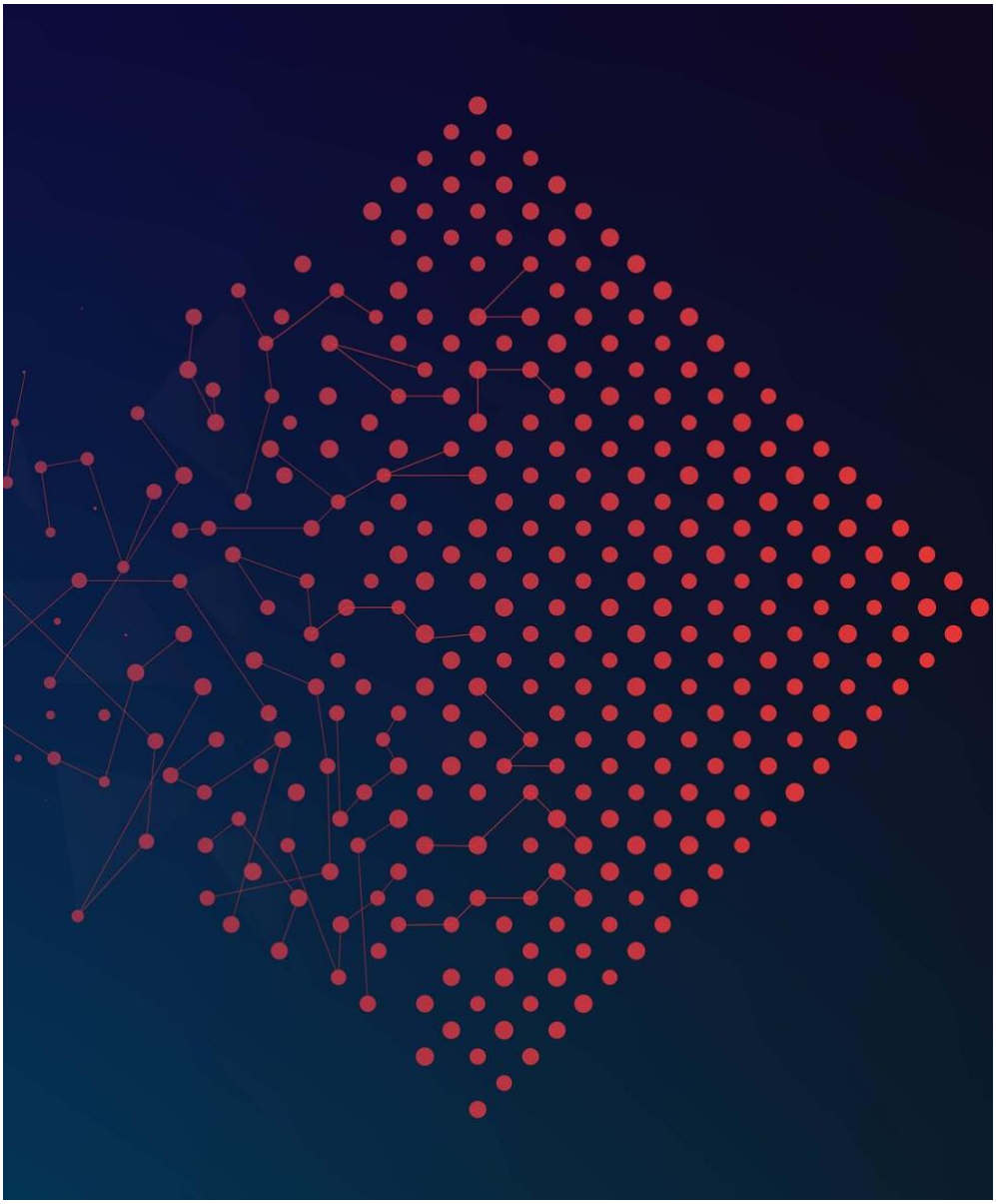
ASSIGNMENT 3:
TURTLE GAMES PROJECT
SEPTEMBER 2022



Javier Conde Pascual

LSE Data Analytics Career Accelerator
London School of Economics and Political Science

I, Javier Conde Pascual, certify that this is an original piece of work. I have acknowledged all sources and citations. No section of this assignment has been plagiarized.

TABLE OF CONTENTS

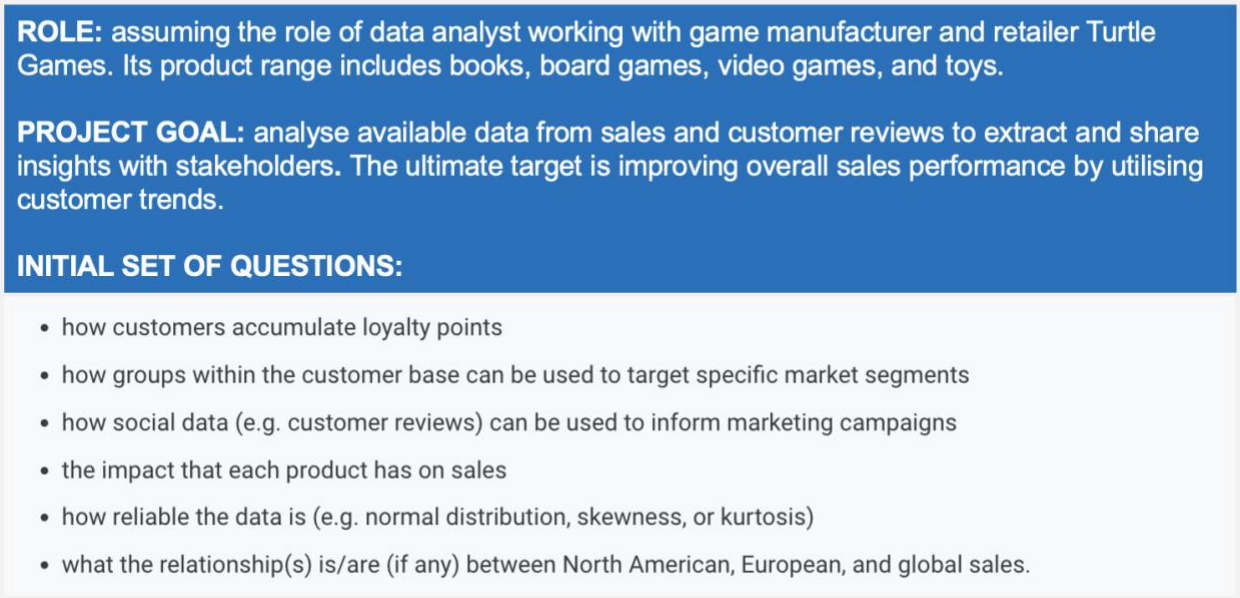
Word count: 933
Own words: 933

Chapter 1

Scenario and system set up

Background, scenario, and context provided beforehand for this report are available here. The business problem and key questions for the analysts' team and the data are also provided, Figure 1.1. below.

Figure 1.1: Background, project goal and initial set of questions to answer



Source: LSE (2022)

All project files are stored in GitHub to control overall project progression, different steps completed along the way and any changes made to the Python and R (using Jupyter/RStudio) code blocks.

Now the central Python/R repository in Github where the project files are stored is prepared, let's study the analytical approach used.

Chapter 2

Analytical approach

This project approach has four key points:

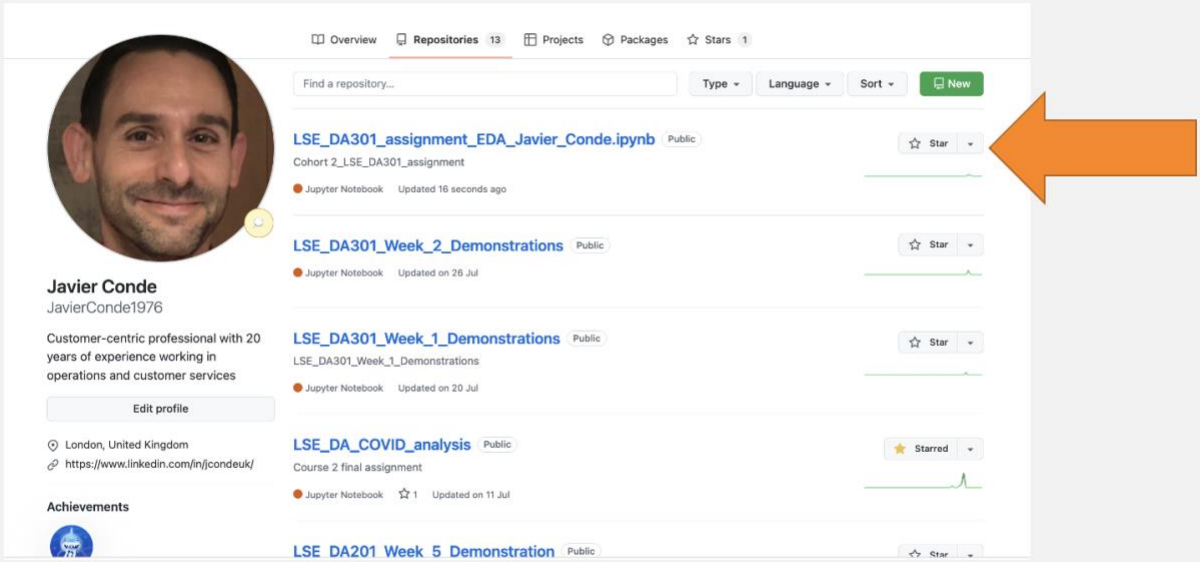
1. Bringing stakeholders (marketing and sales departments) and data analysis team together through understanding the importance and risk-reducing advantages of methodical and thorough data analysis and data quality processes
2. Provide stakeholders with rigorous, accurate and realistic analysis on the data files provided through Python and R:

    - turtle_sales.csv (details of video games sold globally)
    - turtle_reviews.csv (details on customer reviews across products)

    Aim is providing actionable insights and recommendations from them to bring measurable value to the business to reach the project goal.
3. Assessing the quality of the data and results obtained. To achieve best results this project will be divided in two parts: customer data analysis and financial analysis on company sales.
4. Reports/presentations material to be created as accessible as possible through very legible fonts (Arial), big font sizes (10-12 onwards when possible), clear head titles and colors that are color-blind and color-contrast friendly among other features.

