## 1. Background/context of the business:

Turtle Games is a manufacturer and retailer with a global customer base. The company manufactures and sells its own products, along with sourcing and selling products manufactured by other companies. Its product range includes books, board games, video games, and toys.

Turtle Games want to Apply predictive models, advanced data visualisations to understand and predict customer buying patterns, customer background, customer views about products / discounts etc. To employ data driven business strategies like targeted marketing, customer segmentation to give promotional offers, effective manufacturing based on Customer reviews etc. So that they can improve overall sales and profit.

### *1.1 Business Objective:*

Turtle games want to leverage advance data analysis approaches to analyse and convert customer trends into actionable business strategies to improve overall sales performance.

### *1.2 Questions to support business objectives*

- How customers accumulate loyalty points?
- How groups within the customer base can be used to target specific market segments?
- How social data (e.g. customer reviews) can be used to inform marketing campaigns?
- The impact that each product has on sales
- How reliable the data is (e.g., normal distribution, skewness, or kurtosis)
- What the relationship(s) is/are (if any) between North American, European, and global sales?

### *1.3 Additional Questions for client or exploration*

- Does high salary and high spending score together result in more loyalty points?
- Customer classification by educational background can bring some benefits?
- How we can improve customer review and summary capturing so that we get correct inputs and sentiments

## 2. Linear Regression to understand how users accumulate loyalty points.

The marketing department wants to better understand how users accumulate loyalty points. Therefore, it is required to investigate the possible relationships between targeted variable loyalty points and numeric features like age, remuneration, and spending scores.

**<u>Below approach is followed:</u>**

### *2.1 Data ingestion and wrangling*

Imported the necessary libraries (e.g. Pandas and Numpy) for functions and seaborn, matplotlib for visualisation etc.

- Imported csv file into dataframe with proper naming conventions
- Preferred .info () method to explore dataframe as it gives almost full summary of dataframe like colums, rows, null values, datatypes etc. Also Descriptive statistics are explored using describe method.
- Missing values are checked for each column using isna().sum(), there are no missing values
- Cleaned up the dataframe by removing unnecessary columns like language and renamed column with suitable titles

```
# Rename the column headers.
reviews_cleaned = reviews_cleaned.rename(columns={'remuneration (k£)': 'remuneration', \
                                                  'spending_score (1-100)': 'spending_score'})

# View column names.
reviews_cleaned.head()
```

| | 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... |

### *2.2 Predictive analysis and visualisations:*

### *2.2.1 Simple Linear Regression – sklearn*

In this case target variable is continuous, numeric and data to be evaluated is supervised labelled data, hence its best to predict loyalty points using linear regression.

In order to select against which features (X-variables) target value (loyalty points) should be fitted, I used corr() method

| | age | remuneration | spending_score | loyalty_points | product |
|---|---|---|---|---|---|
| **age** | 1.000000 | -0.005708 | -0.224334 | -0.042445 | 0.003081 |
| **remuneration** | -0.005708 | 1.000000 | 0.005612 | 0.616065 | 0.305309 |
| **spending_score** | -0.224334 | 0.005612 | 1.000000 | 0.672310 | -0.001649 |
| **loyalty_points** | -0.042445 | 0.616065 | 0.672310 | 1.000000 | 0.183600 |
| **product** | 0.003081 | 0.305309 | -0.001649 | 0.183600 | 1.000000 |

Looks like Renumeration and spending score has strong positive co-relation with loyalty points. (Close to 1)

Hence decided to run linear regression using sklearn and OLS (Ordinary least square) with loyalty points as feature and Renumeration or spending score as target value

1. Defined independent (Renumeration or spending score) and dependent variables (Loyalty_points)
2. Then Split the data into training (80%) and testing (20%) subsets. More percentage allocated to training data as its required to draw line of best fit and test data is just to check how model performs
3. Reshaped data using array.reshape(-1, 1) as data has a single feature
4. Used linear regression to evaluate possible linear relationships between loyalty points and renumeration/spending score and fitted the model to the training data.
5. Employed the predict method to predict loyalty points based on the x_test dataset
6. Created a scatterplot with regression line with predicted values (line represent predicted values and dots as observations)
7. Observed R2, intercept, and coefficient values

### 2.2.2 *Linear Regression-OLS Method*

Since Sample size is 2000 which is more hence better to explore OLS method for regression between renumeration vs loyalty

OLS and Sklearn Regression methods gave similar results

```
print(regr.score(X_train,y_train))
```

```
0.4483889403237179
```

```
regr.coef_
```

```
array([[33.11355993]])
```

```
regr.intercept_
```

```
array([-81.62216311])
```

OLS Regression Results

| | | | |
|---|---|---|---|
| **Dep. Variable:** | y | **R-squared:** | 0.452 |
| **Model:** | OLS | **Adj. R-squared:** | 0.452 |
| **Method:** | Least Squares | **F-statistic:** | 1648. |
| **Date:** | Tue, 20 Dec 2022 | **Prob (F-statistic):** | 2.92e-263 |
| **Time:** | 21:27:06 | **Log-Likelihood:** | -16550. |
| **No. Observations:** | 2000 | **AIC:** | 3.310e+04 |
| **Df Residuals:** | 1998 | **BIC:** | 3.312e+04 |
| **Df Model:** | 1 | | |
| **Covariance Type:** | nonrobust | | |

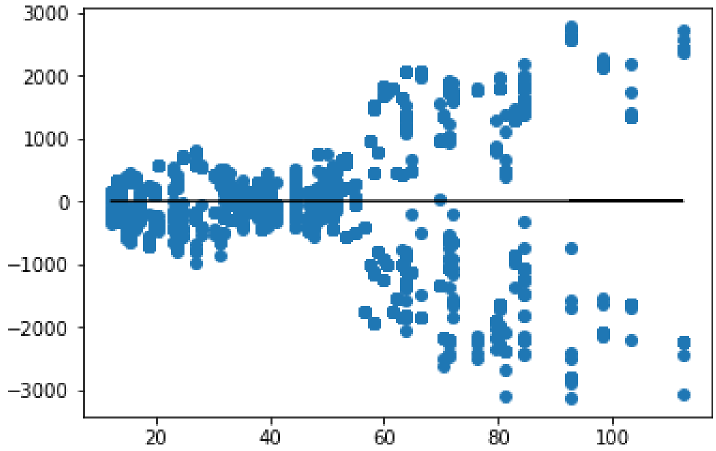
| | coef | std err | t | P>\|t\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| **Intercept** | -75.0527 | 45.931 | -1.634 | 0.102 | -165.129 | 15.024 |
| **X** | 33.0617 | 0.814 | 40.595 | 0.000 | 31.464 | 34.659 |

| | | | |
|---|---|---|---|
| **Omnibus:** | 126.554 | **Durbin-Watson:** | 1.191 |
| **Prob(Omnibus):** | 0.000 | **Jarque-Bera (JB):** | 260.528 |
| **Skew:** | 0.422 | **Prob(JB):** | 2.67e-57 |
| **Kurtosis:** | 4.554 | **Cond. No.** | 122. |

### 2.2.3 *Observations and Insights:*

1. Plotted residuals to check whether there is pattern, this depicted normal behaviour as there are no patterns observed

In [130]:
```python
# Plot residuals ( y- predict - y-observe) versus the x-values
plt.scatter(X, test.predict()-y)

# Plot the regression line (in black).
plt.plot(X, y - y, color='black')

# View the plot.
plt.show()
```




2. Just to cross check model results I used mean value of X variable i.e. spending score in linear equation y = mx+C

```python
# Set the X to spending score mean value 50
y_pred_1 = (-75.052663) + 33.061693 * 50

# View the output
y_pred_1
```

```
1578.031987
```

Which is same as Loyalty mean hence model predictions appear to be correct

| | age | remuneration (k£) | spending_score (1-100) | loyalty_points | product |
|---|---|---|---|---|---|
| count | 2000.000000 | 2000.000000 | 2000.000000 | 2000.000000 | 2000.000000 |
| mean | 39.495000 | 48.079060 | 50.000000 | 1578.032000 | 4320.521500 |
| std | 13.573212 | 23.123984 | 26.094702 | 1283.239705 | 3148.938839 |
| min | 17.000000 | 12.300000 | 1.000000 | 25.000000 | 107.000000 |
| 25% | 29.000000 | 30.340000 | 32.000000 | 772.000000 | 1589.250000 |
| 50% | 38.000000 | 47.150000 | 50.000000 | 1276.000000 | 3624.000000 |
| 75% | 49.000000 | 63.960000 | 73.000000 | 1751.250000 | 6654.000000 |
| max | 72.000000 | 112.340000 | 99.000000 | 6847.000000 | 11086.000000 |

3. Loyalty Vs spending OLS Regression Insights

]:

OLS Regression Results

| | | | |
|---|---|---|---|
| Dep. Variable: | y | R-squared: | 0.452 |
| Model: | OLS | Adj. R-squared: | 0.452 |
| Method: | Least Squares | F-statistic: | 1648. |
| Date: | Tue, 20 Dec 2022 | Prob (F-statistic): | 2.92e-263 |
| Time: | 21:27:06 | Log-Likelihood: | -16550. |
| No. Observations: | 2000 | AIC: | 3.310e+04 |
| Df Residuals: | 1998 | BIC: | 3.312e+04 |
| Df Model: | 1 | | |
| Covariance Type: | nonrobust | | |

| | coef | std err | t | P>\|t\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| Intercept | -75.0527 | 45.931 | -1.634 | 0.102 | -165.129 | 15.024 |
| X | 33.0617 | 0.814 | 40.595 | 0.000 | 31.464 | 34.659 |

| | | | |
|---|---|---|---|
| Omnibus: | 126.554 | Durbin-Watson: | 1.191 |
| Prob(Omnibus): | 0.000 | Jarque-Bera (JB): | 260.528 |
| Skew: | 0.422 | Prob(JB): | 2.67e-57 |
| Kurtosis: | 4.554 | Cond. No. | 122. |

What does the summary indicates

- **R-squared**: 45% of the total variability of y (loyalty points), is explained by the variability of X (spending) which is not highly accurate

- **F-stat:** If the probability of F stat. is smaller than a threshold (usually 0.05), the set of variables of the regression model are significant, else, the regression is not good. This is the p-value – the measure of the probability that the observed difference could have happened by chance. The lower the p-value, the greater the statistical significance.

In this example, the p-value is 2.92e-263, and thus less than 0.05. Therefore, the set of variables of the regression model are significant, hence greater the statistical significance.

- **X: The coefficient of X** describes the slope of the regression line, in other words, how much the response variable y change when X changes by 1 unit.

  Here, if the spending (X) changes by 1 unit (please check units used) the loyalty (y) will change by 33.0617 units.

- The t-value tests the hypothesis that the slope is significant or not. If the corresponding probability is small (typically smaller than 0.05) the slope is significant.

  In this case, the probability of the t-value is zero, thus the estimated slope is significant.

- The last two numbers describe the 95% confidence interval of the true x-coefficient,

  In this case 95% of the samples will derive a slope that is within the interval ( 31.464, 34.659 )

4. Renumeration Vs Loyalty OLS Regression Insights

OLS Regression Results

| | | | |
|---|---|---|---|
| **Dep. Variable:** | y | **R-squared:** | 0.380 |
| **Model:** | OLS | **Adj. R-squared:** | 0.379 |
| **Method:** | Least Squares | **F-statistic:** | 1222. |
| **Date:** | Tue, 20 Dec 2022 | **Prob (F-statistic):** | 2.43e-209 |
| **Time:** | 21:53:05 | **Log-Likelihood:** | -16674. |
| **No. Observations:** | 2000 | **AIC:** | 3.335e+04 |
| **Df Residuals:** | 1998 | **BIC:** | 3.336e+04 |
| **Df Model:** | 1 | | |
| **Covariance Type:** | nonrobust | | |

| | coef | std err | t | P>\|t\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| **Intercept** | -65.6865 | 52.171 | -1.259 | 0.208 | -168.001 | 36.628 |
| **X** | 34.1878 | 0.978 | 34.960 | 0.000 | 32.270 | 36.106 |

| | | | |
|---|---|---|---|
| **Omnibus:** | 21.285 | **Durbin-Watson:** | 3.622 |
| **Prob(Omnibus):** | 0.000 | **Jarque-Bera (JB):** | 31.715 |
| **Skew:** | 0.089 | **Prob(JB):** | 1.30e-07 |
| **Kurtosis:** | 3.590 | **Cond. No.** | 123. |

5. **R-squared:** 38% of the total variability of y (loyalty points), is explained by the variability of X (spending) high variability means high accuracy of model

   Around 38% of the observed variation can be explained by the model's inputs which is not accurate

- **F-stat:** If the probability of F stat. is smaller than a threshold (usually 0.05), the set of variables of the regression model are significant, else, the regression is not good. This is the p-value – the measure of the probability that the observed difference could have happened by chance. The lower the p-value, the greater the statistical significance.

  Here, the p-value is 2.43e-209, and thus less than 0.05.
  Therefore, the set of variables of the regression model are significant, hence greater the statistical significance.

- **X: The coefficient of X** describes the slope of the regression line, in other words, how much the response variable y change when X changes by 1 unit.

Here, if the spending (X) changes by 1 unit (please check units used) the loyalty (y) will change by 34.1878 units.

- **The t-value** tests the hypothesis that the slope is significant or not. If the corresponding probability is small (typically smaller than 0.05) the slope is significant.

  In this case, the probability of the t-value is zero, thus the estimated slope is significant.

- The last two numbers describe the 95% confidence interval of the true x-coefficient,

  In this case 95% of the samples will derive a slope that is within the interval ( 32.27 , 36.106 )

6. Age vs Loyalty are not quite co-related hence not considered for analysis

### 2.2.4 *Multiple linear regression*

Working with multiple variables can give better fit to models, because it relies on more than one feature, and outliers and anomalies can be detected far more effectively. Hence explored multiple regression model with 2 independent variables as simple linear regression gives lower accuracy score. Renumeration and spending score are considered as two independent variables to predict loyalty score.

As part of MLR after setting the variables, fitted regression model and called the predictions for the independent variable. Checked the value of the R2, intercept, and coefficients and below insights determined

### 2.2.5 *Observations and Insights:*

1. R2, intercept, and coefficients

```
         R-squared:  0.826913470198926
         Intercept:  -1700.305097014438
         Coefficients:

Out[136]: [('remuneration', 33.97949882180283), ('spending_score', 32.892694687821006)]
```

- R square is close to 1 explanatory power is strong → 82 percent of variation of loyalty score explained by model
- Intercept---predicted value of loyalty score when dependent values are zero (not valid in this scenario)
- Estimated co-eff independent variables----- 1unit increase in re-numeration then loyalty score will increase by 33.98 and 1 unit increase in spending score then loyalty score will increase by 32.89

2. Some value observations

```
                            OLS Regression Results
==============================================================================
Dep. Variable:         loyalty_points   R-squared:                       0.821
Model:                            OLS   Adj. R-squared:                  0.821
Method:                 Least Squares   F-statistic:                     3665.
Date:                Tue, 20 Dec 2022   Prob (F-statistic):               0.00
Time:                        22:16:44   Log-Likelihood:                -12292.
No. Observations:                1600   AIC:                         2.459e+04
Df Residuals:                    1597   BIC:                         2.461e+04
Df Model:                           2
Covariance Type:            nonrobust
==================================================================================
                     coef    std err          t      P>|t|      [0.025      0.975]
----------------------------------------------------------------------------------
const          -1700.3810     40.400    -42.089      0.000   -1779.623   -1621.138
remuneration      33.6030      0.576     58.322      0.000      32.473      34.733
spending_score    32.9368      0.510     64.595      0.000      31.937      33.937
==============================================================================
Omnibus:                        4.268   Durbin-Watson:                   1.970
Prob(Omnibus):                  0.118   Jarque-Bera (JB):                4.215
Skew:                           0.102   Prob(JB):                        0.122
Kurtosis:                       3.148   Cond. No.                         225.
==============================================================================
```

It indicates smaller standard errors which mean more precise estimates on the basis of 1600 number of training observations.

p>t---- small p values indicate both variables have statistical significance in predictions

95% conf level ----sensitivity of values are between 31.937 and 33.937

3. Multicolinearity may occur if there are strong correlations between two or more independent

multicollinearity causes unreliable coefficient estimates But VIF factor is closer to 1 hence there is no multicollinearity between independent variables

```
   VIF Factor         features
0        9.45            const
1        1.00     remuneration
2        1.00   spending_score
```

4. Model error

Accuracy decreased as the linearity of the data set decreased (Bigger absolute error)

```
Mean Absolute Error (Final): 446.67056349246894
Mean Square Error (Final): 323161.11611347925
```

## 3. Classification and clustering

Want to check whether classification can help to improve sales trend or give some insights

Classification can only be applied to predict the probability of categorical output. As per available data customer can be classified into three educational groups Graduate','Highly-edu','Under-Grad'. If we can build a model to predict customer educational background based on numeric independent variables like spending score, age etc. then this data can be used for promotional strategy like graduate or highly educated group of customers are offered discount on books or video game promotions targeted for under graduates etc.

