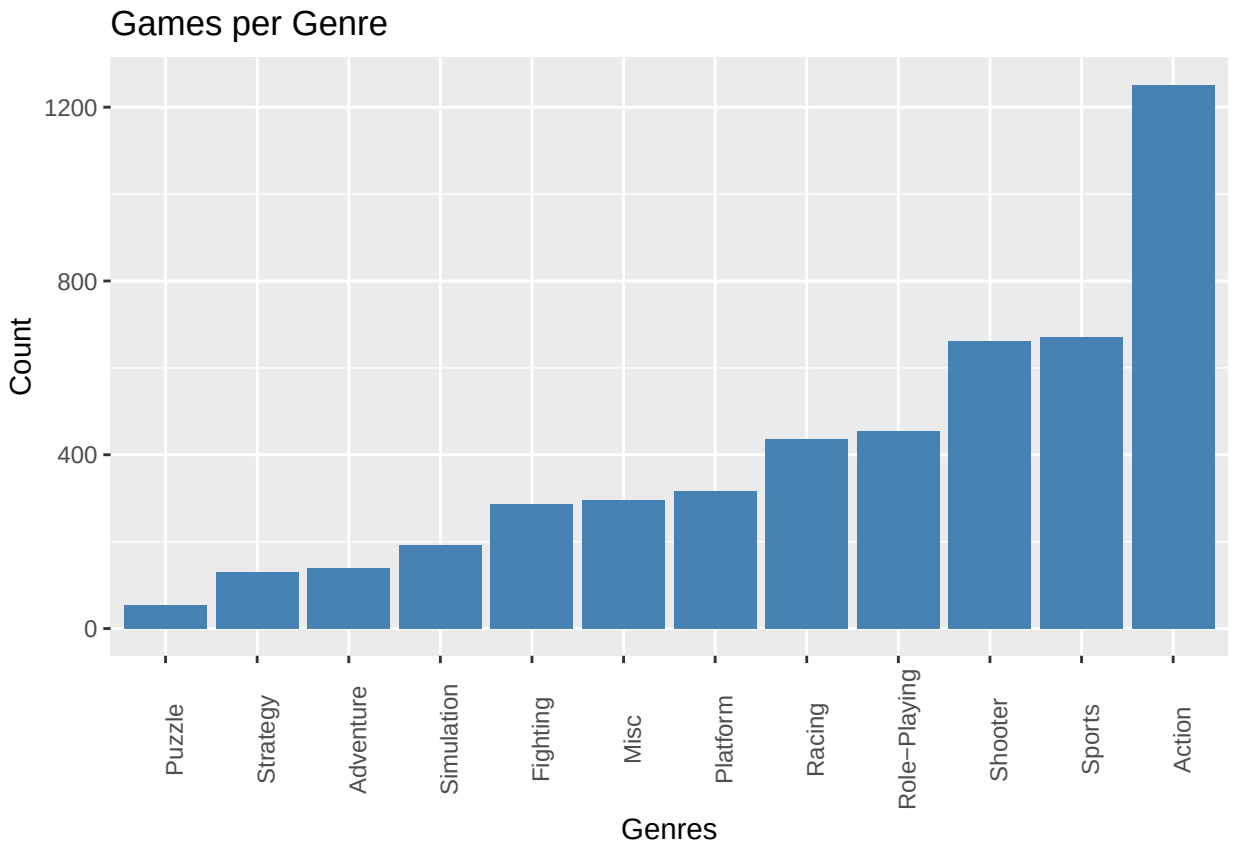
1.6. Create a barplot to find how many games of each Genre there are in the dataset. Use either *valid_video_games* or your initial dataset. State what are the top 3 Genres according to the graph. Rotate Genre names on "x" axis to avoid overlapping text. (7 points)
Hint: ?theme, ?element_text

Exercise to try: Can you order the barplot bars so that they are ordered in decreasing or increasing order of Counts? (+5 bonus points)

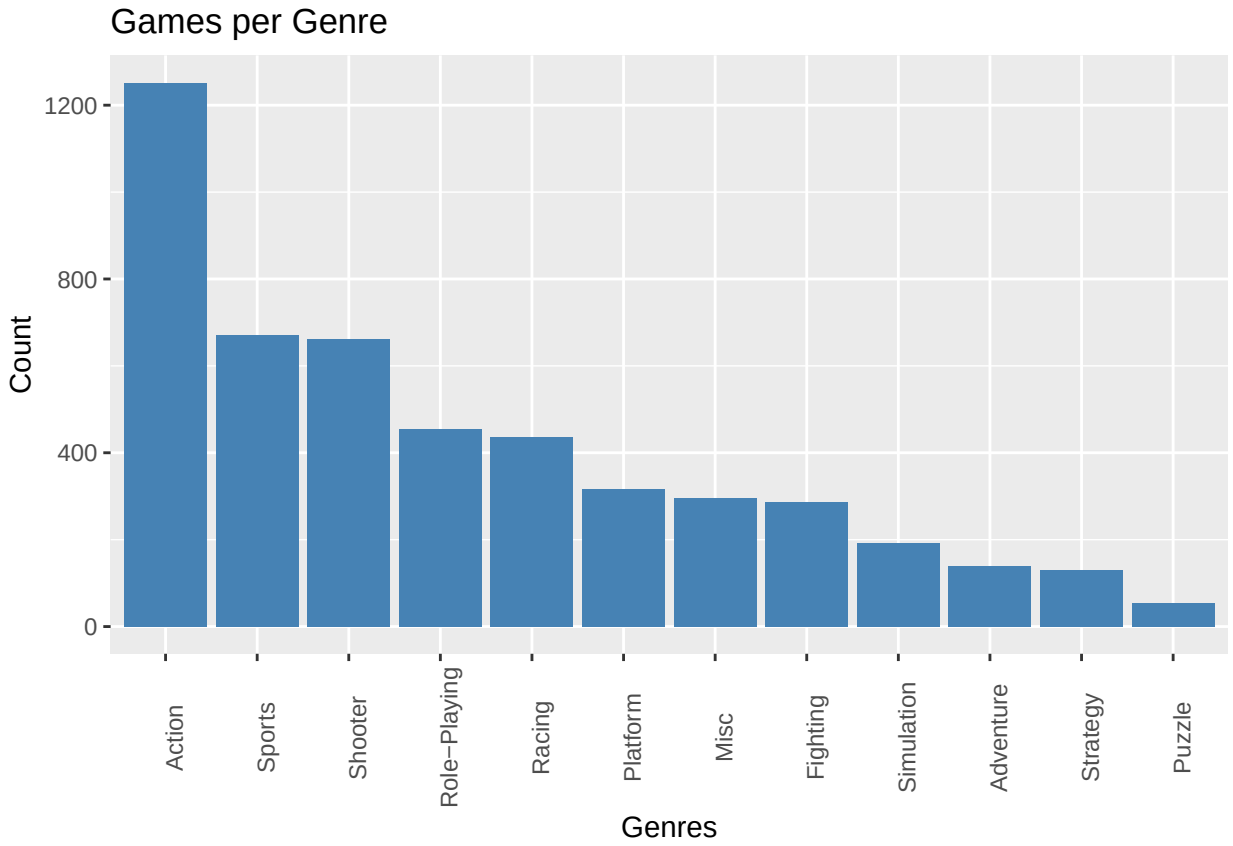
```
ggplot(data = video_games_valid, aes(x = Genre)) +
  geom_bar(fill = "steelblue") +
  theme (axis.text.x = element_text (angle = 90)) +
  ggtitle("Games per Genre") +
  labs(y= "Count", x = "Genres")
```




```
#Descending order
video_games_valid %>%
  count(Genre) %>%
  arrange(desc(n)) %>%
  ggplot(aes(x = reorder(Genre, n), y = n)) +
  geom_bar(fill = "steelblue", stat = "identity") +
  theme(axis.text.x = element_text(angle = 90)) +
  ggtitle("Games per Genre") +
  labs(y = "Count", x = "Genres")
```




```
#Ascending order
video_games_valid %>%
  count(Genre) %>%
  arrange(n) %>%
  ggplot(aes(x = reorder(reorder(Genre, n), -n), y = n)) +
  geom_bar(fill = "steelblue", stat = "identity") +
  theme(axis.text.x = element_text(angle = 90)) +
  ggtitle("Games per Genre") +
  labs(y = "Count", x = "Genres")
```




```
#All the graphs depict a barplot which show the count of genres, first graph
#is scattered and the other two order plot by ascending and descending order
#respectively
```

1.7. Now using dplyr package show the number of video games in each genre in descending order according to the dataset. So you should get the "same result" as in the last exercise, but in a form of a table. (10 points)

```
sort(table(video_games_valid$Genre), decreasing = T)
```

```
##
##       Action       Sports      Shooter Role-Playing       Racing     Platform
##         1252          672          661          455          436          316
##         Misc     Fighting   Simulation    Adventure     Strategy       Puzzle
##          295          287          193          140          129           55
```

1.8. Use dplyr to create a new variable (CU_Score) for each video game that will show the average of Critic score and 10 * User Score (CU_Score = (Critic score + 10 * User Score) / 2). (7 points)

```
video_games_df <- video_games_df %>%
  mutate(CU_Score = (Critic_Score + 10 * User_Score) / 2)
```