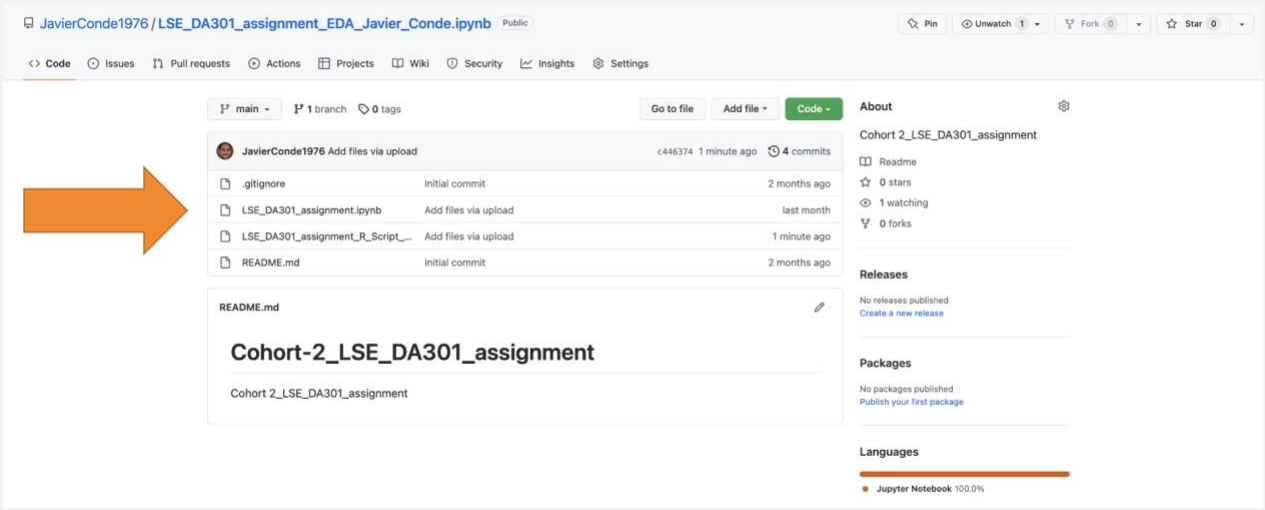
With file preparation for Github, Phyton (Anaconda/Jupyter) and R (RStudio) completed (Figures 1.2 and 1.3 below), let’s now analyze and visualize the data provided.

Figure 1.2: Project Github repository (I)



Source: Javier Conde (2022)

Figure 1.3: Project Github repository (II)



Source: Javier Conde (2022)

2.1 Customer data analysis

2.1.1. First approach and insights to provided data

The file turtle_reviews is cleaned, re-imported, and visualized (Figures 2.1, 2.2, 2.3).

Figure 2.1: Data clean and re-import process (I)

```
In [2]: # Load the CSV file(s) as reviews.

        # Read the 'salary_data.csv' file.
        data = pd.read_csv('turtle_reviews.csv')

        # Print the table.
        data.head()
```

Out[2]:

	gender	age	remuneration (k£)	spending_score (1-100)	loyalty_points	education	language	platform	product	review	summary
0	Male	18	12.30	39	210	graduate	EN	Web	453	When it comes to a DM's screen, the space on t...	The fact that 50% of this space is wasted on a...
1	Male	23	12.30	81	524	graduate	EN	Web	466	An Open Letter to GaleForce9*:\n\nYour unpaint...	Another worthless Dungeon Master's screen from...
2	Female	22	13.12	6	40	graduate	EN	Web	254	Nice art, nice printing. Why two panels are f...	pretty, but also pretty useless
3	Female	25	13.12	77	562	graduate	EN	Web	263	Amazing buy! Bought it as a gift for our new d...	Five Stars
4	Female	33	13.94	40	366	graduate	EN	Web	291	As my review of GF9's previous screens these w...	Money trap

Source: Javier Conde (2022)

Figure 2.2: Data clean and re-import process (II)

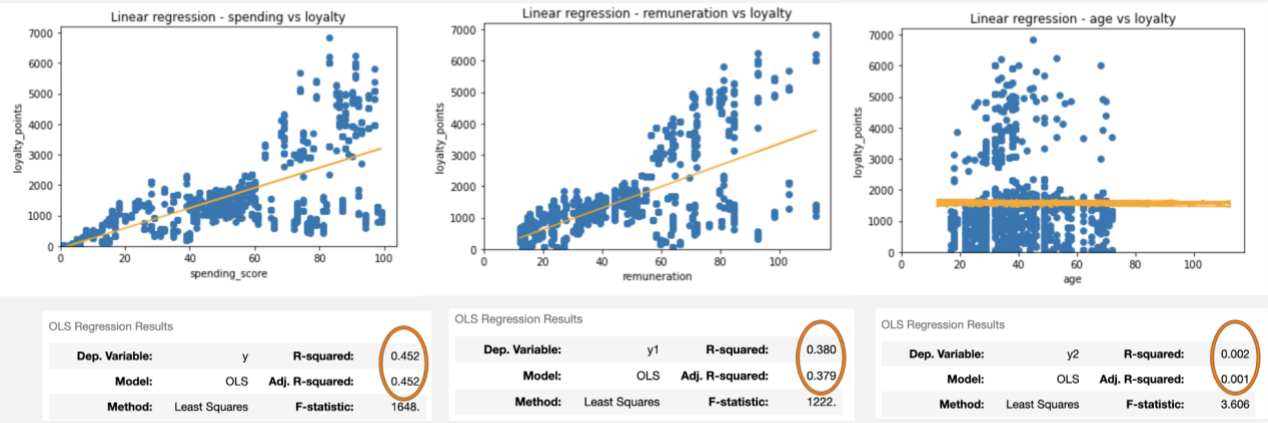
```
In [11]: # Create a CSV file as output.
         data.to_csv(r'turtle_reviews_clean.csv', index = False)

In [12]: # Import new CSV file with Pandas.
         # Read the 'salary_data.csv' file.
         data_new = pd.read_csv('turtle_reviews_clean.csv')
         # View DataFrame.
         data_new.head()
```

Out[12]:

	gender	age	remuneration	spending_score	loyalty_points	education	product	review	summary
0	Male	18	12.30	39	210	graduate	453	When it comes to a DM's screen, the space on t...	The fact that 50% of this space is wasted on a...
1	Male	23	12.30	81	524	graduate	466	An Open Letter to GaleForce9*:\n\nYour unpaint...	Another worthless Dungeon Master's screen from...
2	Female	22	13.12	6	40	graduate	254	Nice art, nice printing. Why two panels are f...	pretty, but also pretty useless
3	Female	25	13.12	77	562	graduate	263	Amazing buy! Bought it as a gift for our new d...	Five Stars
4	Female	33	13.94	40	366	graduate	291	As my review of GF9's previous screens these w...	Money trap

Source: Javier Conde (2022)

Figure 2.3: Loyalty vs spending, remuneration, age



OLS Regression Results

Dep. Variable:	y	R-squared:	0.452
Model:	OLS	Adj. R-squared:	0.452
Method:	Least Squares	F-statistic:	1648.

OLS Regression Results

Dep. Variable:	y1	R-squared:	0.380
Model:	OLS	Adj. R-squared:	0.379
Method:	Least Squares	F-statistic:	1222.

OLS Regression Results

Dep. Variable:	y2	R-squared:	0.002
Model:	OLS	Adj. R-squared:	0.001
Method:	Least Squares	F-statistic:	3.606

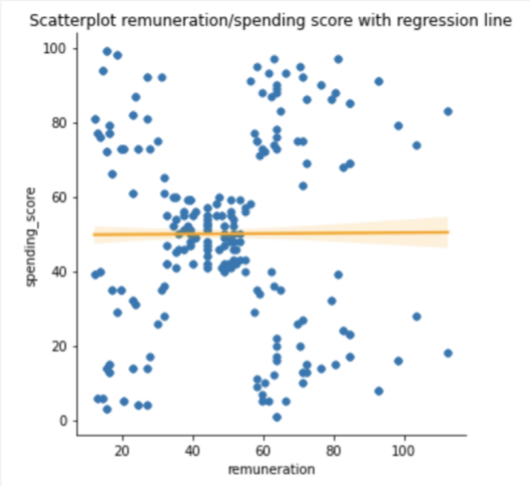
Source: Javier Conde (2022)

There is a slight correlation between variables ‘spending’ and ‘remuneration’ with ‘loyalty’ (Adjusted R^2 measure 0.45 (spending) and 0.37 (remuneration)) but almost no correlation with ‘age’ (Adjusted R^2 0.001). As marketing department requests to study ‘spending’/’remuneration’ further, let’s now identify groups within the customer base that can be used to target specific market segments regarding both variables through k-means clustering.