***3.1 Multinomial logistic regression for classification***

Multinomial logistic regression can predict the probabilities of different possible outcomes of a categorical dependent variable (Y) conditional on a set of independent variables (X's).

Below approach followed:

1.Prepared data frame by dropping nonnumeric columns, which are not helpful for regression

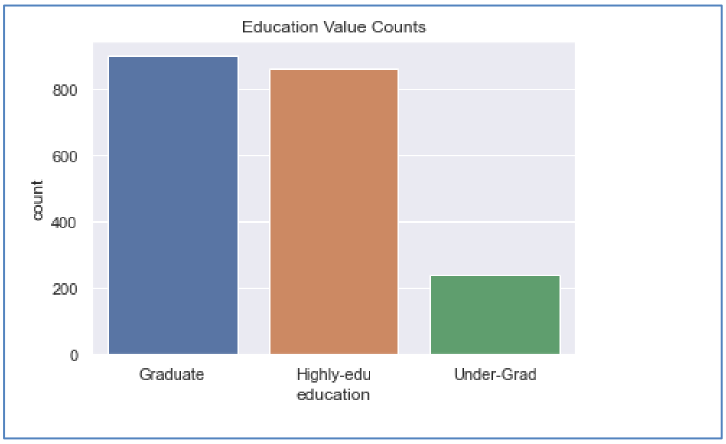
2. Then education values are replaced with three distinct values as

Graduate, Highly-edu (postgraduate, PhD), Under-Grad (diploma, Basic)

3. balanced data set.

Initial data set is not balanced

```
Graduate      900
Highly-edu    860
Under-Grad    240
Name: education, dtype: int64
```




Used SMOTE class to balance out target variable

```
Graduate      629
Highly-edu    629
Under-Grad    629
Name: education, dtype: int64
```

4. Set X and y variables

```
# Set the variables:
X_data = reviews_log.drop('education', axis = 1)
y = reviews_log['education']
```

5. Checked multi collinearity but variance inflation factor is not greater than 30 for any variables hence no need to dropped columns

| | feature | VIF |
|---|---|---|
| 0 | age | 2.554166 |
| 1 | remuneration (k£) | 4.987514 |
| 2 | spending_score (1-100) | 5.555557 |
| 3 | loyalty_points | 8.398247 |

6. Data is normalised using MinMaxScaler

7. Defined the MLR model and set predictions and parameters and fit the model

The parameters are returned with three intercepts and three sets of regression coefficients. The factor level Graduate is the reference level (baseline) for the Education variable. Therefore, it is left out of the output.

8. Confusion matrix is created

Which measures how effective our model is at making positive predictions.

| | predicted_Graduate | predicted_Highly-edu | predicted_Under-Grad |
|---|---|---|---|
| **Graduate** | 148 | 26 | 97 |
| **Highly-edu** | 121 | 38 | 99 |
| **Under-Grad** | 12 | 3 | 56 |

It predicts that 148 of the 271 (sum of the graduate row is 148+26+97), or 54%, were correctly predicted as Graduate.

9.Determined the accuracy of the model

```
Accuracy score: 0.4033333333333333
              precision    recall  f1-score   support

    Graduate       0.53      0.55      0.54       271
  Highly-edu       0.57      0.15      0.23       258
  Under-Grad       0.22      0.79      0.35        71

    accuracy                           0.40       600
   macro avg       0.44      0.49      0.37       600
weighted avg       0.51      0.40      0.38       600
```

### ***3.1.1 Observation and insights***

In this demonstration, correct educational classification is required to support promotions. Keeping this in mind, the accuracy of the model is 40%, which is not very accurate and therefore not useful as a predictive model. It seems that there is a 40% chance of success.

Therefore, if promotional or marketing programmes are very expensive and time-consuming, it might not be the best way to proceed. But if there are no extra budget required then it could be worth exploring promotional strategies as graduate or highly graduate classification has more than 50% of precision.

### ***3.2 Clustering with k-means to identify groups within the customer base that can be used to target specific market segments.***

The marketing department also wants to better understand the usefulness of renumeration and spending scores to identify groups within the customer base that can be used to target specific market segments. Used k-means clustering to identify the optimal number of clusters and then apply and plot the data using the created segments

**Approach:**

Imported the CSV file into data frame and explored data to determine which columns need to be removed, renamed etc.

Explored data and prepared the data for clustering.

Checked descriptive statistics of data frame

```
# Descriptive statistics with categorical values
Clustering_all.describe(include = 'all')
```

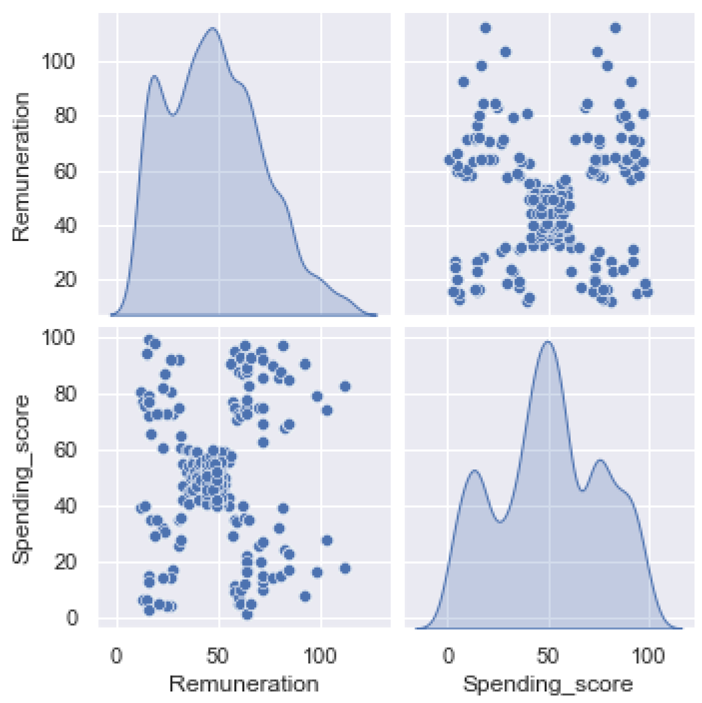
| | gender | age | remuneration (k£) | spending_score (1-100) | loyalty_points | education | language | platform | product | review | summary |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **count** | 2000 | 2000.000000 | 2000.000000 | 2000.000000 | 2000.000000 | 2000 | 2000 | 2000 | 2000.000000 | 2000 | 2000 |
| **unique** | 2 | NaN | NaN | NaN | NaN | 5 | 1 | 1 | NaN | 1980 | 1432 |
| **top** | Female | NaN | NaN | NaN | NaN | graduate | EN | Web | NaN | love it | Five Stars |
| **freq** | 1120 | NaN | NaN | NaN | NaN | 900 | 2000 | 2000 | NaN | 5 | 378 |
| **mean** | NaN | 39.495000 | 48.079060 | 50.000000 | 1578.032000 | NaN | NaN | NaN | 4320.521500 | NaN | NaN |
| **std** | NaN | 13.573212 | 23.123984 | 26.094702 | 1283.239705 | NaN | NaN | NaN | 3148.938839 | NaN | NaN |

Since for clustering we are interested only on numeric features, categorical features need to be dropped. Also, some of the numeric columns like age or product are not making sense and hence need to be removed.

Used drop method to keep only Renumeration and spending score column.

1. Plotted remuneration versus spending score to determine any correlations and possible groups (clusters) using scatterplot and pairplot. It appears to be there are 5 clusters

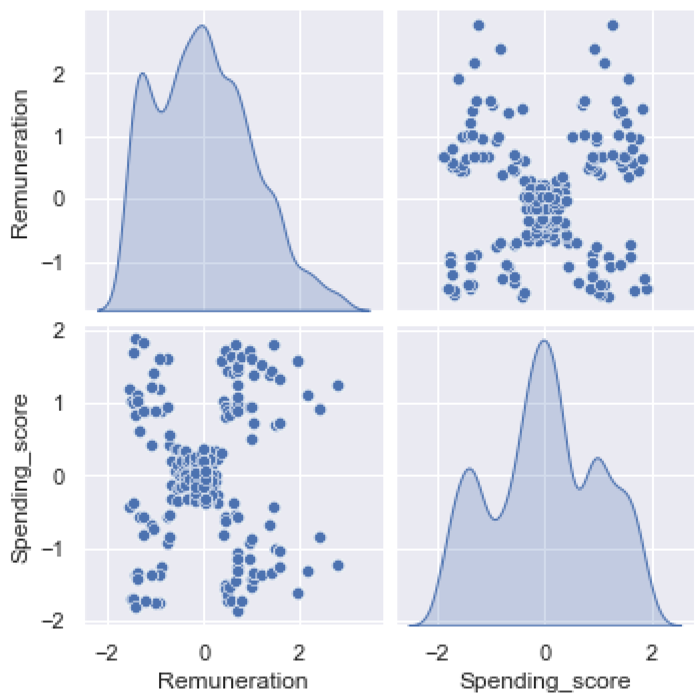
   In case of pair plot middle graphs show distribution. There are 4 clear peaks with respect to Renumeration and spending score



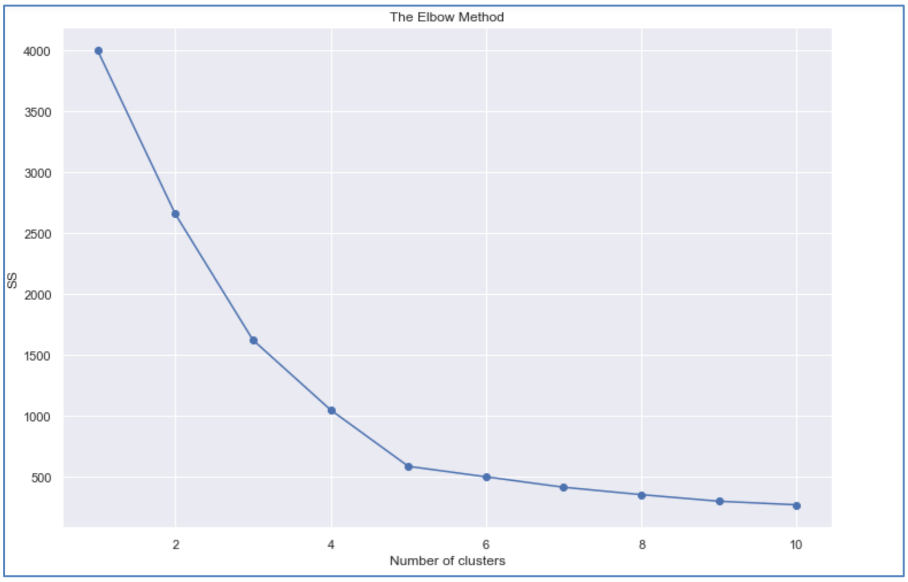

2. Algorithms perform better when numerical input variables are scaled to a standard range hence used StandardScaler function

```
# scaling
from sklearn.preprocessing import StandardScaler
Clustering_df.loc[:,:]= StandardScaler().fit_transform(Clustering_df.loc[:,:])
```

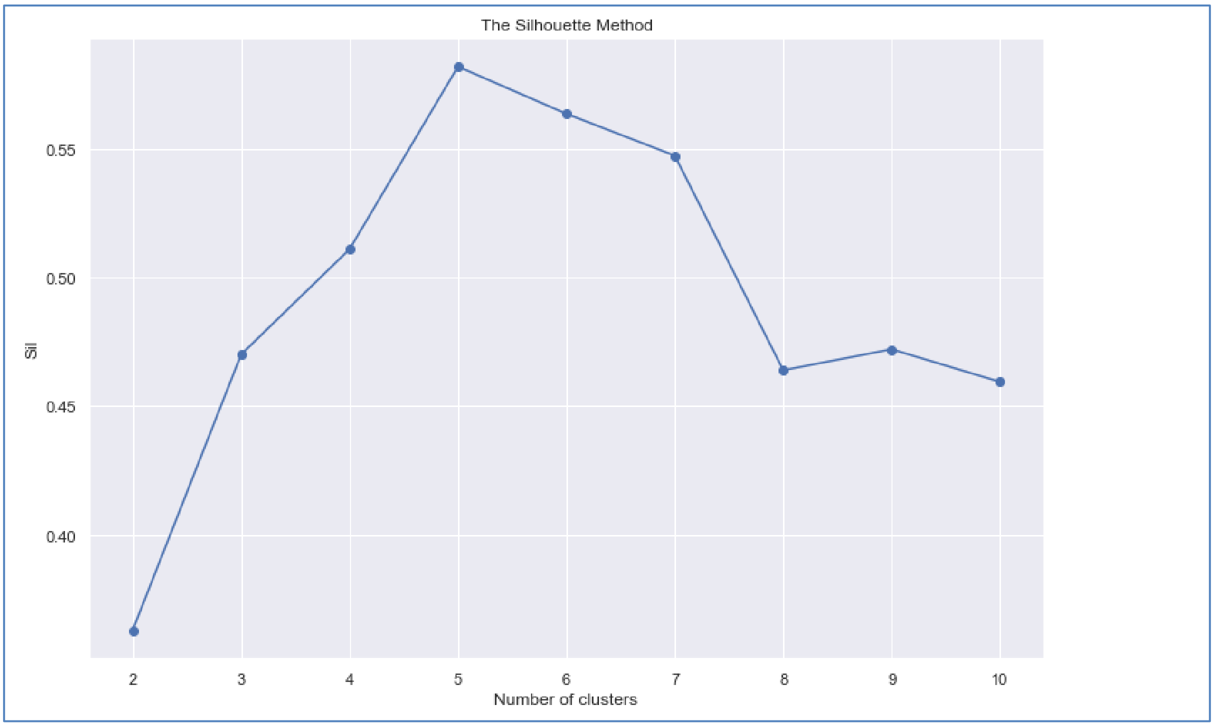
Below is standardised pair plot which again shows 5 clusters



3. Used the Silhouette and Elbow methods to determine the optimal number of clusters for k-means clustering using scaled dataframe.
   Cluster determination has no thumb rule as there are no true labels, fewer clusters mean greater data reduction and greater cluster means more homogeneity between data points hence while selecting clusters both these things need to be cleared out.
   For elbow method sum of square distances increases when clusters are less, as per below diagram SS decreases and graph is quite linear for cluster size 5/6 preferably 5 onwards




   Silhouette score is similarity of object to its assigned cluster and when its nearer to 1 its better, as per below graph cluster value =5 has max value and 6 has below that hence let’s explore 5 and 6 value for clustering

