DSCC/CSC/STAT 462 Final Project

Japnit Singh Ahluwalia, Neel Agarwal, Jayant Patil, Pranav Yeola

2023-12-03

Descroptive Statistics

Statistics for number of seasons:

```
col_stats(sample_data,"number_of_seasons")
```

```
## Table for  number_of_seasons
##
## Numeric Summary:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    1.00    1.00    1.56    1.00  240.00
##
## Coefficient of Variance= 1.91068155329941
## Variance= 8.88481207817618
## Skewness= 14.0225525779676
## IQR= 0
## 10% Trimmed Mean 1.07994563202768
## ____________________-_____________-_______________________
```

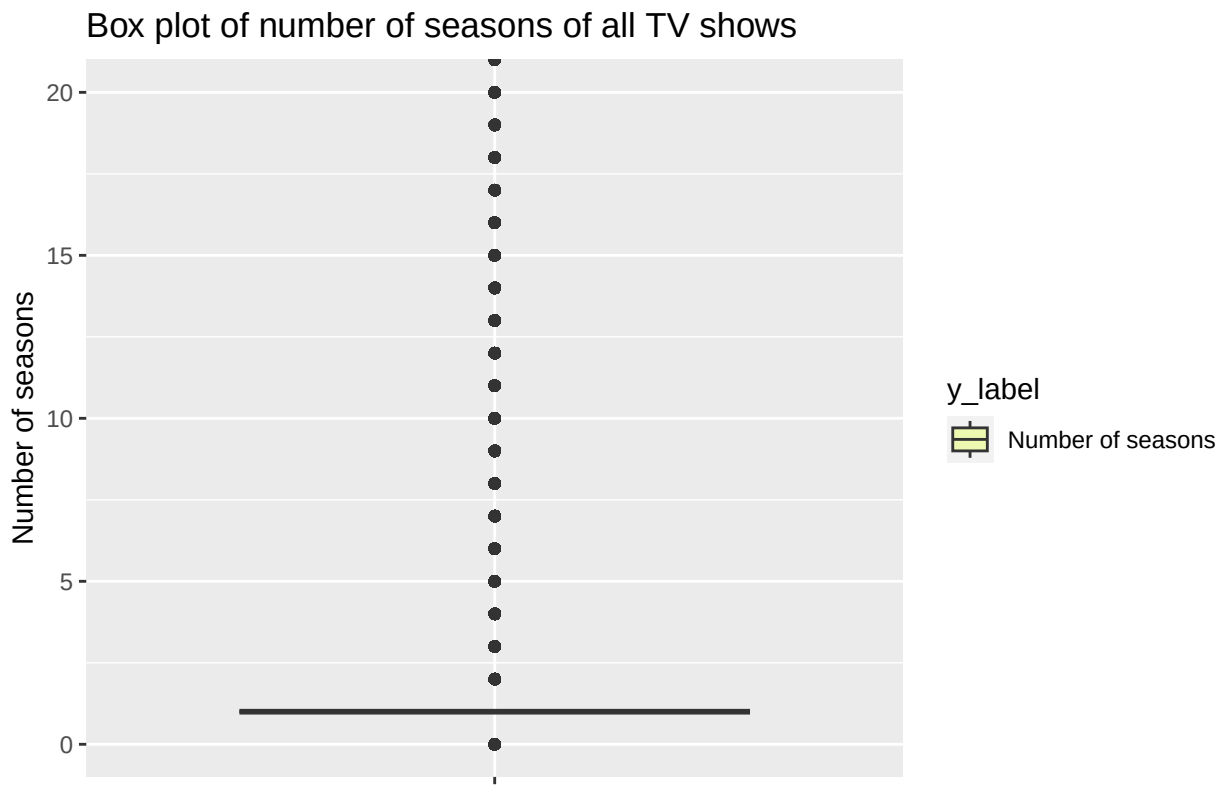
By the summary statistics it is clear that for some of the TV shows, the number of seasons has values as 0 (min), which implies that those shows either did not follow the concept of seasons or the information is not available

Also since Q1=Q2=Q3 we can conclude that almost 75% of the values equal to 1 implying one season only, however the maximum value is 240 which is significantly higher

The skewness value of 14.022 indicates substantial right skewness, i.e., extreme values in the higher range.