2.1.2. Customer Clustering

Let’s study further ‘spending’/’remuneration’ relationship with Seaborn using a regression scatterplot and a pairplot (Figures 2.4 and 2.5).

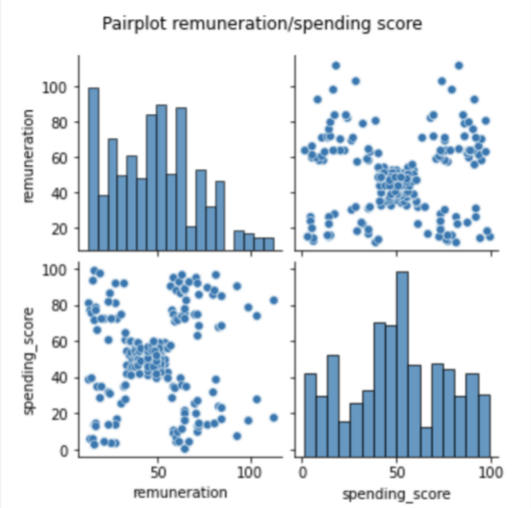
Figure 2.4: Remuneration/spending score Seaborn study: regression scatterplot




```
In [36]: # Scatterplot with a regression line.
         sns.lmplot('remuneration', 'spending_score', data=df2, fit_reg=True, scatter_kws={"marker": "D", "s": 20},
                    line_kws={"color": "orange"})

         plt.title('Scatterplot remuneration/spending score with regression line')
         plt.xlabel('remuneration')
         plt.ylabel('spending_score')
         plt.show()
```

Source: Javier Conde (2022)

Figure 2.5: Remuneration/spending score Seaborn study: pairplot



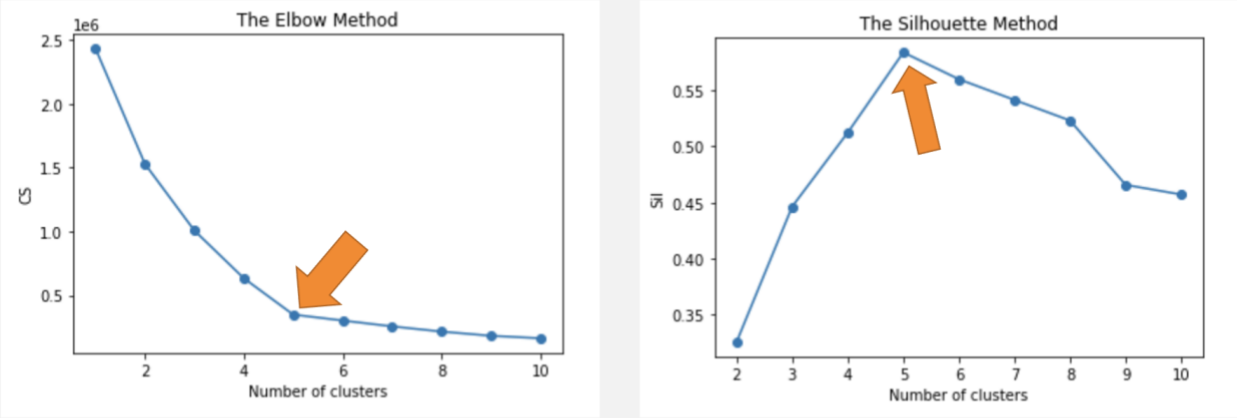

```
In [37]: # Create a pairplot with Seaborn.
         pp = sns.pairplot(df2)
         pp.fig.subplots_adjust(top=.9)
         pp.fig.suptitle('Pairplot remuneration/spending score')
```

Source: Javier Conde (2022)

The pairplot uncovers four/five possible customer clusters that could be targeted. Through the Elbow and Silhouette methods it is determined that optimal cluster number

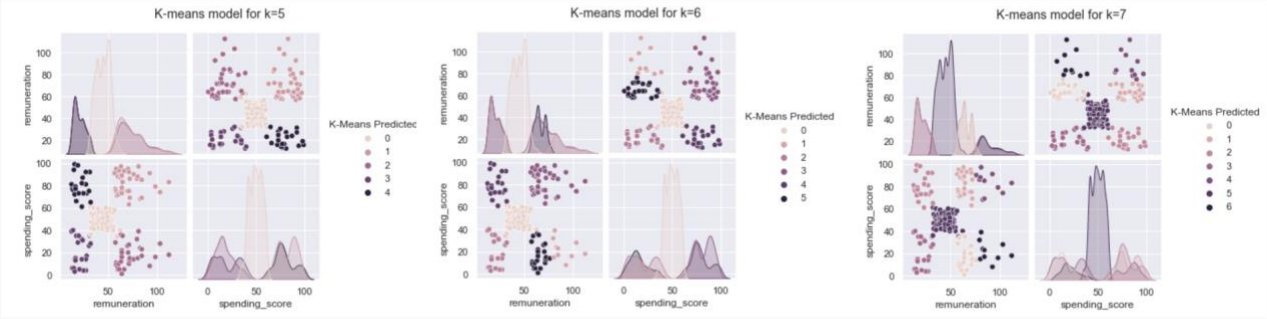
('k') is 5 (Figure 2.6). Let's evaluate suitability of k=5 and other alternatives (i.e., 6/7) through k-means modelling.

Figure 2.6: k=5 (Elbow and Silhouette methods)



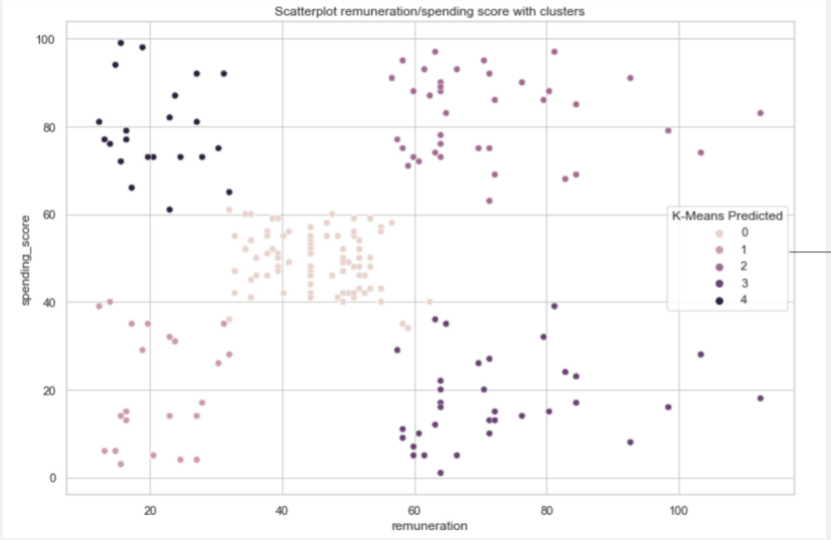
Source: Javier Conde (2022)

Figure 2.7: K-means model for k=5, k=6, k=7



Source: Javier Conde (2022)

Figure 2.8: K-means chosen model for k=5 with cluster interpretation



Source: Javier Conde (2022)

Figure 2.7 and Figure 2.8 show that k=5 returns the best customer grouping as it provides a good separation between groups. Therefore, it would be recommended to Turtle Games' marketing department to consider these 5 possible customer clusters in relation to their remuneration and spending score (as also evidenced by the result of the Elbow/Silhouette methods) among other variables.

As the marketing department has requested more information about reviews to approach future campaigns, let's study the reviews file provided and sentiment behind them through Natural Language Processing (NPL) analysis.

2.1.3. NPL using Python on customer reviews

Focus of the analysis will be 'review' and 'summary' columns from the 'turtle_reviews file'. A new data frame is created only with them, removing lower cases, punctuation, and cleaning duplicates prior to tokenization (Figure 2.9). Then a word cloud with stop words removed is created for visibility (Figure 2.10). Analysis on count of 15 most frequent words and polarity/sentiment in both columns follows (Figure 2.11).