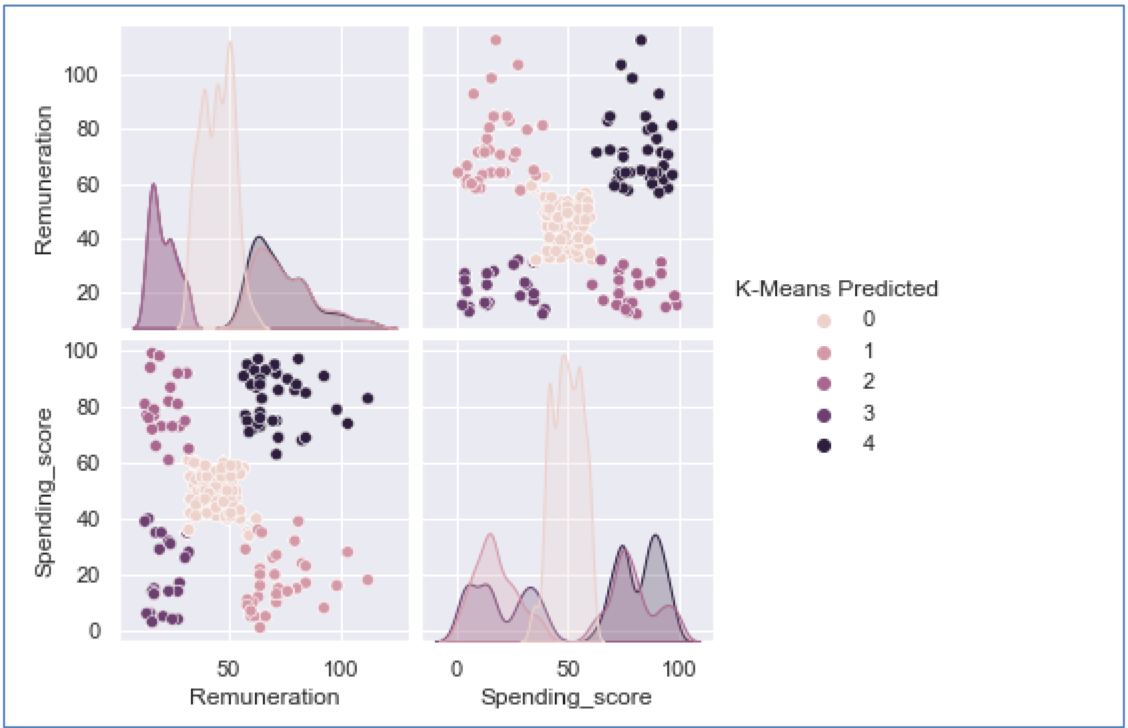


Evaluated k-means model at different values of k 5 and 6
The number of predicted values per class indicates a better distribution for k=5 than k=6

```
0    774
4    356
1    330
3    271
2    269
Name: K-Means Predicted, dtype: int64
```

```
1    767
2    356
0    271
3    269
5    214
4    123
Name: K-Means Predicted, dtype: int64
```

Five clusters look quiet evenly distributed as compared to k=6




Also Means are quite distinguishable and are at the centre of each cluster

```
Mean of Cluster 2 is:
 Remuneration         74.831212
Spending_score        17.424242
K-Means Predicted      1.000000
Name: mean, dtype: float64


Mean of Cluster 3 is:
 Remuneration         20.353680
Spending_score        79.416357
K-Means Predicted      2.000000
Name: mean, dtype: float64


Mean of Cluster 4 is:
 Remuneration         20.424354
Spending_score        19.763838
K-Means Predicted      3.000000
Name: mean, dtype: float64


Mean of Cluster 5 is:
 Remuneration         73.240281
Spending_score        82.008427
K-Means Predicted      4.000000
Name: mean, dtype: float64
```

Middle cluster mean coincides with mean of whole data frame

| | Remuneration | Spending_score |
|---|---|---|
| count | 2000.000000 | 2000.000000 |
| mean | 48.079060 | 50.000000 |
| std | 23.123984 | 26.094702 |
| min | 12.300000 | 1.000000 |
| 25% | 30.340000 | 32.000000 |
| 50% | 47.150000 | 50.000000 |
| 75% | 63.960000 | 73.000000 |
| max | 112.340000 | 99.000000 |

Hence middle cluster is largest for both k=5 and 6

**Outcome:**

5 marketing segments can be determined as The number of predicted values per class indicates a better distribution for k=5 than k=6.

## 4. NLP to explore whether social data (e.g. customer reviews) can be used in marketing campaigns

### *4.1 Approach*

Customer reviews were downloaded from the website of Turtle Games. This data will be used to steer the marketing department on how to approach future campaigns. Therefore, the marketing department want to identify the 15 most common words used in online product reviews. They also want to have a list of the top 20 positive and negative reviews received from the website. This can be achieved by applying NLP on the data set.

1. Loaded required libraries for NLP like word cloud, nltk

2. Then loaded review data into data frame for cleaning and exploration

3. Since NLP works only on free text column, removed all the columns other than review and summary. Also checked for null or missing values

4. Prepared the data for NLP

    A. Change to lower case and join the elements in each of the columns respectively (review and summary).
    B. Replace punctuation in each of the columns respectively (review and summary).
    C. Drop duplicates in both columns (review and summary).

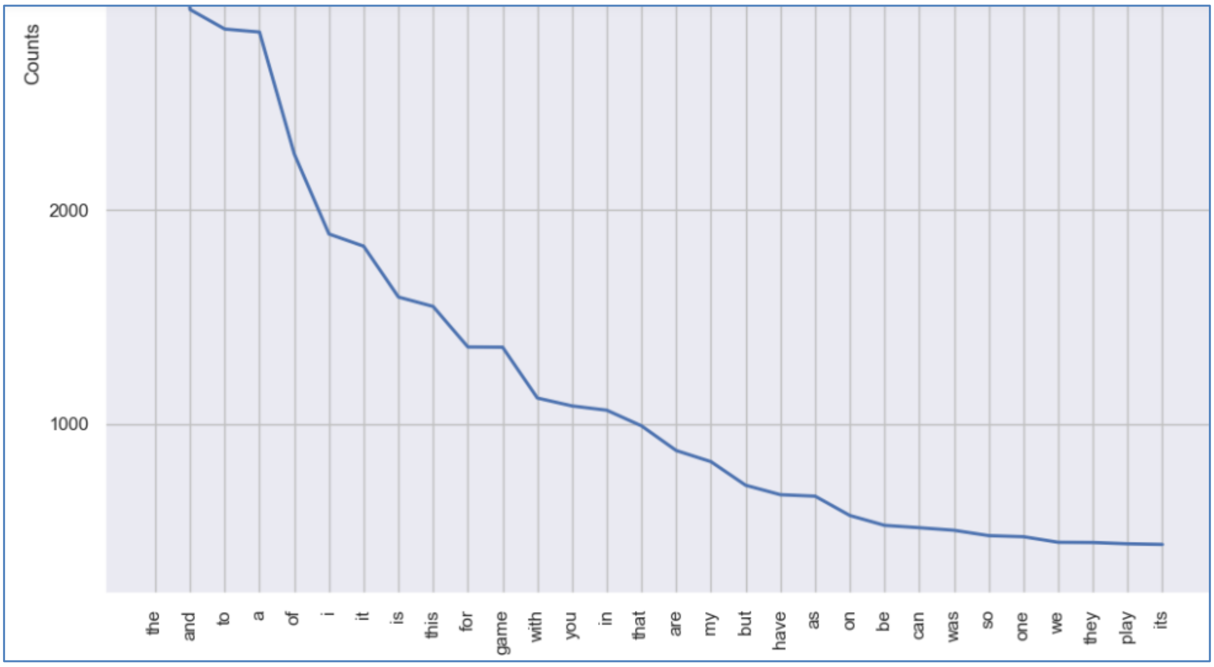
    Summary column had 649 duplicates and review column has 50. This brings Final data set rows from 2000 to 1350

```
: Final_reviews.shape
: (1350, 4)
```

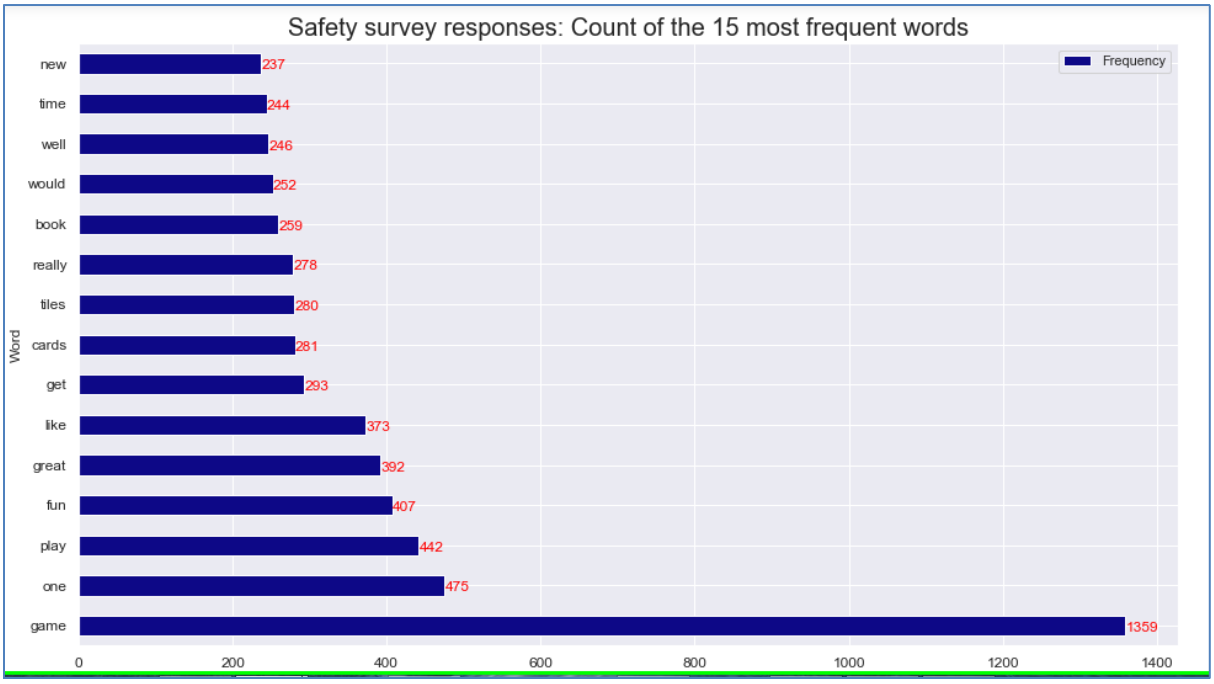
Before tokenisation and before removing stop words checked word cloud just to notice differences, in before diagram I can still see unimportant words which can impact our analysis


5. Tokenised and created wordclouds for the respective columns (separately). Imported nltk and download nltk's resources to assist with tokenisation. Applied word tokenisation and converted it into list

6. Frequency distribution used to check which words are frequently used. For better results, removed alphanumeric characters and stopwords. As before removing stop words results not giving good insights




**<u>Frequency and wordcount observations after removing stop words</u>**

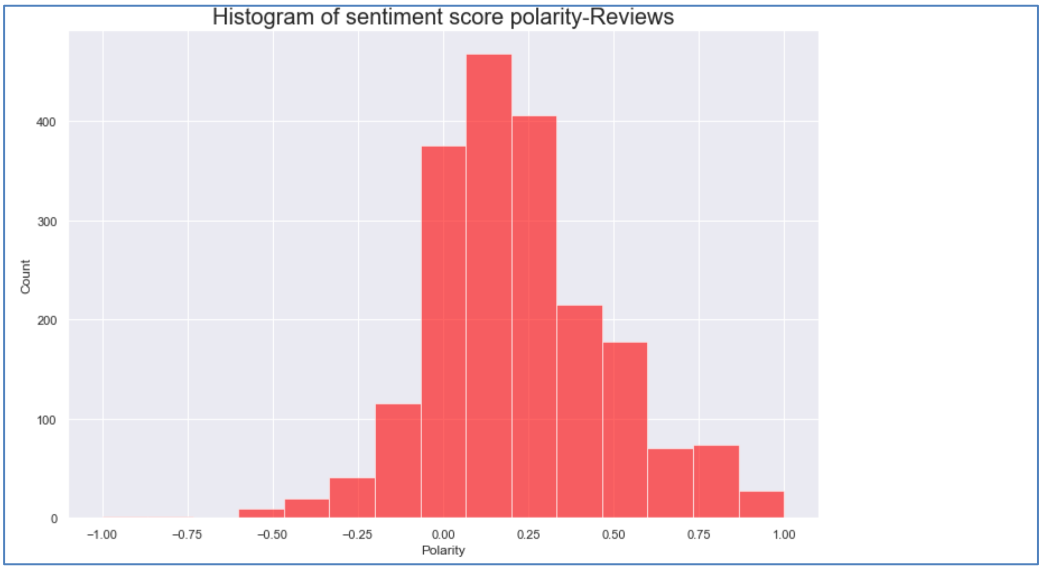



Book, fun, time, new, card, tiles are interesting frequent words to notice here

| Word | Frequency |
|---|---|
| game | 1359 |
| one | 475 |
| play | 442 |
| fun | 407 |
| great | 392 |
| like | 373 |
| get | 293 |
| cards | 281 |
| tiles | 280 |
| really | 278 |
| book | 259 |
| would | 252 |
| well | 246 |
| time | 244 |
| new | 237 |

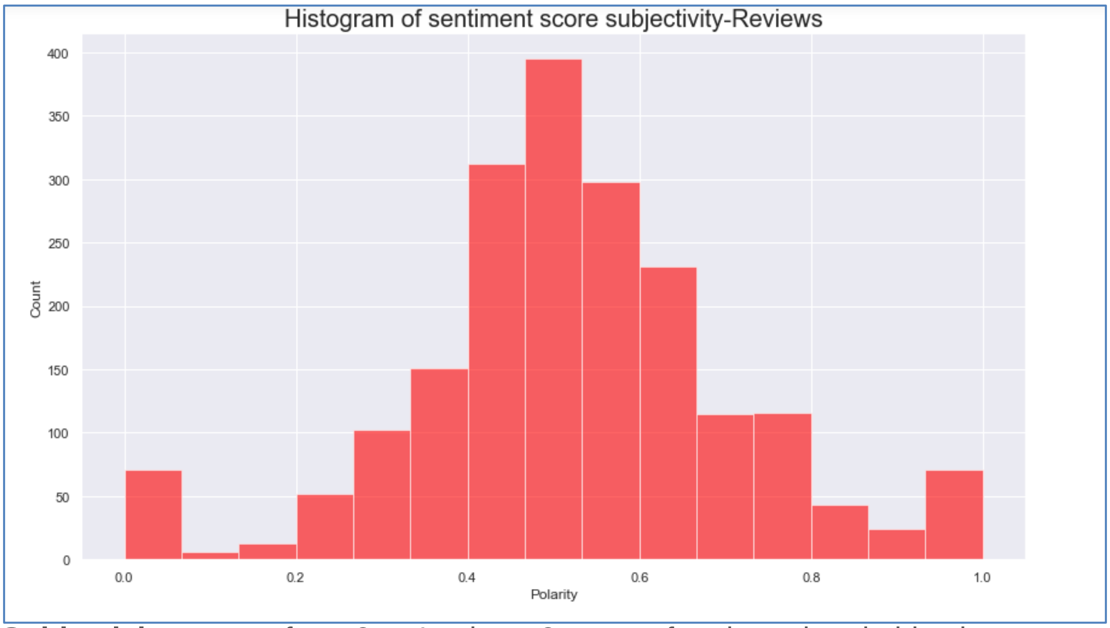
7. Reviewed polarity and sentiment.