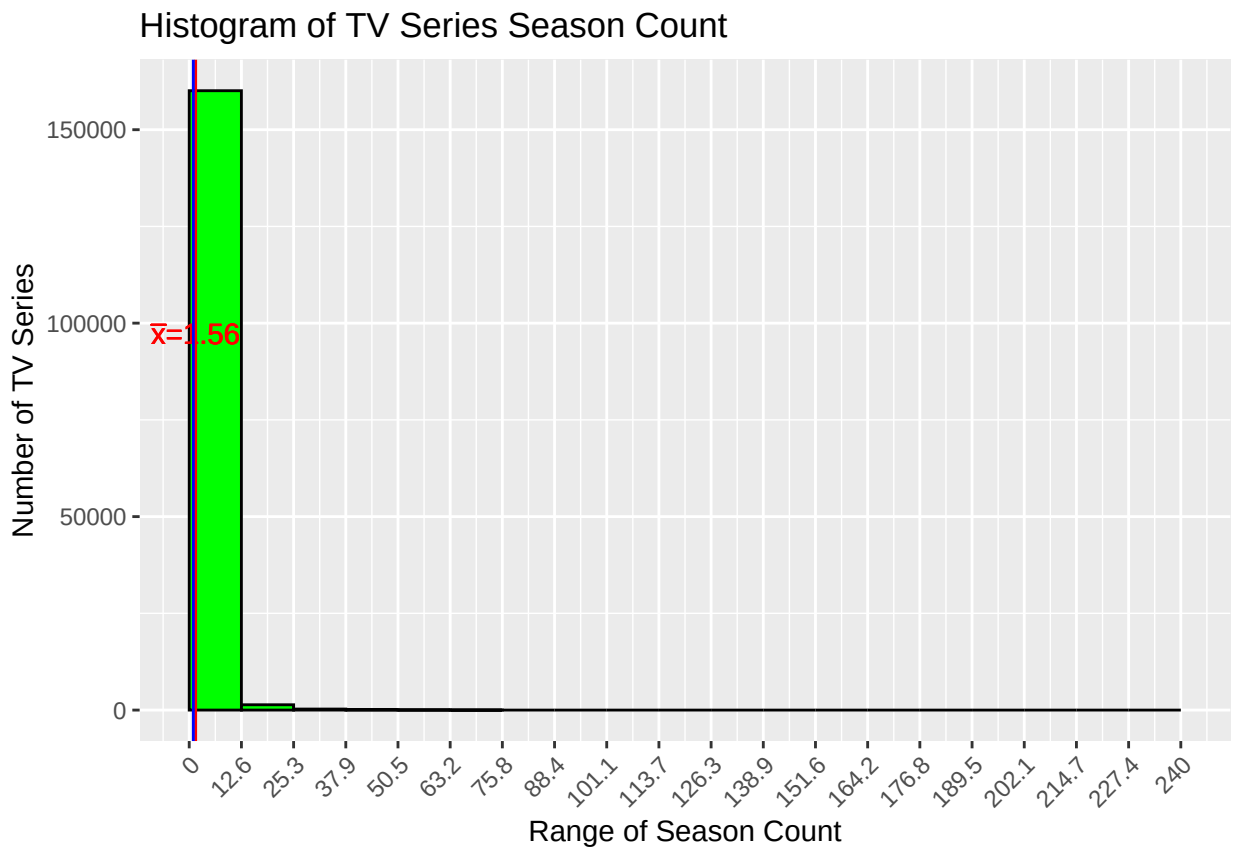
The IQR has been reported as 0, suggesting that the middle 50% of the data points are relatively tightly clustered around the median (i.e., the middle 50% of values = 1)

```
generate_box_plot(
  data = sample_data,
  y_col = "number_of_seasons",
  y_label = "Number of seasons",
  title = "Box plot of number of seasons of all TV shows",
  ylim = c(0, 20)
)
```



The box plot obtained shows that the middle 50% of the values are equal to 1 as the Q1, Q2 and Q3 have converged to 1 whereas higher number of seasons have been shown as outliers in the plot.

```
plot_histogram(
  data = sample_data,
  column_name = "number_of_seasons",
  x_label = "Range of Season Count",
  y_label = "Number of TV Series",
  title = "Histogram of TV Series Season Count",
  type = 0,
  fillColor = "green")
```