Figure 2.9: Data frame with 'review' /'summary' columns clean and ready for tokenisation

Out[66]:

	review	summary
0	when it comes to a dms screen the space on the...	the fact that 50 of this space is wasted on ar...
1	an open letter to galeforce9 your unpainted mi...	another worthless dungeon masters screen from ...
2	nice art nice printing why two panels are fill...	pretty but also pretty useless
3	amazing buy bought it as a gift for our new dm...	five stars
4	as my review of gf9s previous screens these we...	money trap
...	...	...
1995	the perfect word game for mixed ages with mom ...	the perfect word game for mixed ages with mom
1996	great game did not think i would like it when ...	super fun
1997	great game for all keeps the mind nimble	great game
1998	fun game	four stars
1999	this game is fun a lot like scrabble without a...	love this game

1961 rows × 2 columns

```
In [67]: # Study changes after dropping duplicates.
         df3_clean.info()
         df3_clean.shape
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1961 entries, 0 to 1999
Data columns (total 2 columns):
 #   Column   Non-Null Count  Dtype
---  ------   --------------  -----
 0   review   1961 non-null   object
 1   summary  1961 non-null   object
dtypes: object(2)
memory usage: 46.0+ KB
```

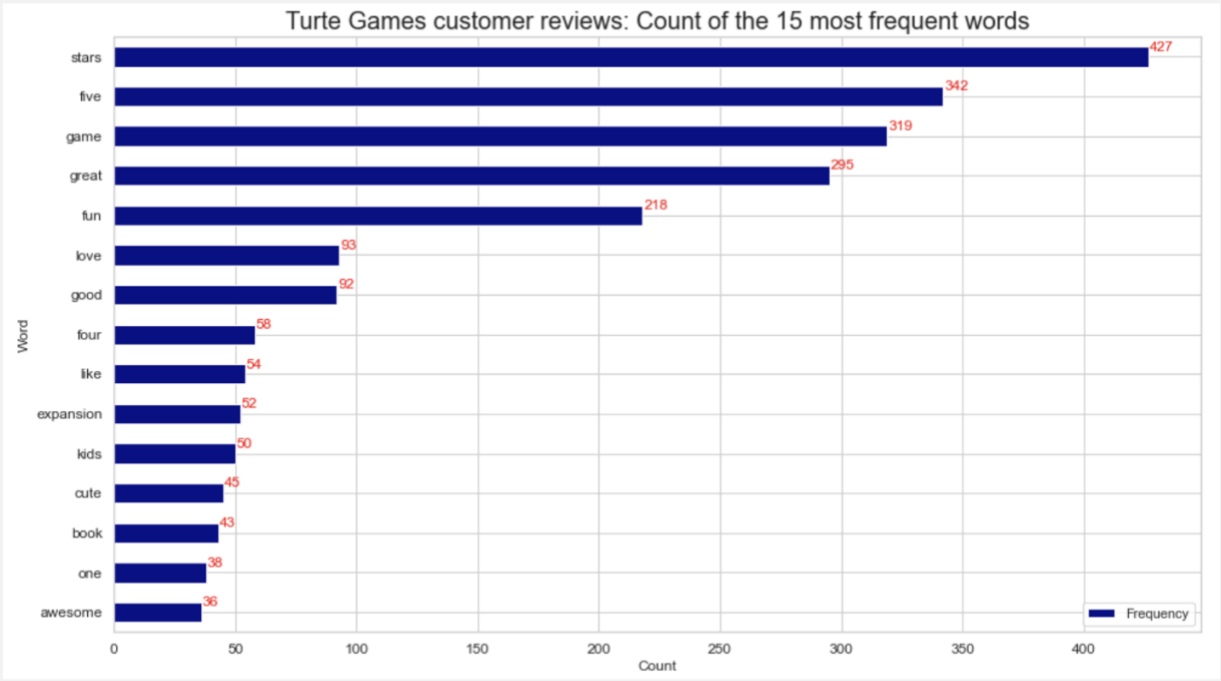
```
Out[67]: (1961, 2)
```

Source: Javier Conde (2022)

Figure 2.10: Word cloud without stopwords from data frame

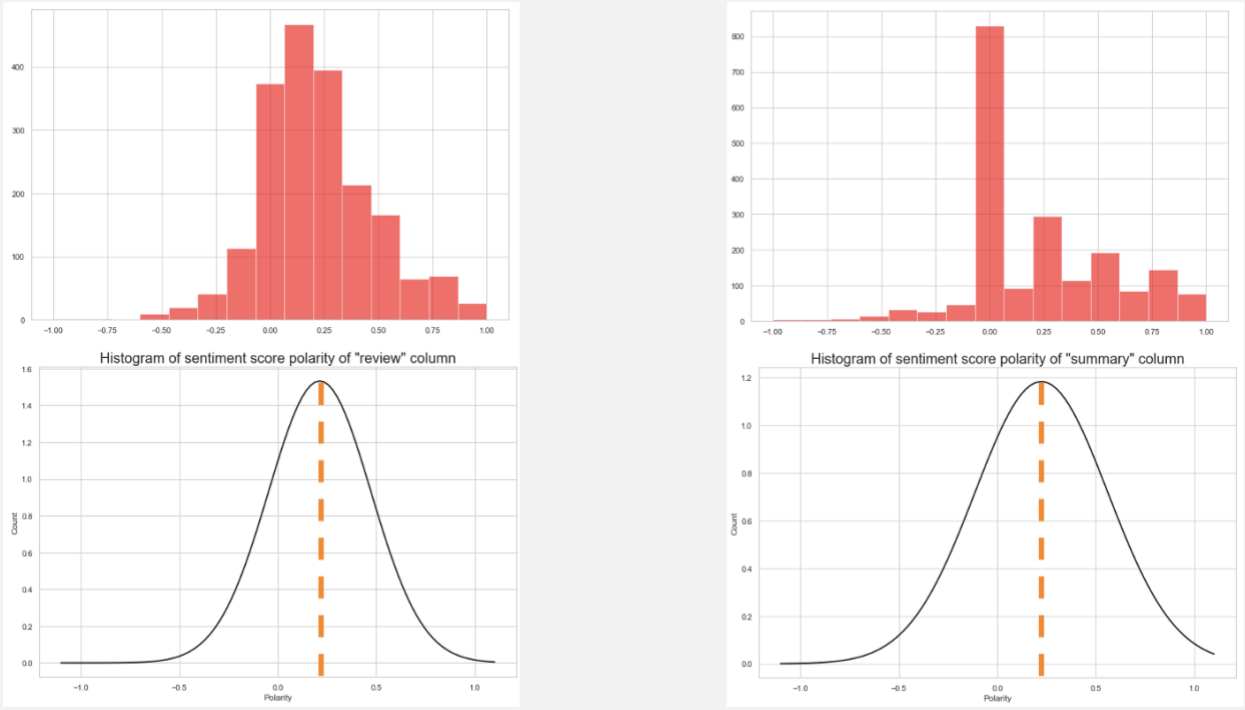
Source: Javier Conde (2022)

Figure 2.11: Plot for the data frame 15 most frequent words



Source: Javier Conde (2022)

Figure 2.12: Sentiment score polarity analysis for columns 'review' and 'summary'



Source: Javier Conde (2022)

The first most frequent words being “stars” and “five” and sentiment/polarity score Mean in both columns between +0.2 to +0.25 signal to a slightly positive sentiment towards Turtle Games products. Further analysis on customer reviews to better understand points of improvement and ‘best of’ comments is recommended.

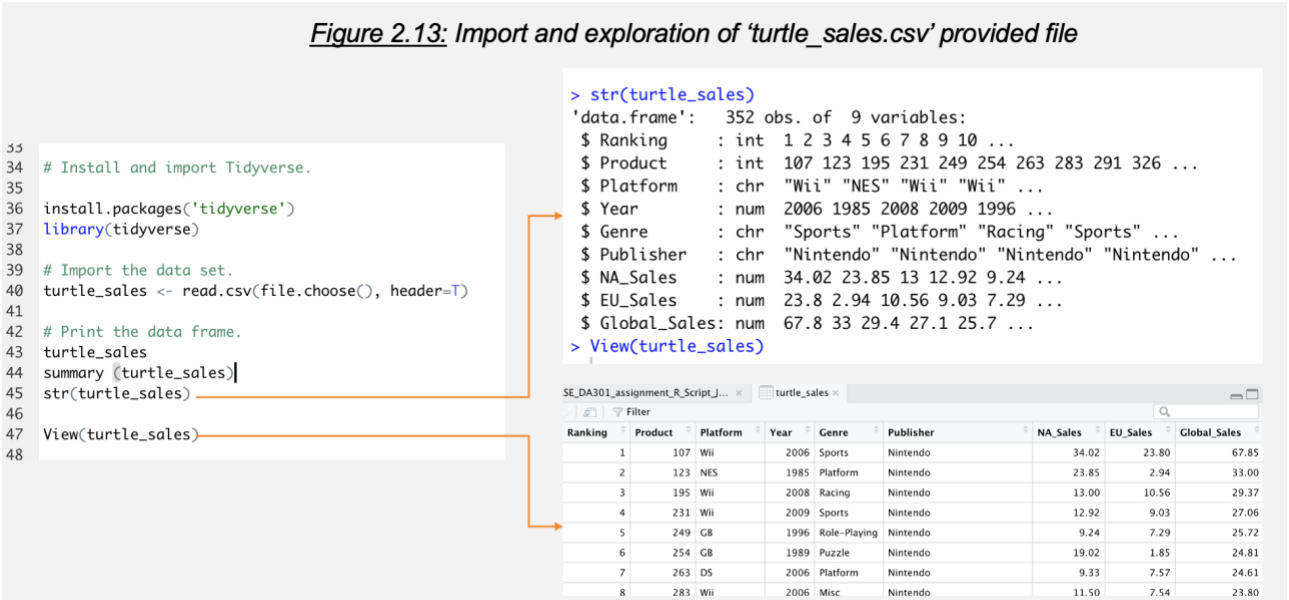
Let’s now analyze sales performance from the company products, this time using R as it is the preferred tool for the sales department.

2.2 Sales data analysis

2.2.1. Exploratory Data Analysis (EDA) on sales with R

After importing the relevant packages and exploring structure and summary of the provided data (‘turtle_sales’) (Figure 2.13), a subset is created (‘turtle_sales2’) containing products ID, creation platforms, and sales figures (European Union (EU), North America (NA) and global) (Figure 2.14).