Plotted histograms of polarity (15 bins)




Sentiment polarity scores are assigned on a range, where -1 is the lowest negative sentiment, and +1 is the highest possible positive sentiment. Overall Histogram shows positive sentiment as it is left skewed

a. Histogram for the sentiment scores for subjectivity




**Subjectivity** ranges from 0 to 1, where 0 means fact-based and objective while 1 means opinion-based and subjective.

Sentiment histogram is normalised, and has highest value at 0 which indicates reviews are more fact based and objective

8. Identified and print the top 20 positive and negative reviews and summaries respectively.

**Negative Reviews**

| | review | summary | polarity_Rev | subjectivity_rev | polarity_sum | subjectivity_sum |
|---|---|---|---|---|---|---|
| 208 | booo unles you are patient know how to measure i didnt have the patience neither did my daughter boring unless you are a craft person which i am not | boring unless you are a craft person which i am | -1.000000 | 1.000000 | -1.000000 | 1.000000 |
| 182 | incomplete kit very disappointing | incomplete kit | -0.780000 | 0.910000 | 0.000000 | 0.000000 |
| 1804 | im sorry i just find this product to be boring and to be frank juvenile | disappointing | -0.583333 | 0.750000 | -0.600000 | 0.700000 |
| 364 | one of my staff will be using this game soon so i dont know how well it works as yet but after looking at the cards i believe it will be helpful in getting a conversation started regarding anger and what to do to control it | anger control game | -0.550000 | 0.300000 | -0.550000 | 0.300000 |
| 117 | i bought this as a christmas gift for my grandson its a sticker book so how can i go wrong with this gift | stickers | -0.500000 | 0.900000 | 0.000000 | 0.000000 |
| 227 | this was a gift for my daughter i found it difficult to use | two stars | -0.500000 | 1.000000 | 0.000000 | 0.000000 |
| 230 | i found the directions difficult | three stars | -0.500000 | 1.000000 | 0.000000 | 0.000000 |
| 290 | instructions are complicated to follow | two stars | -0.500000 | 1.000000 | 0.000000 | 0.000000 |
| 301 | difficult | three stars | -0.500000 | 1.000000 | 0.000000 | 0.000000 |
| 1524 | expensive for what you get | two stars | -0.500000 | 0.700000 | 0.000000 | 0.000000 |
| 174 | i sent this product to my granddaughter the pompom maker comes in two parts and is supposed to snap together to create the pompoms however both parts were the same making it unusable if you cant make the pompoms the kit is useless since this was sent as a gift i do not have it to return very disappointed | faulty product | -0.491667 | 0.433333 | 0.000000 | 0.000000 |
| 347 | my 8 yearold granddaughter and i were very frustrated and discouraged attempting this craft it is definitely not for a young child i too had difficulty understanding the directions we were very disappointed | frustating | -0.446250 | 0.533750 | 0.000000 | 0.000000 |

There are only 4 reviews which have negative polarity above 0.5

| | | | | | |
|---|---|---|---|---|---|
| 144 | my kids grew up with a peg bench and hammer and loved it but i bought this brand for my grandson and was disappointed the pegs fit too loosely into the bench and he does not even use his hammer to pound them in as he can just push them in with his hand or sometimes they fall through automatically my suggestion is to make the pegs fit a little tighter so the kids can learn skills of coordination etc when pounding them in the pegs are nice and thick for little hands but just not snug enough fitting to really use the toy as it is intended | disappointed | 0.108173 | 0.524519 | -0.750000 |
| 631 | eggs split and were unusable | disappointed | 0.000000 | 0.000000 | -0.750000 |
| 793 | my mom already owned an acquire game but she always commented on how poorly it was made so i thought i would get her a new one for christmas the quality of this one was not much better her old one had cards for each player to see how much each hotel cost to buy according to how many tiles it had this one did not even have that i expected better quality for the price i paid for it it didnt even come with a bag for the tiles i think she was disappointed | disappointed | -0.046364 | 0.450455 | -0.750000 |
| 1620 | i was thinking it was a puppet but it is not it is a doll still worked for what i needed but the only way to get the animals in and out is through the mouth which is a little difficult for a little child | disappointed | -0.218750 | 0.750000 | -0.750000 |
| | i found that this card game does the opposite of what it was intended for it actually | | | | |

Most of the negative summary is related to disappointments that means product has not satisfied expectations or commitments.

## Positive Reviews/Summary:

| | review | summary | polarity_Rev | subjectivity_rev |
|---|---|---|---|---|
| 7 | came in perfect condition | five stars | 1.000000 | 1.000000 |
| 165 | awesome book | five stars | 1.000000 | 1.000000 |
| 194 | awesome gift | five stars | 1.000000 | 1.000000 |
| 496 | excellent activity for teaching selfmanagement skills | five stars | 1.000000 | 1.000000 |
| 524 | perfect just what i ordered | five stars | 1.000000 | 1.000000 |
| 591 | wonderful product | five stars | 1.000000 | 1.000000 |
| 609 | delightful product | five stars | 1.000000 | 1.000000 |
| 621 | wonderful for my grandson to learn the resurrection story | five stars | 1.000000 | 1.000000 |
| 790 | perfect | aquire game | 1.000000 | 1.000000 |
| 933 | awesome | five stars | 1.000000 | 1.000000 |
| 1037 | awesome | five stars | 1.000000 | 1.000000 |
| 1135 | awesome set | five stars | 1.000000 | 1.000000 |
| 1168 | best set buy 2 if you have the means | five stars | 1.000000 | 0.300000 |
| 1177 | awesome addition to my rpg gm system | five stars | 1.000000 | 1.000000 |
| 1301 | its awesome | five stars | 1.000000 | 1.000000 |
| 1401 | one of the best board games i played in along time | five stars | 1.000000 | 0.300000 |
| 1550 | my daughter loves her stickers awesome seller thank you | awesome seller thank you | 1.000000 | 1.000000 |
| 1609 | this was perfect to go with the 7 bean bags i just wish they were not separate orders | five stars | 1.000000 | 1.000000 |

Most of the positive reviews have five stars have positive adjectives like awesome, wonderful, delightful.

| | review | summary | polarity_Rev | subjectivity_rev | p |
|---|---|---|---|---|---|
| 6 | i have bought many gm screens over the years but this one is the best i have ever seen it has all the relevant information i need and no crap filler on it very happy with this screen | best gm screen ever | 0.660000 | 0.700000 | |
| 28 | these are intricate designs for older children and adults this book is full of beautiful designs just waiting to be awakened by your choice of colors great for creativity | wonderful designs | 0.541667 | 0.658333 | |
| 32 | awesome my 8 year olds favorite xmas gift its 915 am xmas morning and hes already colored three of these | perfect | 0.750000 | 1.000000 | |
| 80 | my daughter loves these little books theyre the perfect size to keep in the car or a diaper bag or purse i keep them on hand for times when were stuck waiting in a doctors office or anywhere else | theyre the perfect size to keep in the car or a diaper | 0.406250 | 0.750000 | |
| 134 | this occupied my almost3 year old for nearly an hour stickers were durable and easy to peel afterwards he kept going back to the box to see if there were more robot stickers to assemble in there ill probably drop another dollar and buy it again for his christmas stocking three cheers for the short memory of a preschooler | perfect for preschooler | 0.090476 | 0.461905 | |
| 140 | i bought 8 of these for my 3 year old daughters robot themed birthday party as favors for the little ones and it was a great hit i didnt realize that the stickers were robot parts that the kids assemble themselves to create their own robots that was a lot of fun and for the price it was well worth it | awesome sticker activity for the price | 0.318750 | 0.458333 | |
| 161 | my 8 year old son loves this drawing book loves it | awesome book | 0.100000 | 0.200000 | |
| 163 | this was a christmas present for a nephew who loves to draw and he loves superheroes he was very happy with his gift | he was very happy with his gift | 0.500000 | 0.500000 | |
| 187 | great product took a little practice and time but after you get the hang of it it turns into a cute cuddly little friend mine didnt turn out exactly like the picture but it adds a taste of your own sense of style they are super cute and comes with everything it says it will | awesome | 0.326042 | 0.708333 | |
| 210 | i was skeptical but my 9 year old has had so much fun with this kit and it was her favorite christmas present she pretty much made the puppies herself with minimal help from me though i did hot glue some ears rather than use the included glue only downside is the cuttings can be messy but really wonderful instructions wellmade | awesome and welldesigned for 9 | 0.192222 | 0.593889 | |

### *4.2 insights and observation*

Applied simple sentiment analysis techniques to textual data from survey responses. Used word clouds to analyse key topics in the data set. We also used sentiment analysis to evaluate positive and negative tone in the texts, as well as the subjectivity.

This gave below observations and insights:

- Book, fun, time, new, card, tiles are frequently used words
- Reviews are more fact based and objective and polarity are more positive
- Most of the negative comments are related to disappointments where expectation or commitments are not met.

## 5. What is the impact on sales per product

This is explored using R

- Imported all the required libraries like ('tidyverse') required for the analysis.

```
> head(turtle_sales)
  Ranking Product Platform Year        Genre Publisher NA_Sales EU_Sales
1       1     107      Wii 2006       Sports  Nintendo    34.02    23.80
2       2     123      NES 1985     Platform  Nintendo    23.85     2.94
3       3     195      Wii 2008       Racing  Nintendo    13.00    10.56
4       4     231      Wii 2009       Sports  Nintendo    12.92     9.03
5       5     249       GB 1996 Role-Playing  Nintendo     9.24     7.29
6       6     254       GB 1989       Puzzle  Nintendo    19.02     1.85
  Global_Sales
1        67.85
2        33.00
3        29.37
4        27.06
5        25.72
```

- Loaded and explored the data.

```
# 2. Explore the data set

# Convert data frame to a tibble.
as_tibble(turtle_sales)


# Use the glimpse() function.
glimpse(turtle_sales)


# Use the summary() function.
summary(turtle_sales)
```

- Tibble:

```
> as_tibble(turtle_sales)
# A tibble: 352 x 9
   Ranking Product Platform  Year Genre        Publisher NA_Sales EU_Sales Global_~1
     <int>   <int> <chr>    <dbl> <chr>        <chr>        <dbl>    <dbl>     <dbl>
 1       1     107 Wii       2006 Sports       Nintendo      34.0    23.8       67.8
 2       2     123 NES       1985 Platform     Nintendo      23.8     2.94      33
 3       3     195 Wii       2008 Racing       Nintendo      13      10.6       29.4
 4       4     231 Wii       2009 Sports       Nintendo      12.9     9.03      27.1
 5       5     249 GB        1996 Role-Playing Nintendo       9.24    7.29      25.7
 6       6     254 GB        1989 Puzzle       Nintendo      19.0     1.85      24.8
 7       7     263 DS        2006 Platform     Nintendo       9.33    7.57      24.6
 8       8     283 Wii       2006 Misc         Nintendo      11.5     7.54      23.8
 9       9     291 Wii       2009 Platform     Nintendo      12.0     5.79      23.5
10      10     326 NES       1984 Shooter      Nintendo      22.1     0.52      23.2
# ... with 342 more rows, and abbreviated variable name 1: Global_Sales
```

Glimpse

```
> glimpse(turtle_sales)
Rows: 352
Columns: 9
$ Ranking      <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18~
$ Product      <int> 107, 123, 195, 231, 249, 254, 263, 283, 291, 326, 399, 405, 4~
$ Platform     <chr> "Wii", "NES", "Wii", "Wii", "GB", "GB", "DS", "Wii", "Wii", "~
$ Year         <dbl> 2006, 1985, 2008, 2009, 1996, 1989, 2006, 2006, 2009, 1984, 2~
$ Genre        <chr> "Sports", "Platform", "Racing", "Sports", "Role-Playing", "Pu~
$ Publisher    <chr> "Nintendo", "Nintendo", "Nintendo", "Nintendo", "Nintendo", "~
$ NA_Sales     <dbl> 34.02, 23.85, 13.00, 12.92, 9.24, 19.02, 9.33, 11.50, 11.96, ~
$ EU_Sales     <dbl> 23.80, 2.94, 10.56, 9.03, 7.29, 1.85, 7.57, 7.54, 5.79, 0.52,~
$ Global_Sales <dbl> 67.85, 33.00, 29.37, 27.06, 25.72, 24.81, 24.61, 23.80, 23.47~
```

Summary

```
> summary(turtle_sales)
    Ranking             Product          Platform                 Year     
 Min.   :    1.00   Min.   : 107    Length:352         Min.   :1982  
 1st Qu.:   88.75   1st Qu.:1945    Class :character   1st Qu.:2003  
 Median :  176.50   Median :3340    Mode  :character   Median :2009  
 Mean   : 1428.02   Mean   :3607                       Mean   :2007  
 3rd Qu.: 1439.75   3rd Qu.:5436                       3rd Qu.:2012  
 Max.   :16096.00   Max.   :9080                       Max.   :2016  
                                                       NA's   :2     
    Genre             Publisher            NA_Sales          EU_Sales    
 Length:352         Length:352         Min.   : 0.0000   Min.   : 0.000  
 Class :character   Class :character   1st Qu.: 0.4775   1st Qu.: 0.390  
 Mode  :character   Mode  :character   Median : 1.8200   Median : 1.170  
```

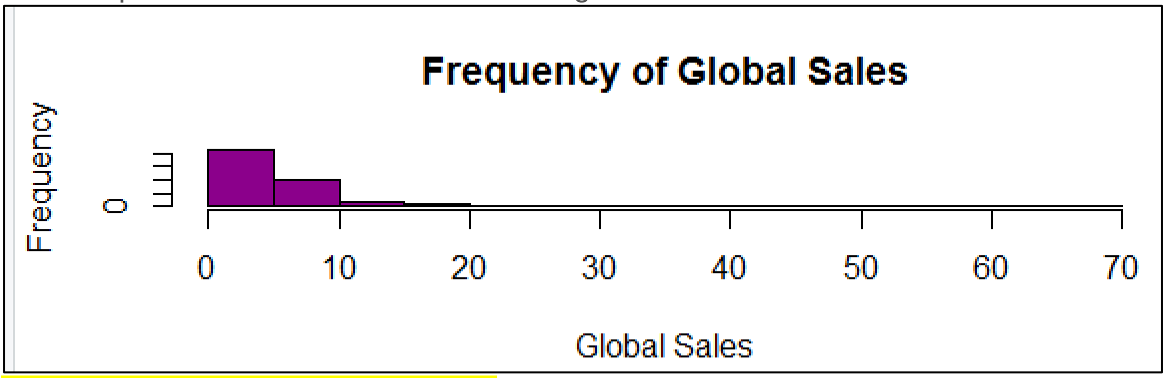
Removed redundant columns (Ranking, Year, Genre, and Publisher) by creating a subset of the data frame

```
# Remove  columns.
turtle_sales2 <- select(turtle_sales, -Ranking, -Year, -Genre, -Publisher )


# Check the new data frame.
head(turtle_sales2)
```

summary of the new data frame.

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

- Created plots to review and determine insights into the sales data set.




Frequency is higher below 10 million that means most of the products have global sale below 10 million

- Use of filter function to get details about highest global sale.

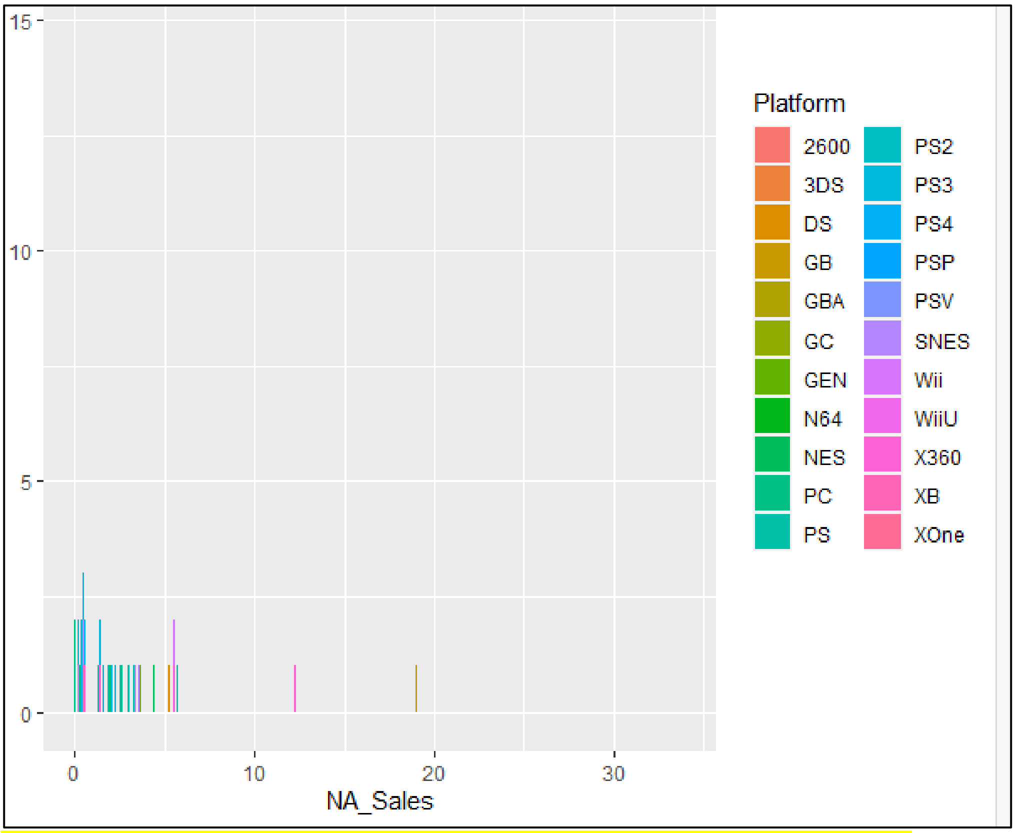
```
High_Global_sale <- filter(turtle_sales2,
                           Global_Sales > 50)

head(High_Global_sale)
```

```
> head(High_Global_sale)
  Product Platform NA_Sales EU_Sales Global_Sales
1     107      Wii    34.02     23.8        67.85
```

Product 107 has highest global sale.