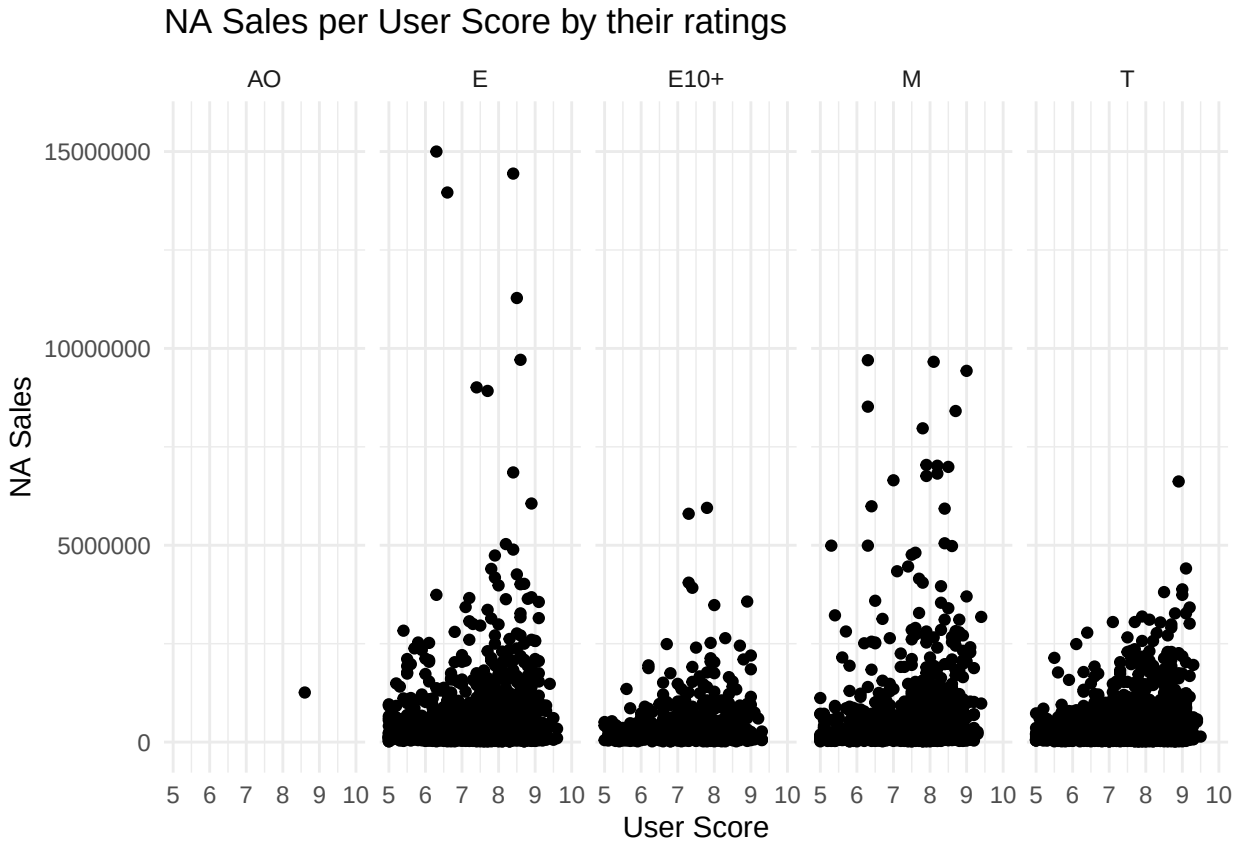
1.9. From this point on: use dataframe from *exercise 1.2.* that was not yet validated.
Define the *Rating* column as Factor (if it is not already so) and use faceting (facet_grid) to plot scatterplot *User_Score* vs *NA_Sales* (North America Sales) for different Ratings. Zoom the graph so User_Score is between 5 and 10, and North American Sales are from 0 to 15500000. Do not forget about the titles and interpretations. (7 points)

```
video_games$Rating <- as.factor(video_games$Rating)

ggplot(video_games, aes(x = User_Score, y = NA_Sales)) +
```

```
  geom_point() +
  facet_grid(~Rating) +
  xlim(5, 10) +
  ylim(0, 15500000) +
  labs(x = "User Score", y = "NA Sales") +
  ggtitle("NA Sales per User Score by their ratings") +
  theme_minimal()
```

```
## Warning: Removed 394 rows containing missing values (`geom_point()`).
```




```
#As far as I understand, we get a warning because we limited