Figure 2.13: Import and exploration of ‘turtle_sales.csv’ provided file



Source: Javier Conde (2022)

Figure 2.14: Data cleaning and subset creation: NA, duplicated observations, unnecessary columns

```
49  # Explore NA and duplicates.
50
51  is.na(turtle_sales)
52  apply(is.na(turtle_sales), 2, which)
53
54  # Only two NA in 'Year' column, rows 180 and 258. Remove these columns in next
55  # step so no action taken.
56
57  duplicated(turtle_sales)
58
59  # No duplicated observations.
60
61  # Create a new data frame from a subset of the sales data frame.
62  # Remove unnecessary columns (Ranking, Year, Genre, Publisher).
63
64  turtle_sales2 <- subset(turtle_sales, select=-c(Ranking, Year, Genre, Publisher))
65
66  # View the data frame and structure.
67
68  turtle_sales2
69  str(turtle_sales2)
70
71  # View the descriptive statistics.
72
73  summary(turtle_sales2)
74
```

```
> apply(is.na(turtle_sales), 2, which)
$Ranking
integer(0)

$Product
integer(0)

$Platform
integer(0)

$Year
[1] 180 258

$Genre
integer(0)

$Publisher
integer(0)

$NA_Sales
integer(0)

$EU_Sales
integer(0)

$Global_Sales
integer(0)
```

```
> duplicated(turtle_sales)
  [1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [12] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [23] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [34] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [45] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [56] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [67] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [78] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [89] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[100] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[111] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[122] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[133] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[144] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[155] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[166] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[177] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[188] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[199] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[210] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[221] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[232] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[243] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[254] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[265] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[276] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[287] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[298] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[309] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[320] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[331] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[342] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
```

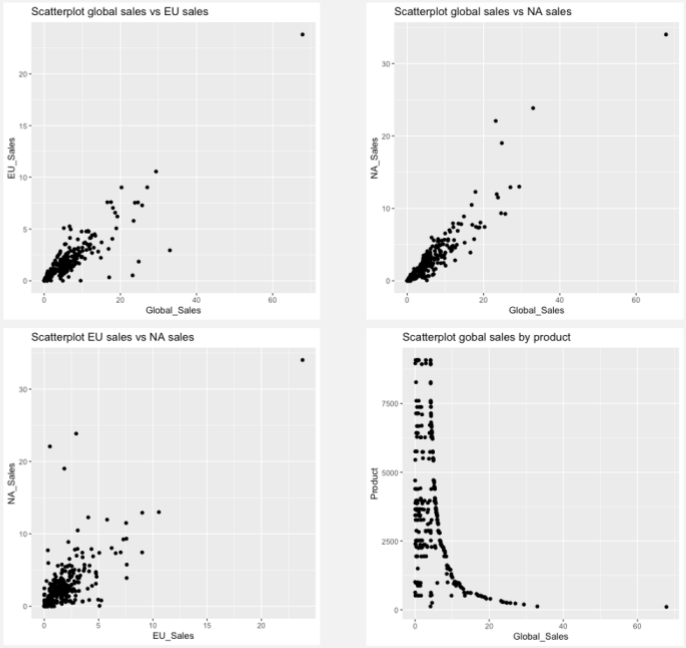
```
> summary(turtle_sales2)
    Product        Platform           NA_Sales         EU_Sales       Global_Sales
 Min.   : 107   Length:352         Min.   : 0.0000   Min.   : 0.000   Min.   : 0.010
 1st Qu.:1945   Class :character   1st Qu.: 0.4775   1st Qu.: 0.390   1st Qu.: 1.115
 Median :3340   Mode  :character   Median : 1.8200   Median : 1.170   Median : 4.320
 Mean   :3607                      Mean   : 2.5160   Mean   : 1.644   Mean   : 5.335
 3rd Qu.:5436                      3rd Qu.: 3.1250   3rd Qu.: 2.160   3rd Qu.: 6.435
 Max.   :9080                      Max.   :34.0200   Max.   :23.800   Max.   :67.850
>
```

Source: Javier Conde (2022)

This clean data set is initially studied through simple exploratory scatterplots (Figure 2.15), histograms (Figure 2.16) and boxplots (Figure 2.17).

Figure 2.15: Exploratory scatterplots with qplot

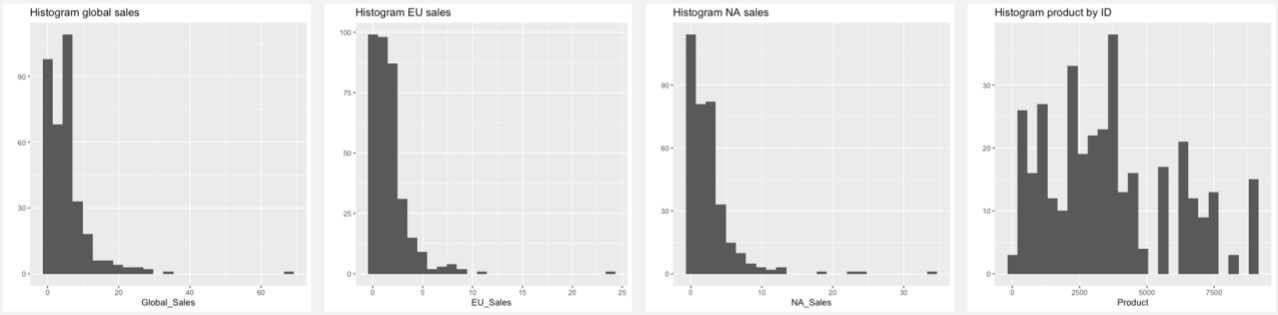
```
83  qplot(Global_Sales, EU_Sales, data=turtle_sales2,
84        main='Scatterplot global sales vs EU sales')
85
86  qplot(Global_Sales, NA_Sales, data=turtle_sales2,
87        main='Scatterplot global sales vs NA sales')
88
89  qplot(EU_Sales, NA_Sales, data=turtle_sales2,
90        main='Scatterplot EU sales vs NA sales')
91
92  qplot(Global_Sales, Product, data=turtle_sales2,
93        main='Scatterplot gobal sales by product')
```



Source: Javier Conde (2022)

Figure 2.16: Exploratory histograms with qplot

```
98  qplot(Global_Sales, bins=25, data=turtle_sales2, main='Histogram global sales')
99  qplot(EU_Sales, bins=25, data=turtle_sales2, main='Histogram EU sales')
100 qplot(NA_Sales, bins=25, data=turtle_sales2, main='Histogram NA sales')
101 qplot(Product, bins=25, data=turtle_sales2, main='Histogram product by ID')
```



Source: Javier Conde (2022)

Figure 2.18: First observations and further exploration

- The 'turtle_sales' data set provided is of great data quality (no duplicates, just two NA in 'Year' column)
- From the initial scatterplot is visible a possible relationship between the variables 'Global_Sales' and 'NA/EU_Sales'
- It needs to be studied further if it is possible building a regression model that is robust enough to allow predictive studies on future global sales depending on the observation of EU and NA values.
- EU, NA and global sales boxplots reveal quite a number of outliers that would recommended to keep monitored and possibly acted upon if further calculations require using min, max, mean, SD, variance, IR Range, etc.
- Histograms reveal possible skewness to the right (positive) of Global, EU, and NA sales. Study on the data set variables' normality and more advanced and detailed plotting is needed.

Source: Javier Conde (2022)

This first approach reveals more study needed on normality with a more detailed plotting package (Figure 2.18). Let's explore these topics.

2.2.2. Data set normality and advanced plotting

Figure 2.19: Data aggregation by product, overview and summary

```
184 turtle_sales_product <- turtle_sales %>% group_by(Product) %>%
185   summarise(across(.cols = c('NA_Sales', 'EU_Sales', 'Global_Sales'), ~sum(.)))
186
187 # View the data frame.
188
189 as_tibble(turtle_sales_product)
190
191 # Summary and View of the new data frame.
192
193 summary(turtle_sales_product)
```

```
> as_tibble(turtle_sales_product)
# A tibble: 175 × 4
   Product NA_Sales EU_Sales Global_Sales
     <int>    <dbl>    <dbl>        <dbl>
 1     107     34.0     23.8         67.8
 2     123     26.6      4.01        37.2
 3     195     13       10.6         29.4
 4     231     12.9      9.03        27.1
 5     249      9.24     7.29        25.7
 6     254     21.5      2.42        29.4
 7     263      9.33     7.57        24.6
 8     283     11.5      7.54        23.8
 9     291     12.0      5.79        23.5
10     326     22.1      0.52        23.2
# … with 165 more rows
# ℹ Use `print(n = ...)` to see more rows
```