- Qplot to check sale for north America region by different platforms

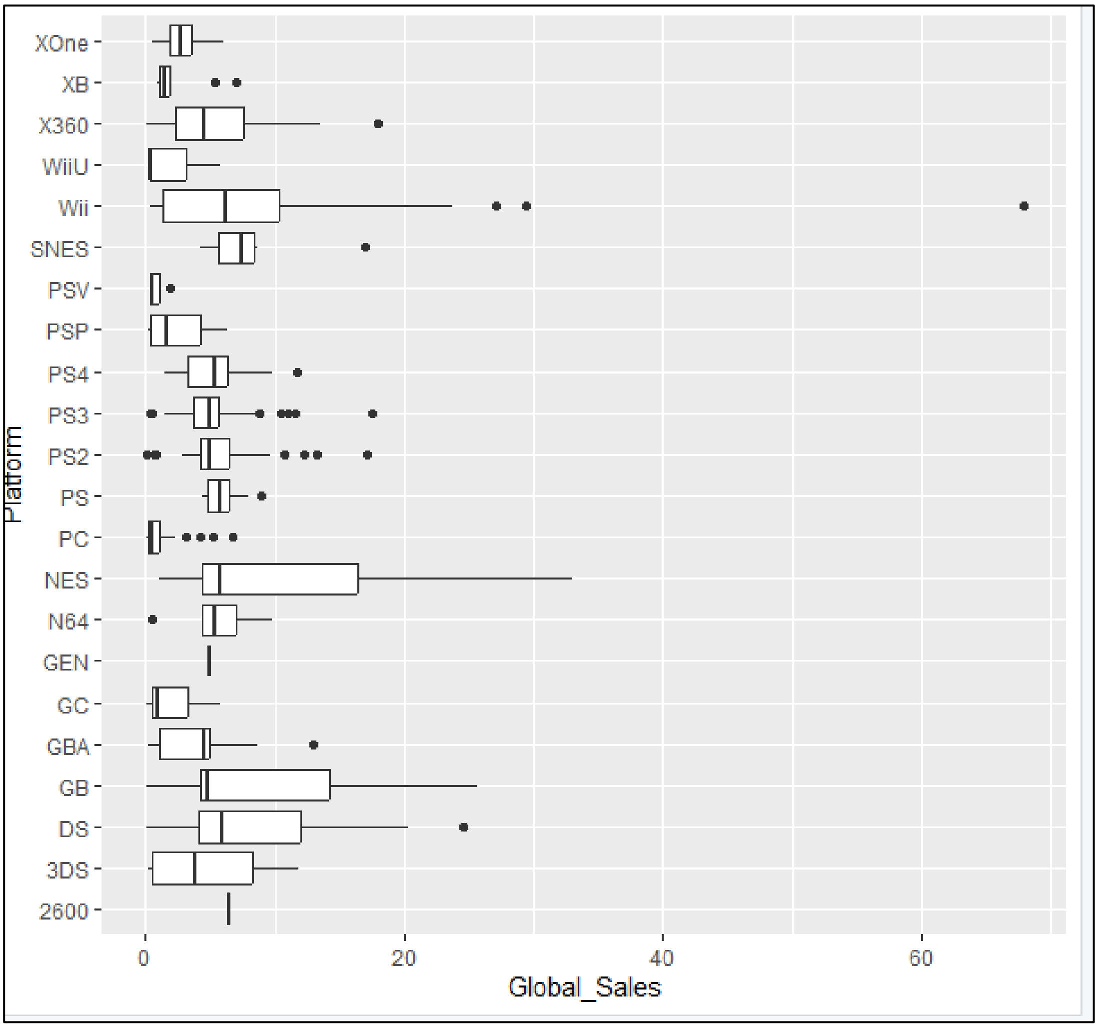
```
qplot(NA_Sales, fill=Platform, data=turtle_sales2, geom='bar')
qplot(EU_Sales, fill=Platform, data=turtle_sales2, geom='bar')
```




Most of the game platform has NA sale below 10 million pound and maximum NA sale is through PS games

- Box plot to check outliers

```
qplot(Global_Sales, Platform, data=turtle_sales2, geom='boxplot')
```




The output shows the global sales represented with respect to different platforms. We can see that the outlier of the Wii platform is particularly unique.

Some platform like GB, 3DS, NES, GC has no outliers and global sales are greater than mean for most of the times.

- Used mutate function to add another column with addition of EU and NA sale which gives idea of other sale with respect to

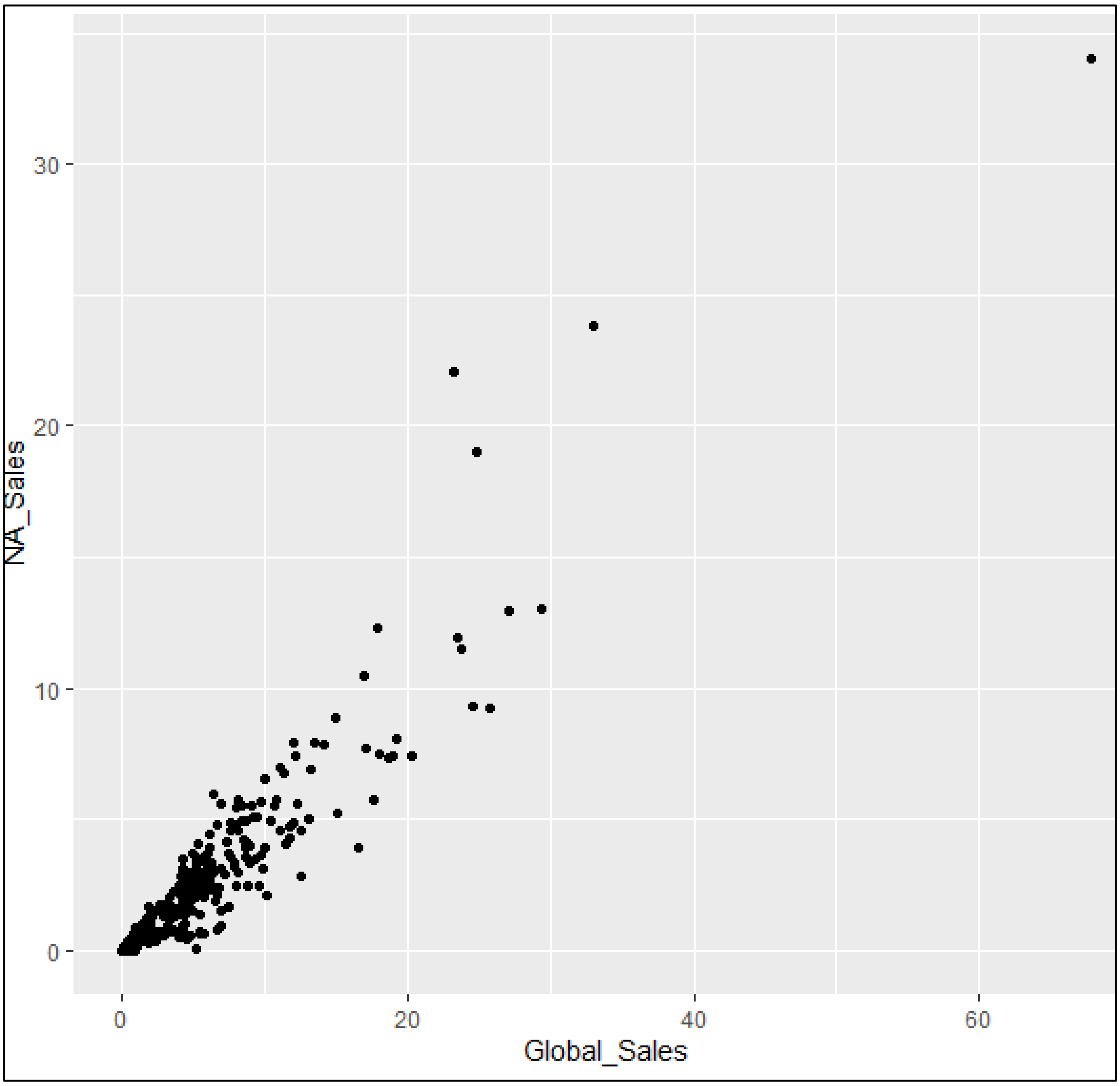
```
# To create a new element EU + NA sale.
turtle_sales2 <- mutate(turtle_sales2, new_var=EU_Sales + NA_Sales)

# View first 10 rows of the data frame.
head(turtle_sales2, 10)
## new var, EU sale
qplot(Global_Sales, NA_Sales, data=turtle_sales2)

qplot(y=Global_Sales, data=turtle_sales2)
```

|  | Product | Platform | NA_Sales | EU_Sales | Global_Sales | new_var |
|---|---|---|---|---|---|---|
| 1 | 107 | Wii | 34.02 | 23.80 | 67.85 | 57.82 |
| 2 | 123 | NES | 23.85 | 2.94 | 33.00 | 26.79 |
| 3 | 195 | Wii | 13.00 | 10.56 | 29.37 | 23.56 |
| 4 | 231 | Wii | 12.92 | 9.03 | 27.06 | 21.95 |
| 5 | 249 | GB | 9.24 | 7.29 | 25.72 | 16.53 |
| 6 | 254 | GB | 19.02 | 1.85 | 24.81 | 20.87 |
| 7 | 263 | DS | 9.33 | 7.57 | 24.61 | 16.90 |
| 8 | 283 | Wii | 11.50 | 7.54 | 23.80 | 19.04 |
| 9 | 291 | Wii | 11.96 | 5.79 | 23.47 | 17.75 |
| 10 | 326 | NES | 22.08 | 0.52 | 23.21 | 22.60 |

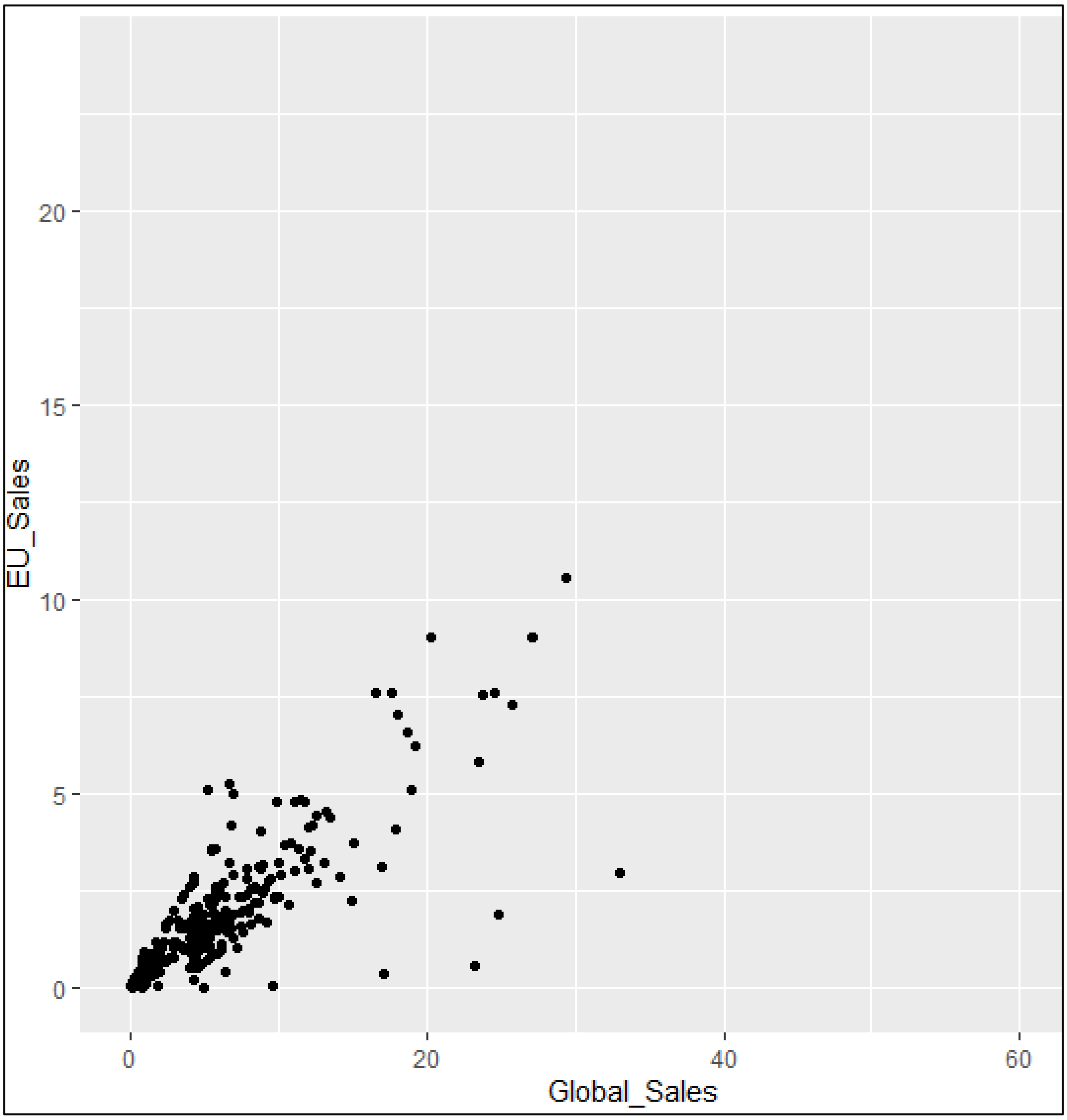
>

- Scatter plot to analyse co-relation between different sales columns

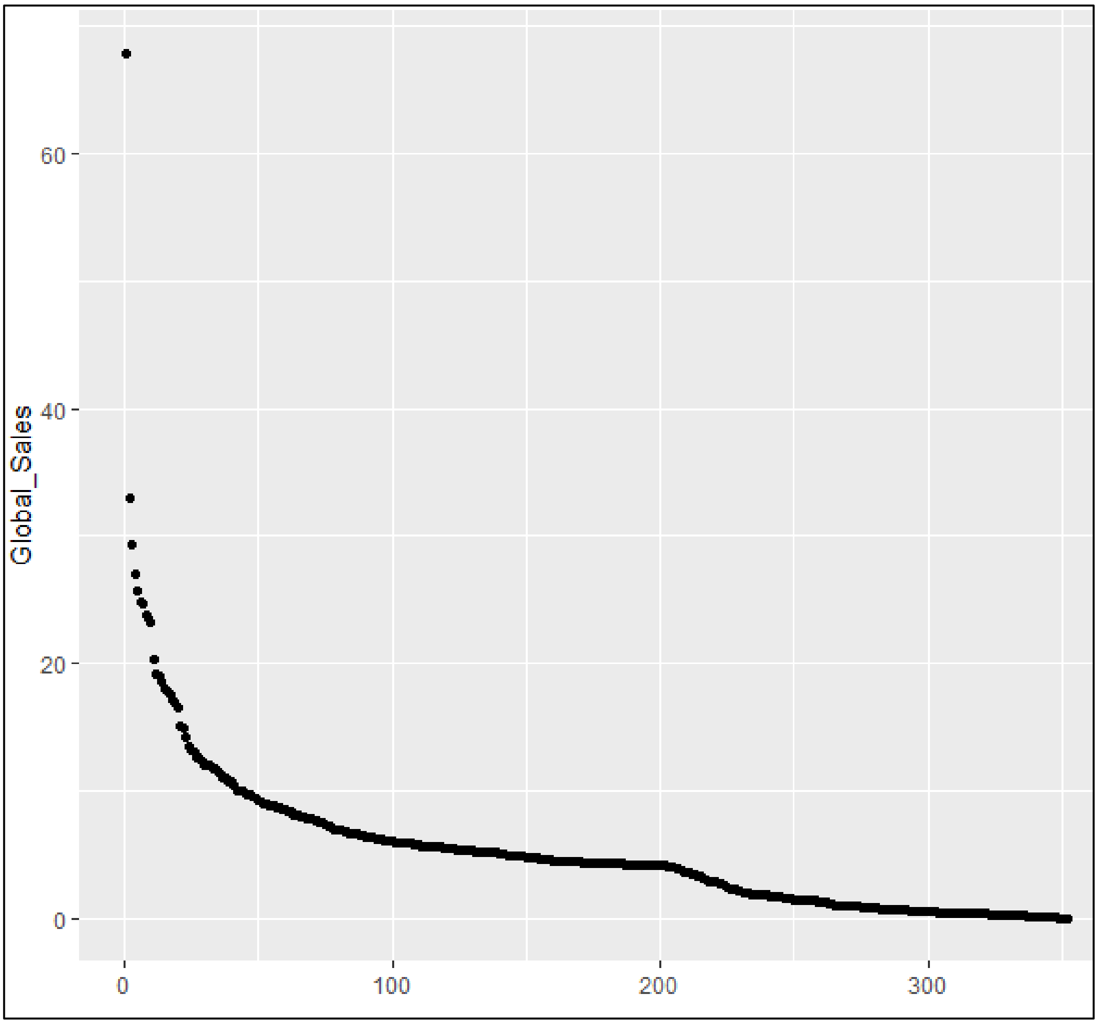
```
qplot(Global_Sales, NA_Sales, data=turtle_sales2)
```




Scatter plot is quiet co-related until 10 million and more scattered after 20 million.