# x and y to certain values
#The graph shows the user scores factored by their ratings, which have the
#same y values for them -> Sales in North America, for example we see that AO
#only had a 1 value, it is an outlier in our data
```

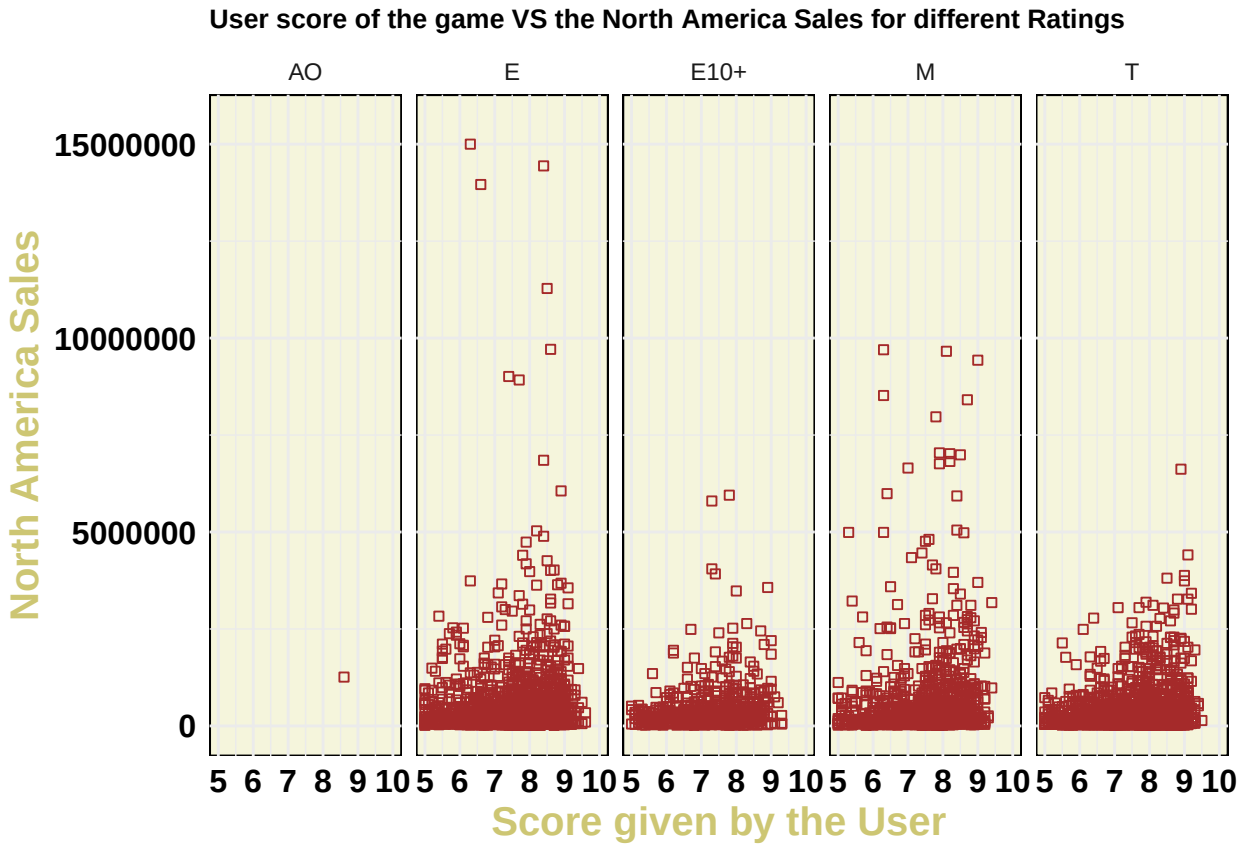
1.10. Make these design changes on the plot from the previous exercise:
- x axis name – "Score given by the User" color: "khaki3", bold, size: 15,
- y axis name – "North America Sales" color: "khaki3", bold, size: 15,
- points (shape - square, color: "brown", size: 1.5),
- title of the plot – "User score of the game VS the North America Sales for different Ratings",
- panel background color: "beige",
- axis texts bold black.

(7 points)

Note: If you have not done the last exercise successfully, you can take any graph from this homework, to make changes on.

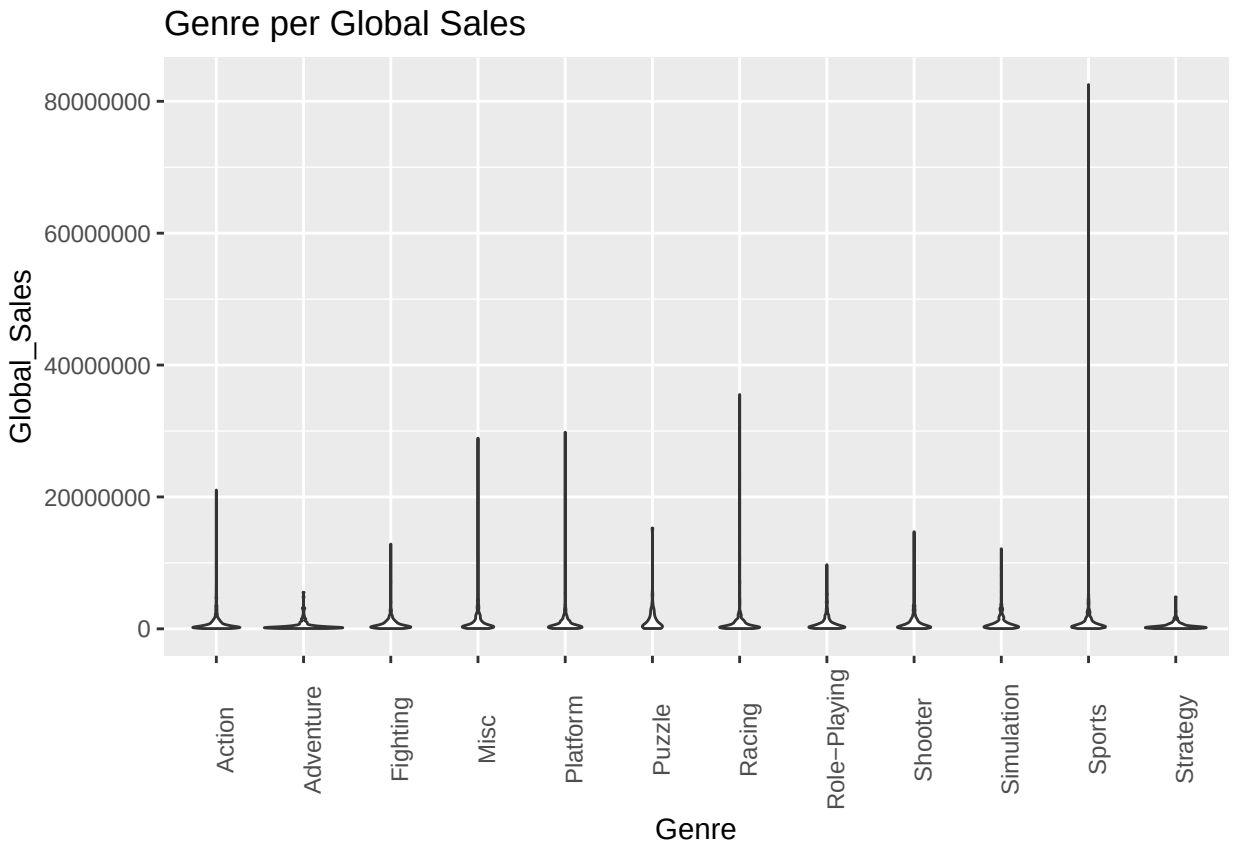
```
ggplot(video_games, aes(x = User_Score, y = NA_Sales)) +
  geom_point(shape = 22, size = 1.5, color = "brown") +