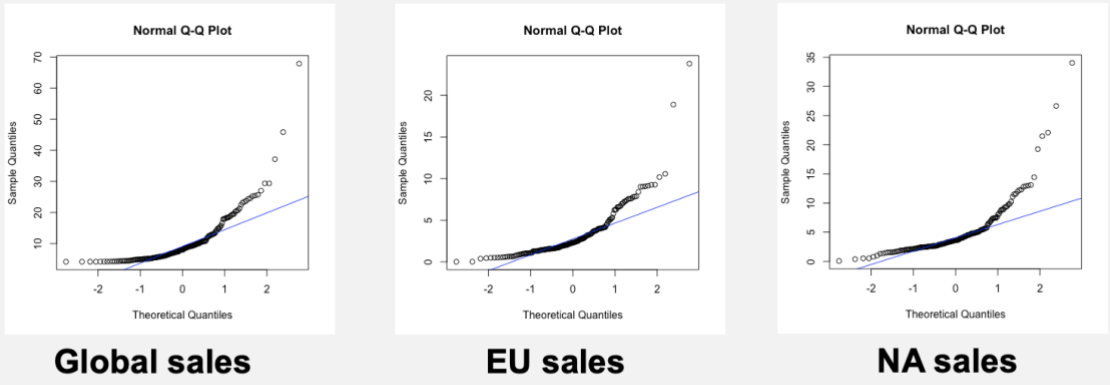
```
> summary(turtle_sales_product)
    Product         NA_Sales         EU_Sales        Global_Sales
 Min.   : 107   Min.   : 0.060   Min.   : 0.000   Min.   : 4.200
 1st Qu.:1468   1st Qu.: 2.495   1st Qu.: 1.460   1st Qu.: 5.515
 Median :3158   Median : 3.610   Median : 2.300   Median : 8.090
 Mean   :3490   Mean   : 5.061   Mean   : 3.306   Mean   :10.730
 3rd Qu.:5442   3rd Qu.: 5.570   3rd Qu.: 4.025   3rd Qu.:12.785
 Max.   :9080   Max.   :34.020   Max.   :23.800   Max.   :67.850
>
```

Source: Javier Conde (2022)

To facilitate the analysis let's aggregate the sales data based on product and check results with as_tibble() function. Result data frame 'turtle_sales_product' (Figure 2.19) is used for analysis on QQ plotting, Shapiro-Wilk testing, skewness and kurtosis, variables correlation, and advanced plotting (Figures 2.20 to 2.25).

Figure 2.20: Normality study: QQ Plots

```
237 qqnorm(turtle_sales_product$Global_Sales)
238 # Add a reference line:
239 qqline(turtle_sales_product$Global_Sales, col='blue')
```



Global sales **EU sales** **NA sales**

Source: Javier Conde (2022)

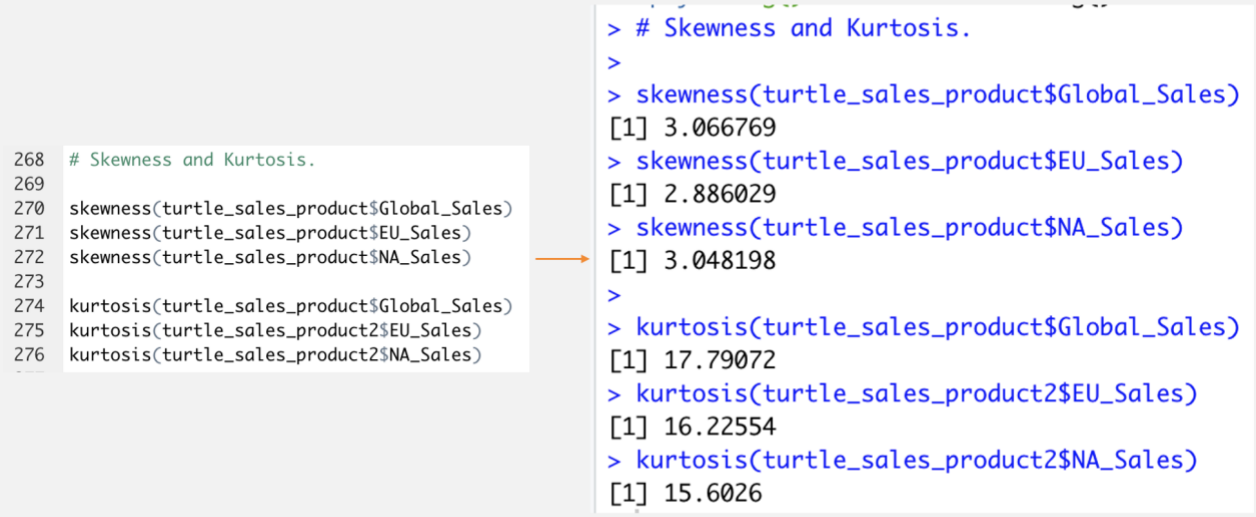
Figure 2.21: Normality study: Shapiro-Wilk test

```
252 # Install and import Moments.
253
254 install.packages('moments')
255 library(moments)
256
257 # Perform Shapiro-Wilk test.
258
259 shapiro.test((turtle_sales_product$Global_Sales))
260 shapiro.test((turtle_sales_product$EU_Sales))
261 shapiro.test((turtle_sales_product$NA_Sales))
```

```
> shapiro.test((turtle_sales_product$Global_Sales))

        Shapiro-Wilk normality test

data:  (turtle_sales_product$Global_Sales)
W = 0.70955, p-value < 2.2e-16

> shapiro.test((turtle_sales_product$EU_Sales))

        Shapiro-Wilk normality test

data:  (turtle_sales_product$EU_Sales)
W = 0.74058, p-value = 2.987e-16

> shapiro.test((turtle_sales_product$NA_Sales))

        Shapiro-Wilk normality test

data:  (turtle_sales_product$NA_Sales)
W = 0.69813, p-value < 2.2e-16
```

Source: Javier Conde (2022)

Figure 2.22: Normality study: skewness and kurtosis



```
268 # Skewness and Kurtosis.
269
270 skewness(turtle_sales_product$Global_Sales)
271 skewness(turtle_sales_product$EU_Sales)
272 skewness(turtle_sales_product$NA_Sales)
273
274 kurtosis(turtle_sales_product$Global_Sales)
275 kurtosis(turtle_sales_product2$EU_Sales)
276 kurtosis(turtle_sales_product2$NA_Sales)
```

```
> # Skewness and Kurtosis.
>
> skewness(turtle_sales_product$Global_Sales)
[1] 3.066769
> skewness(turtle_sales_product$EU_Sales)
[1] 2.886029
> skewness(turtle_sales_product$NA_Sales)
[1] 3.048198
>
> kurtosis(turtle_sales_product$Global_Sales)
[1] 17.79072
> kurtosis(turtle_sales_product2$EU_Sales)
[1] 16.22554
> kurtosis(turtle_sales_product2$NA_Sales)
[1] 15.6026
```

Source: Javier Conde (2022)

Figure 2.23: Variables correlation

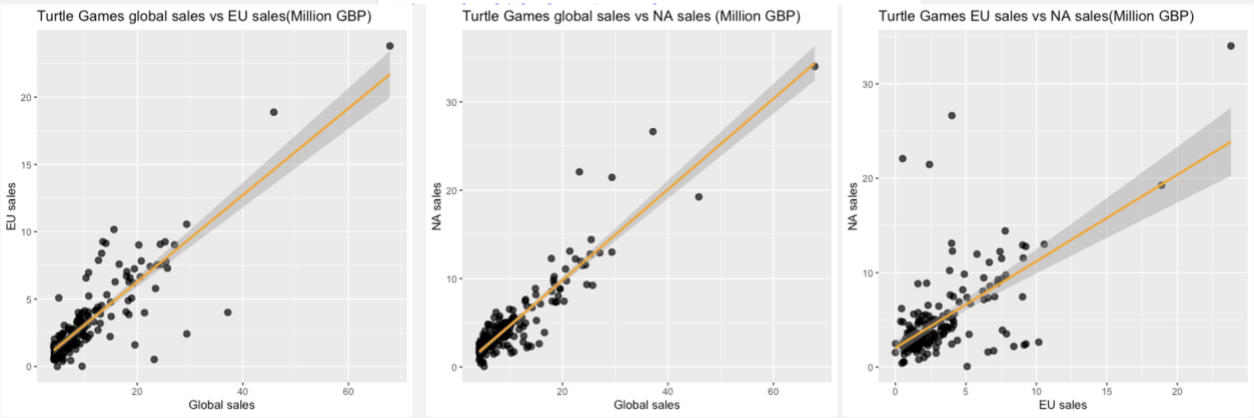
```
287  # Correlation between the sales data columns.
288
289  round(cor(turtle_sales_product), digits=2)
```

```
> round(cor(turtle_sales_product), digits=2)
             Product NA_Sales EU_Sales Global_Sales
Product         1.00    -0.54    -0.45        -0.61
NA_Sales       -0.54     1.00     0.62         0.92
EU_Sales       -0.45     0.62     1.00         0.85
Global_Sales   -0.61     0.92     0.85         1.00
```

Source: Javier Conde (2022)

Figure 2.24: Advanced plotting (ggplot with linear approach)