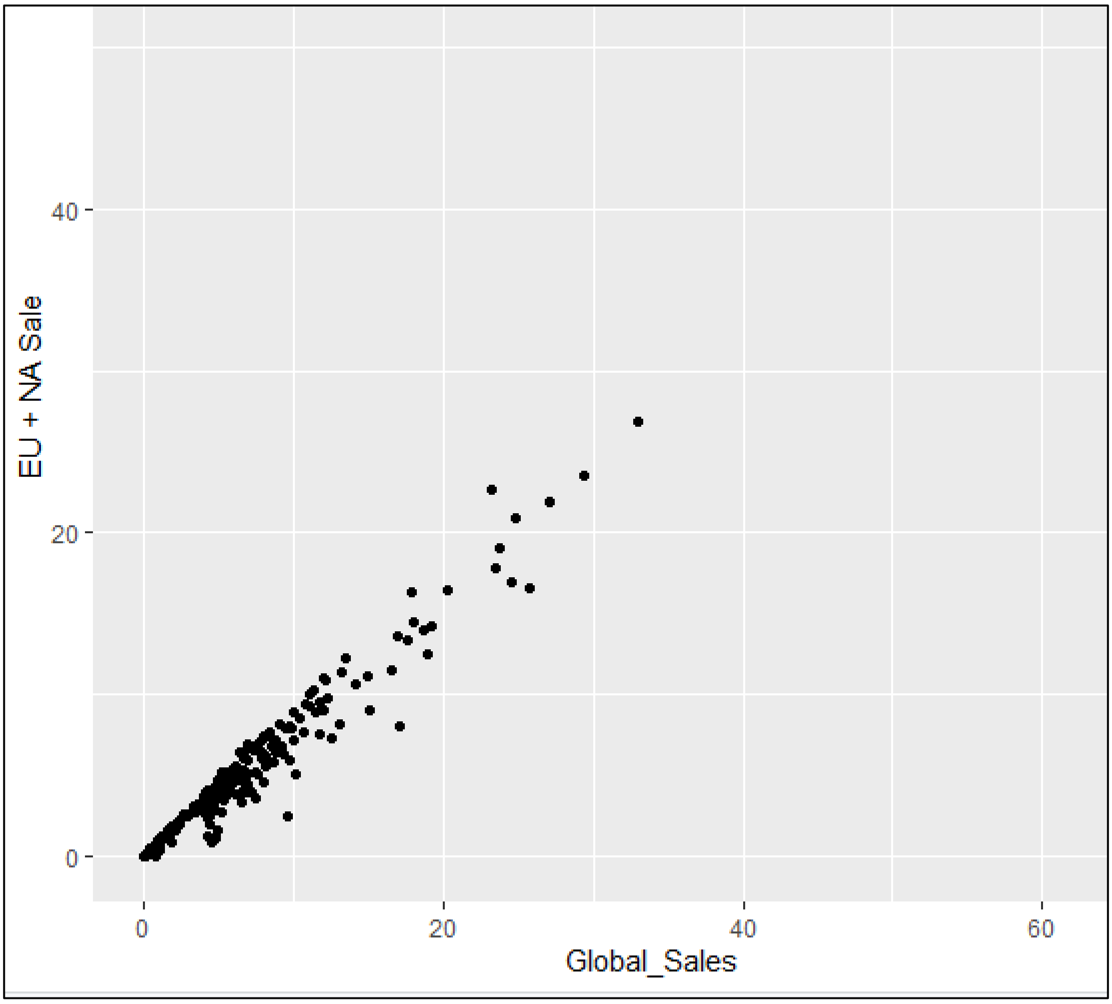



Less co-related (more scattered) as compared to scatter plot with NA Sale, Indicating NA region sale has more contribution in global sale




Global sale for most of the video game products is less than 10 million pounds

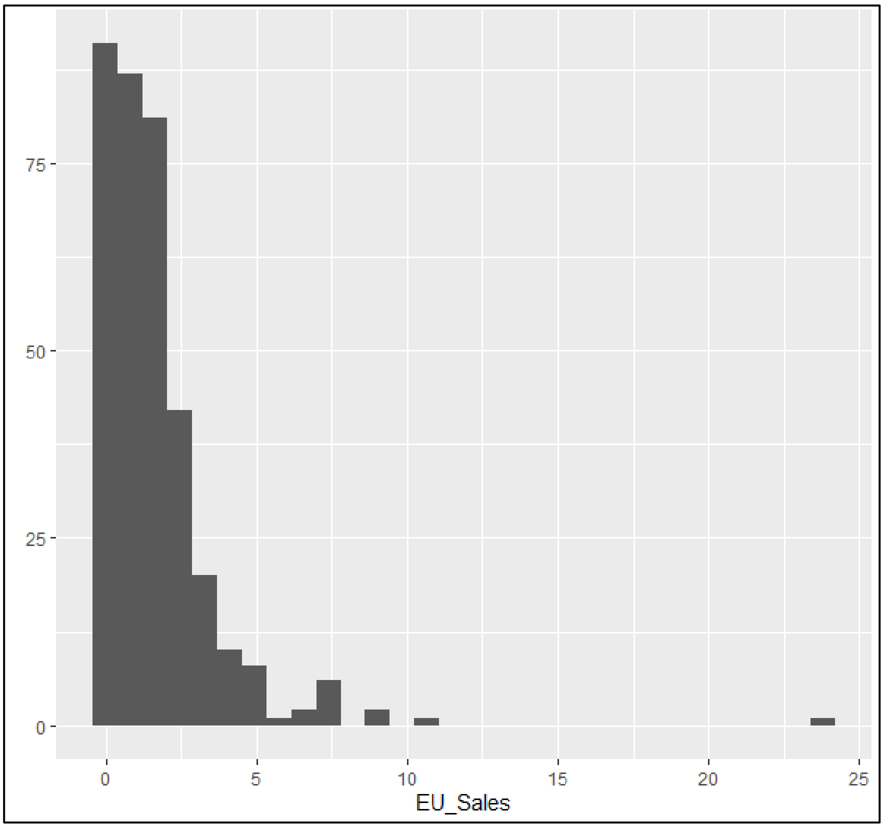
```
qplot(Global_Sales, new_var, data=turtle_sales2,
      ylab ="EU + NA Sale")
```



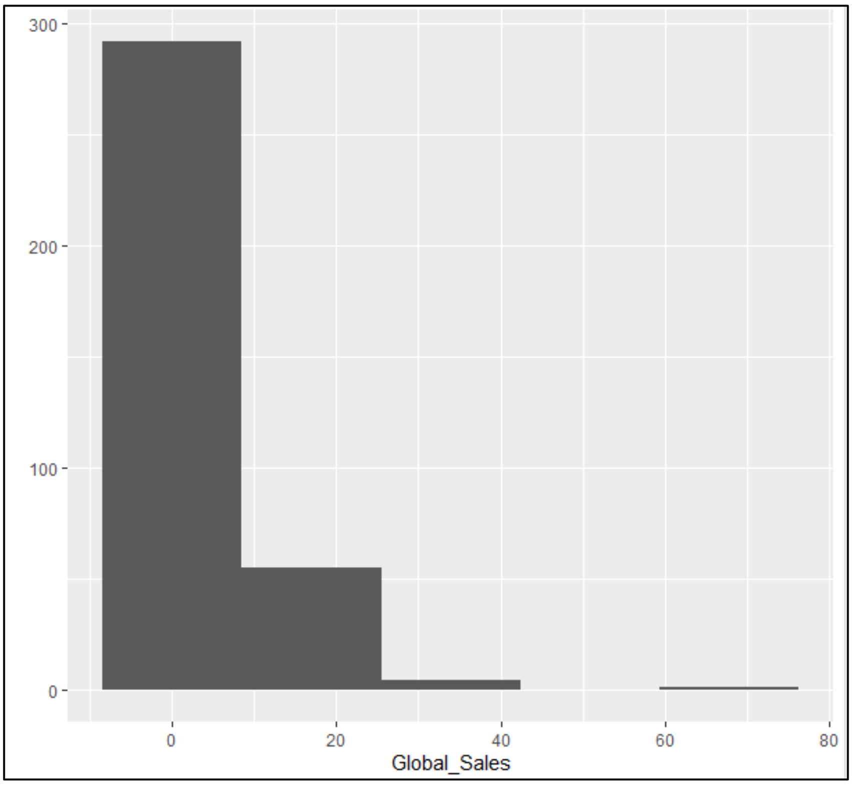

EU+ NA sale is linearly co-related with Global sale indicating other sale contribution is very less

- Analysis with Histogram

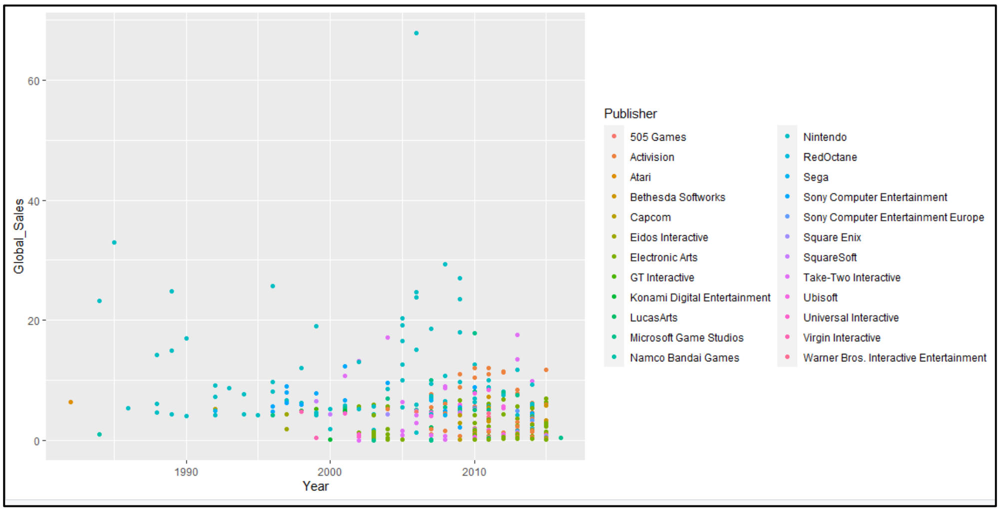
```
######Histogram
# First pass the x-variable, then specify the data source.
qplot(EU_Sales, data=turtle_sales2)

qplot(Global_Sales, bins=5, data=turtle_sales2)
```



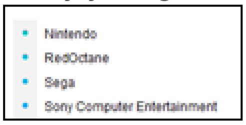

Data is right skewed




Data is right skewed




Every year global sales are higher for these publishers:




## 6. The reliability of the data (e.g. normal distribution, Skewness, Kurtosis)

Load and explore the data

- Viewed the data frame to sense-check the data set.
- Determined the min, max and mean values of all the sales data (three columns).

```
> apply(turtle_sales2, 2, min)
    Product    Platform    NA_Sales    EU_Sales Global_Sales     new_var
     " 107"      "2600"     " 0.00"     " 0.00"      " 0.01"     " 0.00"
> apply(turtle_sales2, 2, max)
    Product    Platform    NA_Sales    EU_Sales Global_Sales     new_var
     "9080"      "XOne"     "34.02"     "23.80"      "67.85"     "57.82"
```

- Create a summary of the data frame.

```
> apply(turtle_sale_only, 2, mean)
    NA_Sales     EU_Sales Global_Sales      new_var
    2.515966     1.643778     5.334688     4.159744
```

- Use the group_by, apply(), and/or aggregate functions to sum the values grouped by product to determine the impact on sales per product_id

```
> aggregate(Global_Sales~Platform, turtle_sales2, sum)
   Platform Global_Sales
1      2600         6.40
2       3DS        73.20
3        DS       205.02
4        GB       133.97
5       GBA        47.10
6        GC        21.66
7       GEN         4.94
8       N64        44.50
9       NES        91.40
10       PC        43.08
11       PS        82.92
12      PS2       131.87
13      PS3       211.61
```

Looks like highest sale is done by PS3

```
> aggregate(Global_Sales~Product, turtle_sales2, sum)
    Product Global_Sales
1       107        67.85
2       123        37.16
3       195        29.37
4       231        27.06
5       249        25.72
6       254        29.39
7       263        24.61
8       283        23.80
9       291        23.47
10      326        23.21
11      399        20.30
12      405        19.20
13      453        18.94
```

Highest Global sale is through product 107

```
> aggregate(Global_Sales~Product+Platform, turtle_sales2, sum)
    Product Platform Global_Sales
1      2829     2600         6.40
2       977      3DS        11.77
3      1183      3DS        10.01
4      1473      3DS         9.29
5      1577      3DS         8.85
6      2114      3DS         8.05
7      2286      3DS         7.45
8      2518      3DS         0.24
9      2521      3DS         0.62
10     3112      3DS         3.45
11     3165      3DS         6.11
```

Mean global sale per product and platform. Example mean sale for 3DS is 73.2 which is distributed among different products

```
> # EU sale grouped by product
> df_sale <- turtle_sales2 %>% group_by(Product, Platform) %>%
+   summarise(sum_EU_Sale=sum(EU_Sales),
+   .groups='drop') %>%
+   arrange(desc(sum_EU_Sale))
> # View the results.
> df_sale
# A tibble: 352 x 3
   Product Platform sum_EU_Sale
     <int> <chr>          <dbl>
 1     107 Wii             23.8
 2     195 Wii             10.6
 3     231 Wii              9.03
```

107 has maximum EU sale and Wii platform has more sale

Total sale contributions

```
> # Total Sale
> lapply(turtle_sale_only, sum)
$NA_Sales
[1] 885.62

$EU_Sales
[1] 578.61

$Global_Sales
[1] 1877.81

$new_var
[1] 1464.23
```

```
> # NA sale grouped by product
> df_sale_NA <- turtle_sales2 %>% group_by(Product, Platform) %>%
+   summarise(sum_NA_Sale=sum(NA_Sales),
+             .groups='drop') %>%
+   arrange(desc(sum_NA_Sale))
> # View the results.
> df_sale_NA
# A tibble: 352 x 3
   Product Platform sum_NA_Sale
     <int> <chr>          <dbl>
 1     107 Wii             34.0
 2     123 NES             23.8
 3     326 NES             22.1
```

```
   Product Platform sum_NA_Sale
     <int> <chr>          <dbl>
 1     107 Wii             34.0
 2     123 NES             23.8
 3     326 NES             22.1
 4     254 GB              19.0
 5     195 Wii             13
 6     231 Wii             12.9
 7     504 X360            12.3
 8     291 Wii             12.0
 9     283 Wii             11.5
10     535 SNES            10.5
```