  facet_grid(~Rating) +
  xlim(5, 10) +
  ylim(0, 15500000) +
  labs(x = "Score given by the User", y = "North America Sales") +
  ggtitle("User score of the game VS the North America Sales for different Ratings") +
  theme_minimal() +
  theme(axis.title = element_text(size = 15, color = "khaki3", face = "bold"),
        axis.text = element_text(size = 12, face = "bold", color = "black"),
        plot.title = element_text(size = 10, face = "bold"),
        panel.background = element_rect(fill = "beige"),)
```

```
## Warning: Removed 394 rows containing missing values (`geom_point()`).
```



1.11. Create a violin-plot (it is a plot like the boxplot, google to see the details) where x-axis represents the *Genre* and y-axis the *Global_Sales* of the video game for a particular Genre. Make the text on "x" axis vertical. Make comments about results. (7 points)
Hint: ?theme, ?element_text

```
ggplot(data = video_games, aes(x = Genre, y = Global_Sales)) +
  geom_violin() +
  theme(axis.text.x = element_text(angle = 90)) +
  ggtitle("Genre per Global Sales") +
  labs(x = "Genre", y = "Global_Sales")
```



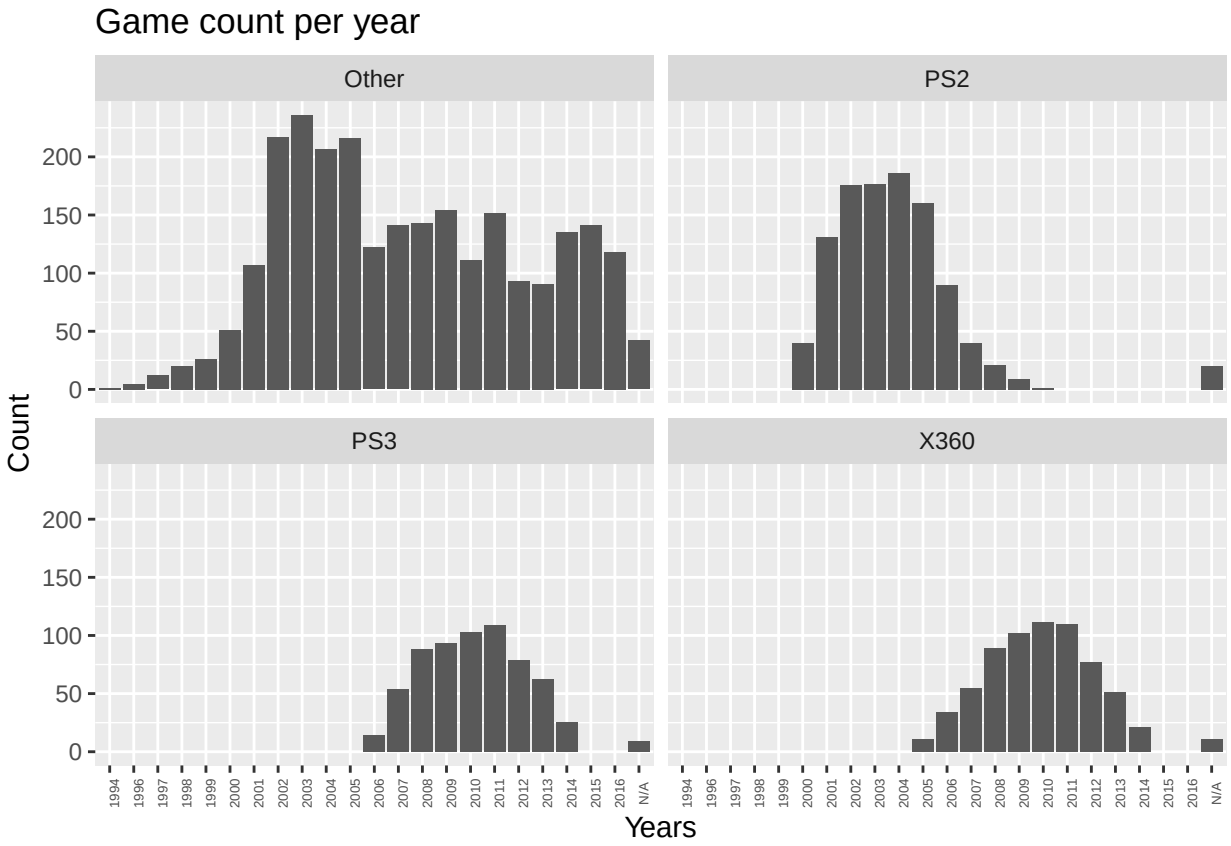

```
#Since in violin plots, the width of each "violin" represents the density of
# data points at each value, we can see that Adventure has the biggest density,
#There are more data points on those thicker parts. The shape of the violin is
#also an indication of the distribution of the data - a symmetric distribution
# will have a violin that is roughly symmetrical. I do not see any symmetric
#violins in this data
```

1.12. We are interested in the number of video games developed for platforms *'PS2', 'X360', 'PS3'* for different years.
- Make other platforms, that are not from these 3 as "Other" using dplyr (Hint: ifelse statement).
- Remove all observations from dataframe which have any NA values (Hint: ?complete.cases).
- Use faceting to draw the distribution of games for each year for each platform. Make text on "x" axis vertical and size=5.
Make comments how the number of video games changed for each platform for different years.
(20 points)

```
video_games <- video_games %>%
  mutate(Other = ifelse(Platform %in% c("PS2", "X360", "PS3"), Platform, "Other"))

video_games <- video_games[complete.cases(video_games),]

ggplot(video_games, aes(x=Year)) +
  geom_bar() +
  facet_wrap(~ Other) +
  theme(axis.text.x = element_text(angle = 90, size=5))+
  ggtitle("Game count per year") +
  labs(x = "Years", y = "Count")
```




```
#FOR OTHER TYPES -> The number of video games produced increased for small
#period of time, then it got decreased starting from about 2005
#For PS2, the number of games increased till 2004, then it decreased
#For PS3, the number of games increased from 2006 till 2011, then it quickly
#dropped
#Almost the same graph is with x360 as with PS3
```