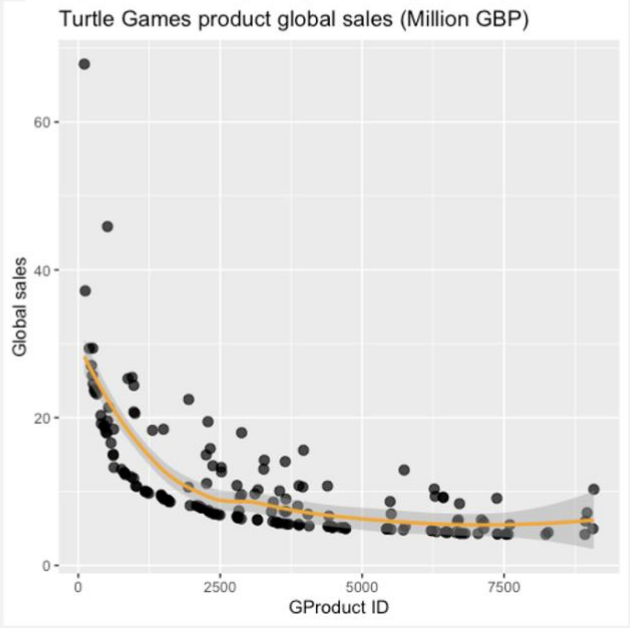
```
> ggplot(data=turtle_sales_product,mapping=aes(x=Global_Sales, y=NA_Sales)) +
+   geom_point(color='black',
+              alpha=0.75,
+              size=2.5) +
+   geom_smooth(method='lm', color='orange') +
+   scale_x_continuous("Global sales") +
+   scale_y_continuous("North America sales") +
+   labs(title="Turtle Games global sales vs North America sales (Million GBP)")
`geom_smooth()` using formula 'y ~ x'
```



Source: Javier Conde (2022)

Figure 2.25: Advanced plotting (ggplot with non linear approach)

```
ggplot(data=turtle_sales_product,mapping=aes(x=Product, y=Global_Sales)) +
  geom_point(color='black',
             alpha=0.75,
             size=2.5) +
  geom_smooth(color='orange') +
  scale_x_continuous("GProduct ID") +
  scale_y_continuous("Global sales") +
  labs(title="Turtle Games product global sales (Million GBP)")
```



Source: Javier Conde (2022)

Figure 2.26: Further findings

- Grouping data based on product allows exploring easily sales based in product ID
- Exploring data set normality through Q-Q Plots, Shapiro-Wilk testing, skewness and kurtosis investigation, variables correlation, and advanced plotting using ggplot:
    - p-values for sales are way below 0.05 (global 2.2e-16, EU 2.987e-16, NA 2.2e-16)
    - all three sales variables are skewed to the right/positively skewed (global 3.06, EU 2.88, NA 3.04)
    - Kurtosis values are way above 3-4 (global 17.7, EU 16.2, NA 15.6) , all a strong departure from normality
- It can be concluded that the sales variables, at least per the sample data provided, are not normally distributed
- There is a strong correlation between Global_sales and NA_sales (0.92), Global_sales an EU_sales (0.85). Less good correlation exists between EU_sales and NA_sales (0.62)
- Advance plotting with ggplot confirms the strong correlation between sales variables. This opens the possibility of building a multiple linear regression model to predict global sales values

Source: Javier Conde (2022)

The findings (Figure 2.26) lead the project to conclusion evaluating how robust a potential Multiple Linear Regression model (MLR) predicting global sales based in NA and EU sales could be.

2.2.3. MLR model and value prediction

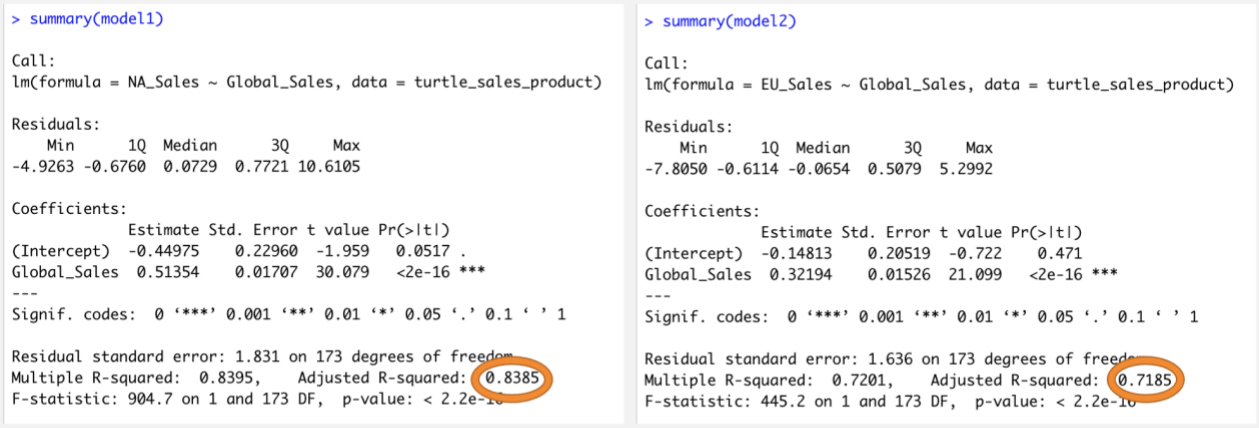
Once analysed correlation strength between all three variables through simple crossed modelling (Figures 2.27, 2.28), we use EU and NA sales to create a MLR model (modelA, Figure 2.31) and an alternative including variable ‘product’ (modelB, Figure 2.32) for further study. We complete our study testing modelA’s accuracy on 5 aleatory observed values (Figures 2.33 to 2.35).

Figure 2.27: Simple linear regression model for sales variables (I)

```
383  # Create a linear regression model
384
385  model1 <- lm(NA_Sales ~ Global_Sales, data=turtle_sales_product)
386  model2 <- lm(EU_Sales ~ Global_Sales, data=turtle_sales_product)
387  model3 <- lm(EU_Sales ~ NA_Sales, data=turtle_sales_product)
388  model4 <- lm(NA_Sales ~ EU_Sales, data=turtle_sales_product)
389
390  # View the model.
391  model1
392  summary(model1)
393  model2
394  summary(model2)
395  model3
396  summary(model3)
397  model4
398  summary(model4)
```

Source: Javier Conde (2022)

Figure 2.28: Simple linear regression model for sales variables (II)

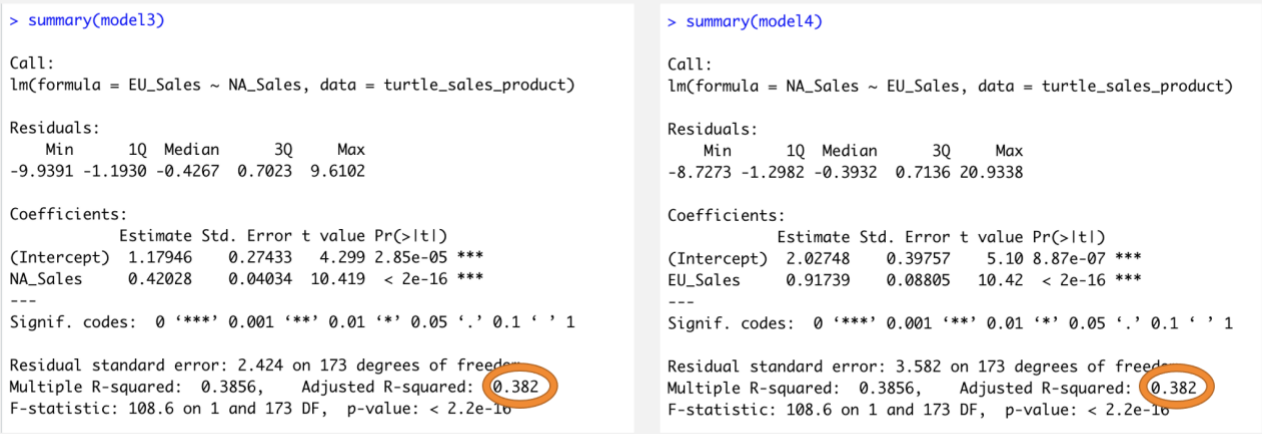


```
> summary(model1)

Call:
lm(formula = NA_Sales ~ Global_Sales, data = turtle_sales_product)

Residuals:
    Min      1Q  Median      3Q     Max
-4.9263 -0.6760  0.0729  0.7721 10.6105

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  -0.44975    0.22960  -1.959   0.0517 .
Global_Sales  0.51354    0.01707  30.079   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.831 on 173 degrees of freedom
Multiple R-squared:  0.8395,    Adjusted R-squared:  0.8385
F-statistic: 904.7 on 1 and 173 DF,  p-value: < 2.2e-16
```

```
> summary(model2)

Call:
lm(formula = EU_Sales ~ Global_Sales, data = turtle_sales_product)

Residuals:
    Min      1Q  Median      3Q     Max
-7.8050 -0.6114 -0.0654  0.5079  5.2992

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  -0.14813    0.20519  -0.722    0.471
Global_Sales  0.32194    0.01526  21.099   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.636 on 173 degrees of freedom
Multiple R-squared:  0.7201,    Adjusted R-squared:  0.7185
F-statistic: 445.2 on 1 and 173 DF,  p-value: < 2.2e-16
```

Source: Javier Conde (2022)

Figure 2.29: Simple linear regression model for sales variables (III)



```
> summary(model3)

Call:
lm(formula = EU_Sales ~ NA_Sales, data = turtle_sales_product)

Residuals:
    Min      1Q  Median      3Q     Max
-9.9391 -1.1930 -0.4267  0.7023  9.6102

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.17946    0.27433   4.299 2.85e-05 ***
NA_Sales     0.42028    0.04034  10.419  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.424 on 173 degrees of freedom
Multiple R-squared:  0.3856,    Adjusted R-squared:  0.382
F-statistic: 108.6 on 1 and 173 DF,  p-value: < 2.2e-16
```

```
> summary(model4)

Call:
lm(formula = NA_Sales ~ EU_Sales, data = turtle_sales_product)

Residuals:
    Min      1Q  Median      3Q     Max
-8.7273 -1.2982 -0.3932  0.7136 20.9338

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.02748    0.39757    5.10 8.87e-07 ***
EU_Sales     0.91739    0.08805   10.42  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.582 on 173 degrees of freedom
Multiple R-squared:  0.3856,    Adjusted R-squared:  0.382
F-statistic: 108.6 on 1 and 173 DF,  p-value: < 2.2e-16
```