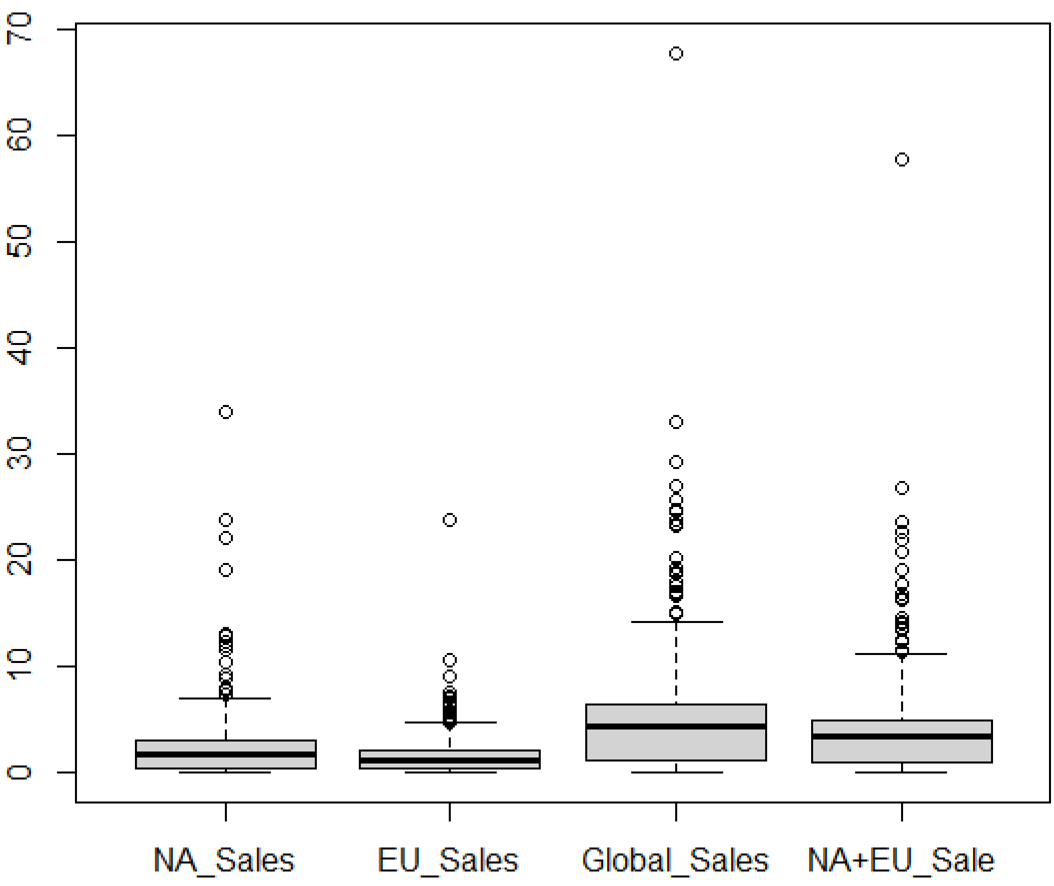
Major NA sale is from wii platform

- Summary of sale data

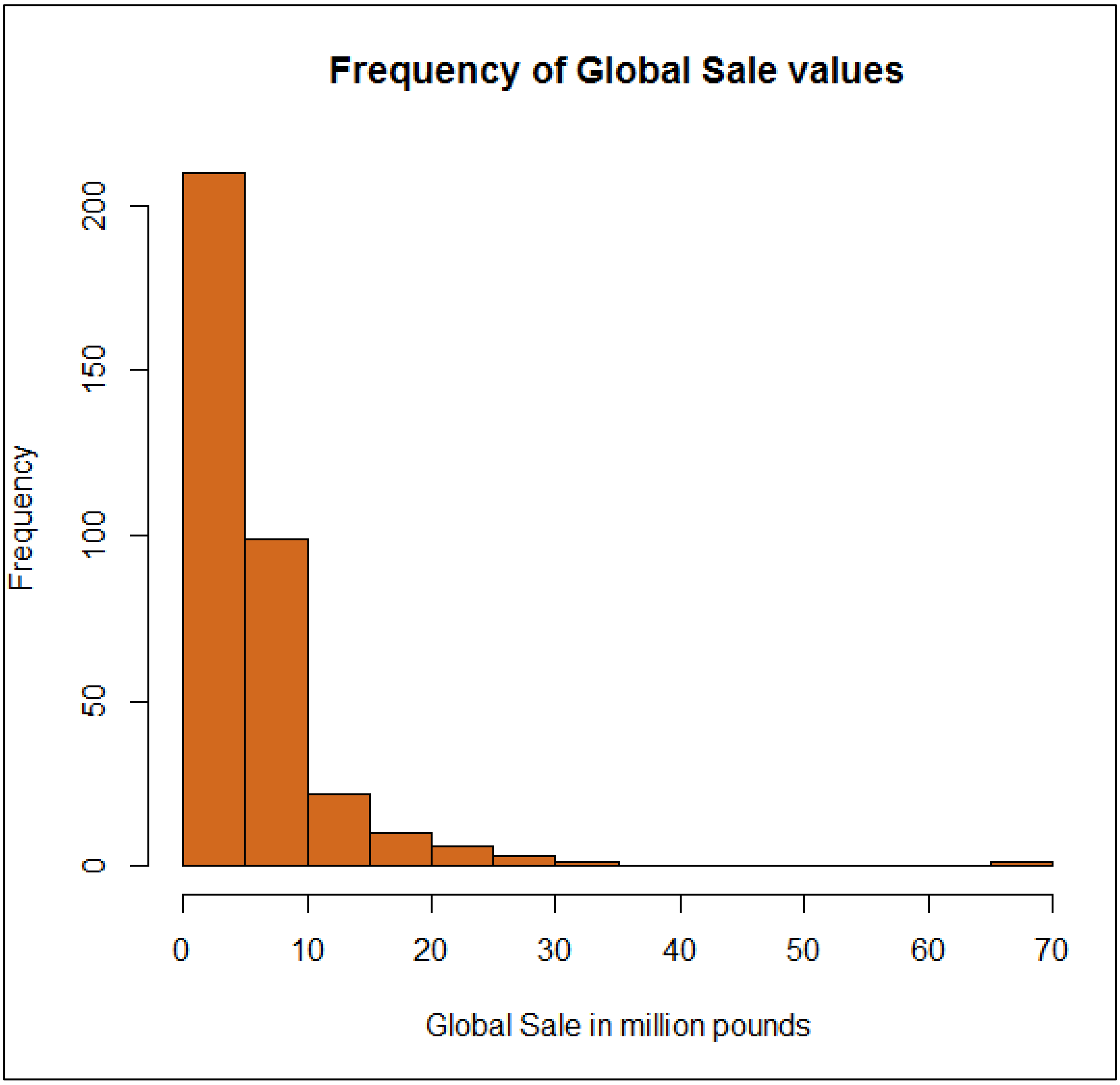
```
> turtle_sale_only <- turtle_sale_only %>%
+   rename("NA+EU_Sale" = "new_var" )
> summary(turtle_sale_only)
    NA_Sales           EU_Sales         Global_Sales         NA+EU_Sale
 Min.   : 0.0000   Min.   : 0.000   Min.   : 0.010   Min.   : 0.000
 1st Qu.: 0.4775   1st Qu.: 0.390   1st Qu.: 1.115   1st Qu.: 0.945
 Median : 1.8200   Median : 1.170   Median : 4.320   Median : 3.390
 Mean   : 2.5160   Mean   : 1.644   Mean   : 5.335   Mean   : 4.160
 3rd Qu.: 3.1250   3rd Qu.: 2.160   3rd Qu.: 6.435   3rd Qu.: 5.010
 Max.   :34.0200   Max.   :23.800   Max.   :67.850   Max.   :57.820
```

IQR range (q3-q1) or major of dispersion( Sd) is above 5 for global sale

- Boxplot---Fix line is median and line at top is longer hence result is +ve skewed

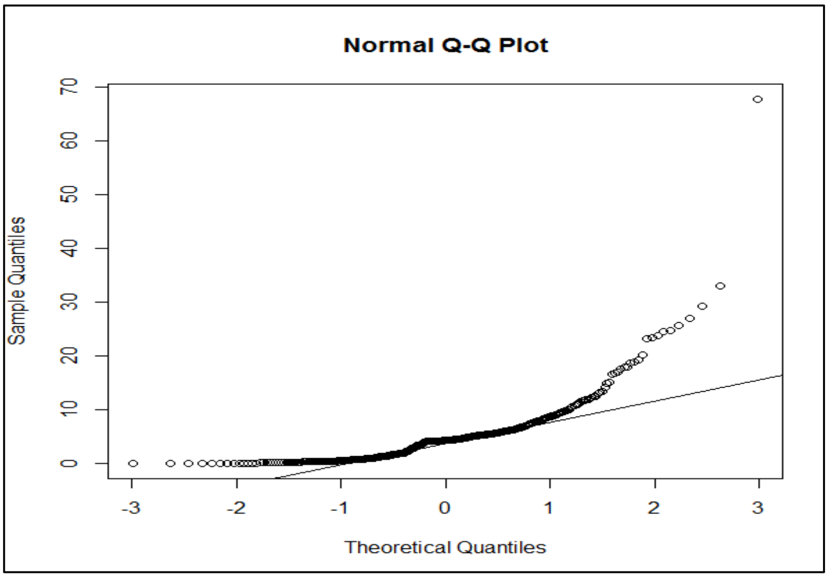



- Histogram




Hump at zero, positive or right skewed

- Plot and compare with normal distribution →qqplot




Data values near mean and one std deviation above / below mean of the normal are on line (-1 to 1) and are perfect fits points But tails on both side are quiet far from mean

- Determined the normality of the data set (sales data).
  - Created and explored Q-Q plots for all sales data.
  - Performed a Shapiro-Wilk test on all the sales data.

```
#Hypothesis test
shapiro.test(turtle_sale_only$Global_Sales)
```

```
        Shapiro-Wilk normality test

data:  turtle_sale_only$Global_Sales
W = 0.6818, p-value < 2.2e-16
```

Very small P-value hence null hypothesis can be rejected.

Since data size is less, and QQ plot is not quiet linear, normality can be rejected.

```
> dim(turtle_sale_only)
[1] 352   4
```

- Determine the Skewness and Kurtosis of all the sales data.

```
> skewness(turtle_sale_only$Global_Sales)
[1] 4.045582
```

Strong positive skew, strong positive value means heavy tailed distribution.

```
> kurtosis(turtle_sale_only$Global_Sales)
[1] 32.63966
```

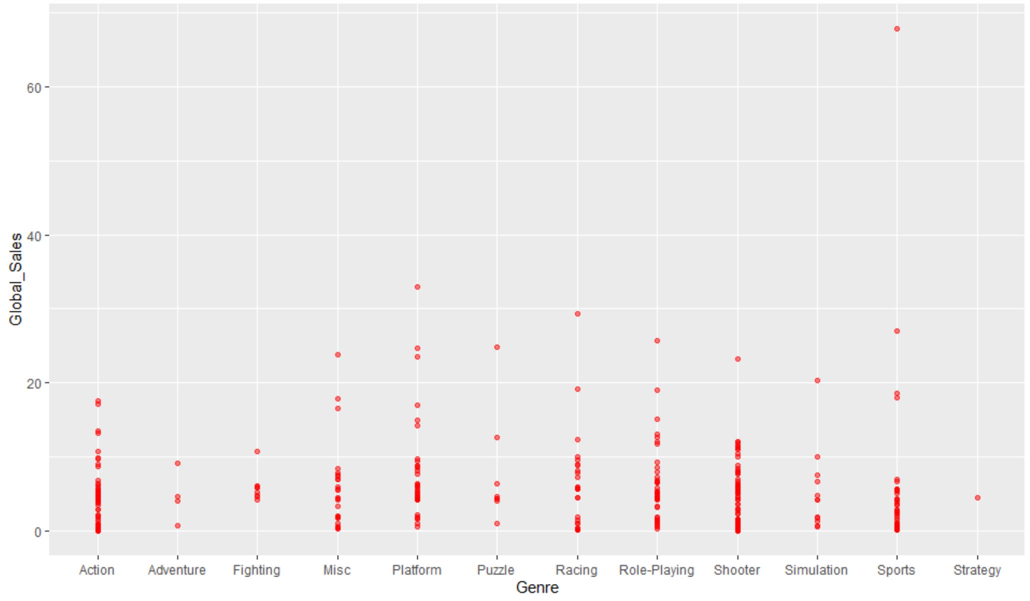
Heavy tailed as kurtosis much greater than 3

- Determine if there is any correlation between the sales data columns.

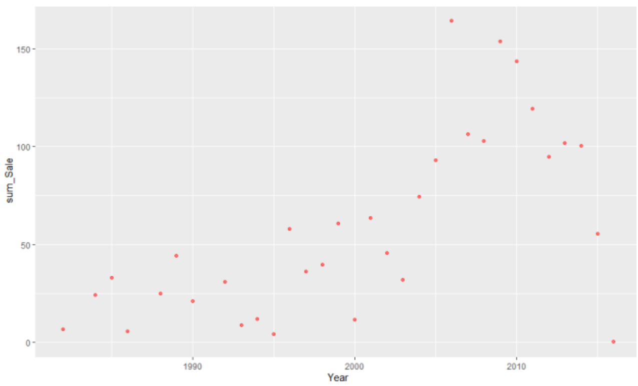
```
> round (cor(turtle_sale_only),
+        digits=2)
             NA_Sales EU_Sales Global_Sales NA+EU_Sale
NA_Sales         1.00     0.71         0.93       0.96
EU_Sales         0.71     1.00         0.88       0.88
Global_Sales     0.93     0.88         1.00       0.98
NA+EU_Sale       0.96     0.88         0.98       1.00
```

- The correlation between NA and Global sale is 0.93. A positive correlation coefficient suggests that the two variables vary in the same direction. That means as the one increases, so does the other; and if one decreases, the other does too. Again the coefficient is closer to 1, meaning there is a strong positive correlation. This means that NA sale strongly correlates with Global sale

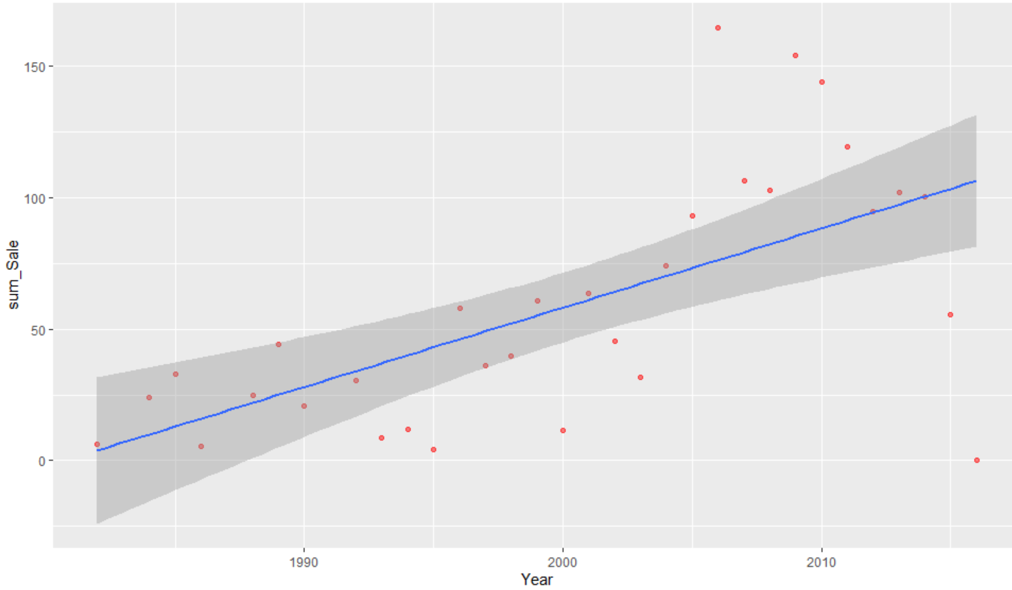
- Created plots to gain further insights into the sales data by Genre




- Created a scatter plot with the point colour red, the size set at 1.5, and the alpha to 0.5.
  There is no clear relationship visible between Year and Sale. Let's explore more.