Source: Javier Conde (2022)

Figure 2.30: Subset sales data frame without the product column

```
# Select only numeric columns.

names(turtle_sales_product)
turtle_sales_noproduct <- subset(turtle_sales_product, select=-c(Product))

str(turtle_sales_noproduct)
summary(turtle_sales_noproduct)
```

Source: Javier Conde (2022)

Figure 2.31: Multiple linear regression model (sales)

```
> modelA = lm(Global_Sales~NA_Sales+EU_Sales, data=turtle_sales_noproduct)
> summary(modelA)

Call:
lm(formula = Global_Sales ~ NA_Sales + EU_Sales, data = turtle_sales_noproduct)

Residuals:
    Min      1Q  Median      3Q     Max
-3.4156 -1.0112 -0.3344  0.6516  6.6163

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.04242    0.17736   5.877 2.11e-08 ***
NA_Sales     1.13040    0.03162  35.745  < 2e-16 ***
EU_Sales     1.19992    0.04672  25.682  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.49 on 172 degrees of freedom
Multiple R-squared:  0.9668,    Adjusted R-squared:  0.9664
F-statistic:  2504 on 2 and 172 DF,  p-value: < 2.2e-16
```

Source: Javier Conde (2022)

Figure 2.32: Multiple linear regression model (sales and product ID)

```
> modelB = lm(Global_Sales~NA_Sales+EU_Sales+Product, data=turtle_sales_product)
> summary(modelB)

Call:
lm(formula = Global_Sales ~ NA_Sales + EU_Sales + Product, data = turtle_sales_product)

Residuals:
    Min      1Q  Median      3Q     Max
-3.3388 -0.9149 -0.2399  0.7364  5.9643

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.451e+00  3.167e-01   7.741 8.24e-13 ***
NA_Sales     1.068e+00  3.179e-02  33.601  < 2e-16 ***
EU_Sales     1.160e+00  4.421e-02  26.233  < 2e-16 ***
Product     -2.753e-04  5.278e-05  -5.215 5.26e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.388 on 171 degrees of freedom
Multiple R-squared:  0.9714,    Adjusted R-squared:  0.9709
F-statistic:  1933 on 3 and 171 DF,  p-value: < 2.2e-16
```

Source: Javier Conde (2022)

Figure 2.33: Value prediction (model A, sales) (I)

```
457 # A. NA_Sales_sum of 34.02 and EU_Sales_sum of 23.80
458
459 NA_Sales <- c(34.02)
460 EU_Sales <- c(23.80)
461
462 sales1 <- data.frame(NA_Sales, EU_Sales)
463
464 # Predicted Global_Sales value
465 predict(modelA, newdata = sales1)
466
467 # Predicted value 68.056 vs observation value 67.85: good
```

```
469 # B. NA_Sales_sum of 3.93 and EU_Sales_sum of 1.56.
470 # Values not on provided data set
471 # Most similar 3.94/1.28 with observed Global_sales value 8.36
472
473 NA_Sales <- c(3.94)
474 EU_Sales <- c(1.28)
475
476 sales2 <- data.frame(NA_Sales, EU_Sales)
477
478 # Predicted Global_Sales value
479 predict(modelA, newdata = sales2)
480
481 # Predicted value 7.03 vs observation value 8.36: average
```

Source: Javier Conde (2022)

Figure 2.34: Value prediction (model A, sales)(II)

```
# C. NA_Sales_sum of 2.73 and EU_Sales_sum of 0.65, observed value 4.32

NA_Sales <- c(2.73)
EU_Sales <- c(0.65)

sales3 <- data.frame(NA_Sales, EU_Sales)

# Predicted Global_Sales value
predict(modelA, newdata = sales3)

# Predicted value 4.90 vs observation value 4.32 good
```

```
# D. NA_Sales_sum of 2.26 and EU_Sales_sum of 0.97.
# Values not on provided data set
# Most similar 2.27/2.30 with observed Global_sales value 5.60

NA_Sales <- c(2.27)
EU_Sales <- c(2.30)

sales4 <- data.frame(NA_Sales, EU_Sales)

# Predicted Global_Sales value
predict(modelA, newdata = sales4)

# Predicted value 6.36 vs observation value 5.60 average
```

Source: Javier Conde (2022)

Figure 2.35: Value prediction (model A, sales)(III)

```
509  # E. NA_Sales_sum of 22.08 and EU_Sales_sum of 0.52, Global sales 23.21
510
511  NA_Sales <- c(22.08)
512  EU_Sales <- c(0.52)
513
514  sales <- data.frame(NA_Sales, EU_Sales)
515
516  # Predicted Global_Sales value
517  predict(modelA, newdata = sales)
518
519  # Predicted value 26.62 vs observation value 23.21: average
```

Source: Javier Conde (2022)

With the analysis now fully completed, it is time to review insights and recommendations to the business with all the answers from the key initial questions.

Chapter 3

Insights and recommendations

Insights: Figures 3.1, 3.2.

Figure 3.1: Insights on customer reviews

Customer reviews

- Loyalty has a slight correlation with spending and remunerations score. This is advised to be studied further.

- Loyalty has almost no correlation with customer's age

- K-means clustering analysis reveals 5 potential groups marketing department could consider:

  - low income/low spending
  - low income/higher spending
  - average income/average spending
  - higher income/low spending
  - higher income/higher spending

- The 5 most used words in the reviews are 'stars' (427) 'five' (342), 'game'(319), 'great' (295) and 'fun'(218), suggesting a positive sentiment confirmed by the polarity analysis (Mean +0.2 to +0.25)

Source: Javier Conde (2022)

Figure 3.2: Insights on Turtle Games sales

Turtle Games sales

- The 'turtle_sales' data set provided is of great data quality (no duplicates, just two NA in 'Year' column). Outliers are not eliminated and they need to be closely monitored in further calculations

- From the initial scatterplot is visible a possible relationship between the variables 'Global_Sales' and 'NA/EU_Sales' robust enough to allow predictive studies on future global sales through a Multiple Linear Regression model with EU and NA sales as independent variables (modelA)

- Data sets provided on sales are far from normality, very relevant for further study

- Building a second model (modelB) including the variable "Product' for further exploration could be considered, as it adds some robustness (but also complexity) to modelA.

- Tested on 5 aleatory observations, predictive accuracy of modelA results average to good

Source: Javier Conde (2022)

Recommendations: Figures 3.3 to 3.5.

Figure 3.3: Recommendations to the marketing department

- Consider the 5 groups uncovered during the analysis to investigate further other possibly useful relationships (suggested to start with loyalty/gender, loyalty/education, loyalty/product (any product customers keep coming back for?)
- Study further the products associated to great reviews where these words appear. Are there any products with great reviews that haven't been given the marketing exposure?
- Study further also products associated to more negative reviews, data may be available here to understand lack of sales/interest and how to amend it
- Due to the good quality of the data provided, if budget and resources allow consider expanding the study to a bigger sample for more in-depth insights, maybe considering other social media (Instagram, Twitter)
- Ensure communication with other departments is fluid, this may be key to shift projects' priorities (i.e. understanding best selling products per region, provided by the sales department) (Figure 3.4)

Source: Javier Conde (2022)

Figure 3.4: Turtle Games best sellers

Product	NA_Sales	EU_Sales	Global_Sales
107	34.02	23.80	67.85
515	19.25	18.88	45.86
123	26.64	4.01	37.16
254	21.46	2.42	29.39
195	13.00	10.56	29.37
231	12.92	9.03	27.06
249	9.24	7.29	25.72
948	14.42	7.79	25.45

Product	NA_Sales	EU_Sales	Global_Sales
107	34.02	23.80	67.85
515	19.25	18.88	45.86
195	13.00	10.56	29.37
3967	2.63	10.17	15.59
2371	2.44	9.26	13.49
876	12.77	9.25	25.28
3645	2.33	9.14	14.06
979	11.55	9.07	24.36

Product	NA_Sales	EU_Sales	Global_Sales
107	34.02	23.80	67.85
123	26.64	4.01	37.16
326	22.08	0.52	23.21
254	21.46	2.42	29.39
515	19.25	18.88	45.86
948	14.42	7.79	25.45
535	13.11	3.99	21.38
195	13.00	10.56	29.37

Source: Javier Conde (2022)

Figure 3.5: Recommendations to the sales department

- Consider investing budget and resources in further study on MLR models, maybe including variables like product or game genre, with an extended data set to improve accuracy. This may also improve normality in the data set

- Consider comparing sales data from various years and stablish a 3/5 year time-series study with the sales evolution of products/game genres/platforms per region

- Communicate with other departments (i.e. marketing) sales figures to develop a joint strategy on how to promote products that could be potential hits in the future. Consider creating an interactive dashboard (i.e. Tableau, interactive visualisations in RStudio) for other departments to have access to sales information in real time

Source: Javier Conde (2022)

List of Figures