- Further added a best line of fit to determine whether we have a relationship between year and sale.
- In ggplot2, a smoothing line is the same as a line-of-best fit, which is a line through a scatterplot that best expresses the relationship between all the points.
- Compare all the sales data (columns) for any correlation(s).
- Add a trend line to the plots for ease of interpretation.



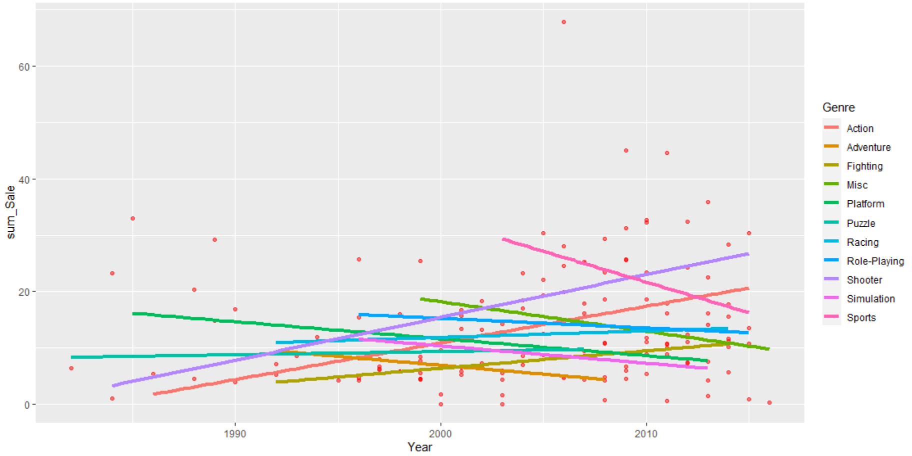

- 

```
> head(df_sale_plot)
# A tibble: 6 x 2
   Year sum_Sale
  <dbl>    <dbl>
1  2016     0.35
2  2015    55.5
3  2014   100.
4  2013   102.
5  2012    94.9
6  2011   119.
> Platform
```



Note the blue line-of-best-fit running through the points. The line tells us that sale increase per year. The faint shaded space on either side of the line represents the confidence intervals.

- Yearly sale among different Genres
  Yearly Sale is increasing for Action and simulation and decreasing for sports,

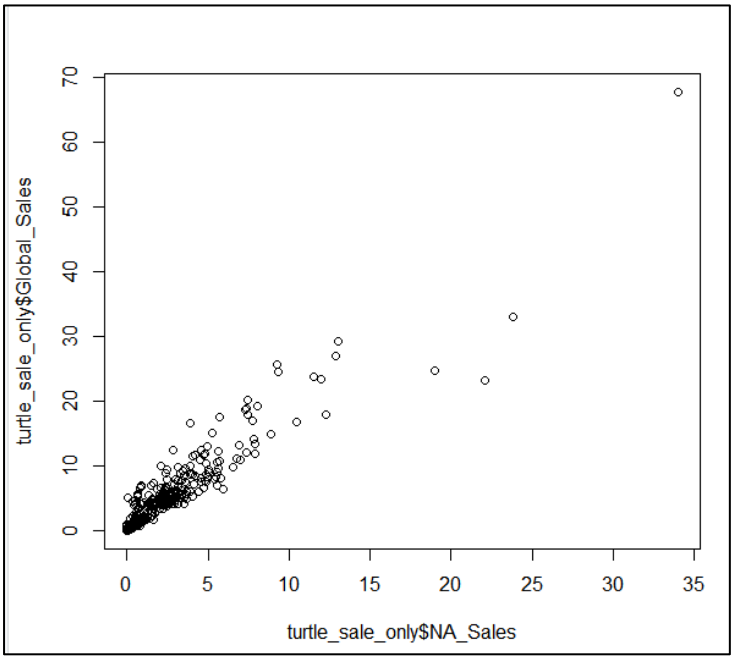


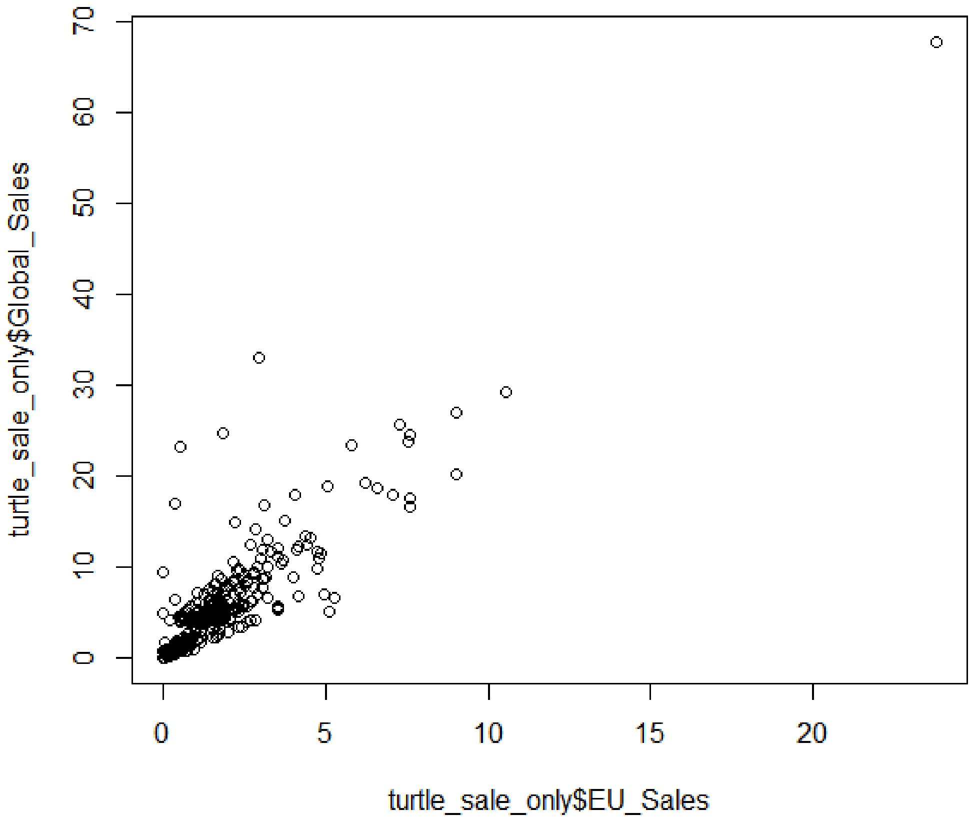

## 7. The sales department wants to better understand whether there is any relationship between North America, Europe, and global sales.

- Created a simple linear regression model.
  - Determined the correlation between the sales columns.
  - View the output.

```
> cor(turtle_sale_only)
             NA_Sales  EU_Sales Global_Sales
NA_Sales     1.0000000 0.7055236    0.9349455
EU_Sales     0.7055236 1.0000000    0.8775575
Global_Sales 0.9349455 0.8775575    1.0000000
```

  - Created plots to view the linear regression.( Global and NA sale strongly co-related as compared to EU )






- NA Sale

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

Residuals:
     Min       1Q   Median       3Q      Max 
-15.7352  -1.0341  -0.5555   0.6247   8.8676 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  1.01232    0.14752   6.862 3.09e-11 ***
NA_Sales     1.71797    0.03485  49.300  < 2e-16 ***
```

```
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.226 on 350 degrees of freedom
Multiple R-squared:  0.8741,	Adjusted R-squared:  0.8738 
F-statistic:  2430 on 1 and 350 DF,  p-value: < 2.2e-16
```

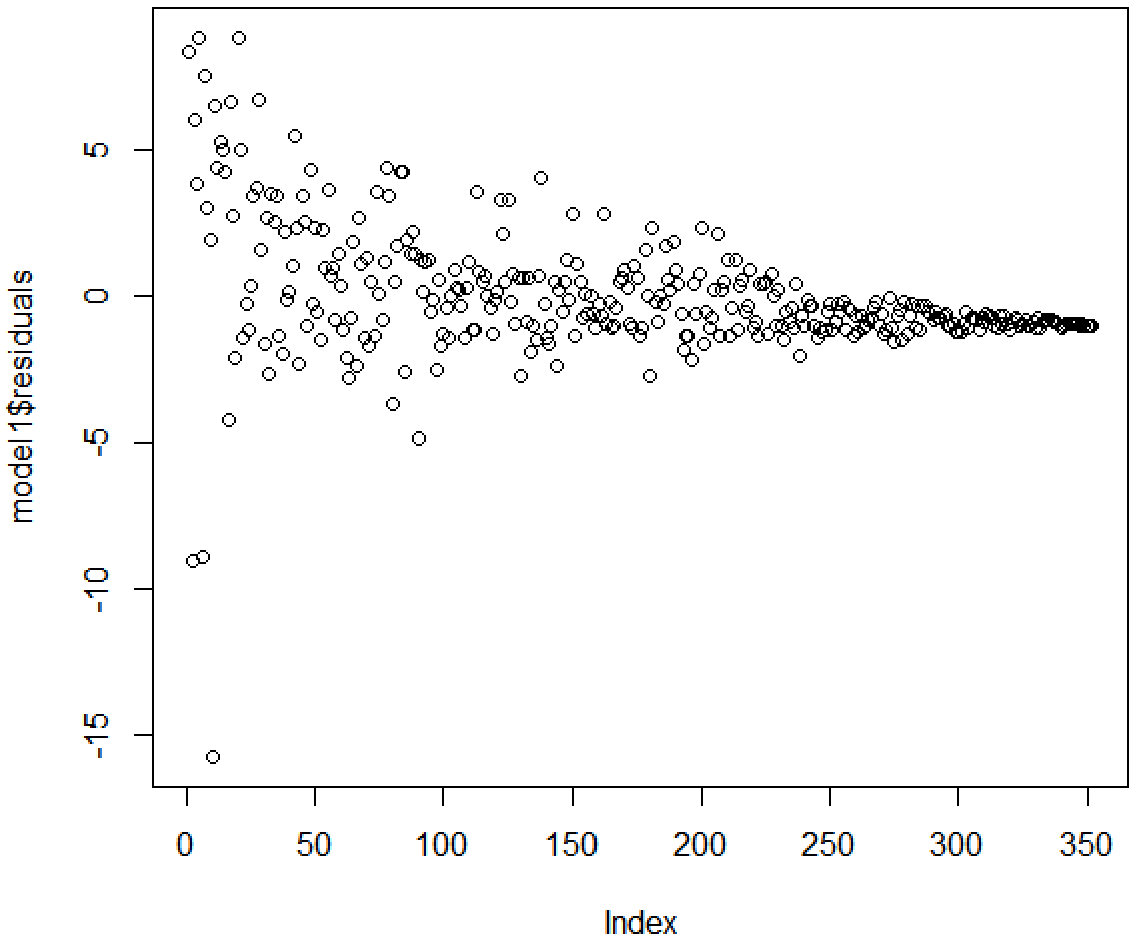
- EU Sale

```
> # View more outputs for the model - the full regression table.
> summary(model2)

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

Residuals:
    Min      1Q  Median      3Q     Max 
-9.5377 -1.2173 -0.6040  0.8755 24.1474 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  0.87350    0.20660   4.228 3.01e-05 ***
EU_Sales     2.71399    0.07926  34.241  < 2e-16 ***
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.008 on 350 degrees of freedom
Multiple R-squared:  0.7701,	Adjusted R-squared:  0.7695 
F-statistic:  1172 on 1 and 350 DF,  p-value: < 2.2e-16
```

# NA and EU sale has a highly significant value, explaining over 87.4% and 77%  of the variability respectively.

**Plot the model**



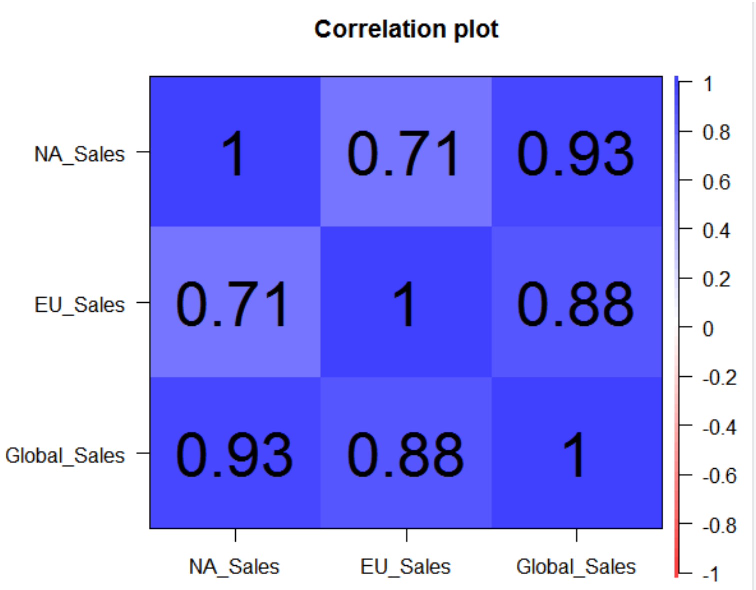

No pattern

**Conclusion:**

The  data consists of two columns (Yearly Global sale and sales from region EU and NA) . There is a strong positive correlation (87%) between Global Sale and NA Sale, with a coefficient of 1.718 (model1). Therefore, the global sale will increase by 1.718units every year. The standard error is low(0.14752), the R2 of 85.53% indicated a good fit.

The residual plot indicates  no pattern

- Created a multiple linear regression model.
  - Select only the numeric columns.
  - Determine the correlation between the sales columns.




- Create a new object or model, modela
- Specify the summary () model and pass modela to print summary statistics of the MLR.

```
> # specify the lm function and the variables.
> modela = lm(Global_Sales~NA_Sales+EU_Sales, data=turtle_sale_only)
>
> # Print the summary statistics.
> summary(modela)

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

Residuals:
    Min      1Q  Median      3Q     Max
-3.6186 -0.4234 -0.2692  0.0796  7.4639

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.22175    0.07760   2.858  0.00453 **
NA_Sales     1.15543    0.02456  47.047  < 2e-16 ***
EU_Sales     1.34197    0.04134  32.466  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.112 on 349 degrees of freedom
Multiple R-squared:  0.9687,    Adjusted R-squared:  0.9685
F-statistic:  5398 on 2 and 349 DF,  p-value: < 2.2e-16
```

Multiple R2 is 96%---very accurate

- Predict global sales based on provided values. Compare your prediction to the observed value(s).

| Ranking | Product | Platform | Year | Genre | Publisher | NA_Sales | EU_Sales | Global_Sales |
|---|---|---|---|---|---|---|---|---|
| 1 | 107 | Wii | 2006 | Sports | Nintendo | 34.02 | 23.8 | 67.85 |
| 10 | 326 | NES | 1984 | Shooter | Nintendo | 22.08 | 0.52 | 23.21 |
| 99 | 3267 | X360 | 2008 | Shooter | Activision | 3.93 | 1.56 | 6.04 |
| 176 | 6815 | N64 | 1999 | Platform | Nintendo | 2.73 | 0.65 | 4.32 |
| 258 | 2877 | X360 | 2014 | Shooter | Activision | 2.26 | 0.97 | 3.53 |

Predicted Global sale values

| | fit | lwr | upr |
|---|---|---|---|
| 1 | 71.468572 | 70.162421 | 72.774723 |
| 2 | 26.431567 | 25.413344 | 27.449791 |
| 3 | 6.856083 | 6.718420 | 6.993745 |
| 4 | 4.248367 | 4.102094 | 4.394639 |
| 5 | 4.134744 | 4.009122 | 4.260365 |

Overall, we can conclude that modelc best predicts the Global Sale, as it most accurately predicted actual values.