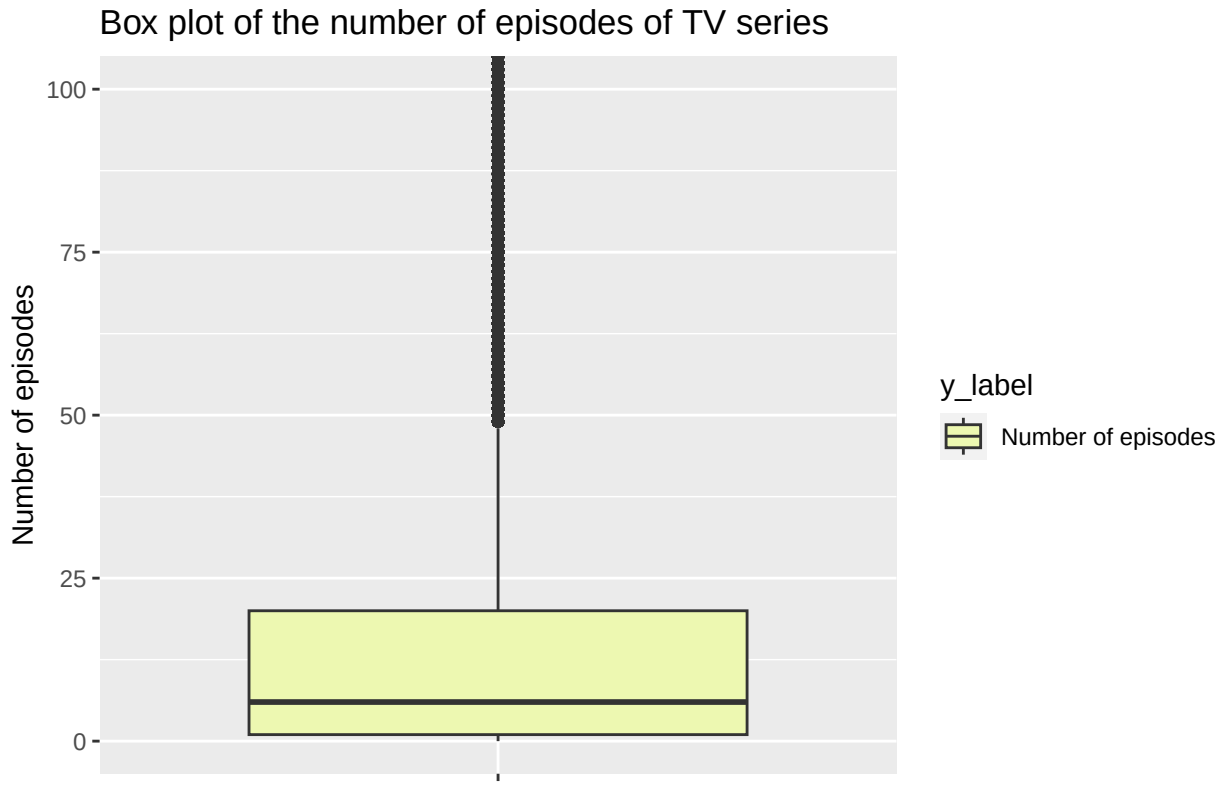
The histogram further confirms the right skewness of the variable and that more than 100000 observations have 0 or 1 season. But since Q1 = Q2 = Q3 = 1, the majority of observations seem to have a single season only

Statistics on number of episodes

```
#NUMBER OF EPISODES
col_stats(sample_data,"number_of_episodes");
```

```
## Table for  number_of_episodes
##
## Numeric Summary:
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##     0.00     1.00     6.00    25.06    20.00 20839.00
##
## Coefficient of Variance= 5.48439470714151
## Variance= 18889.2597903455
## Skewness= 54.3759907210613
## IQR= 19
## 10% Trimmed Mean 10.5101244903003
## --------------------_-------------_-----------------------
```

```
generate_box_plot(
  data = sample_data,
  y_col = "number_of_episodes",
  y_label = "Number of episodes",
  title = "Box plot of the number of episodes of TV series",
  ylim = c(0, 100)
);
```

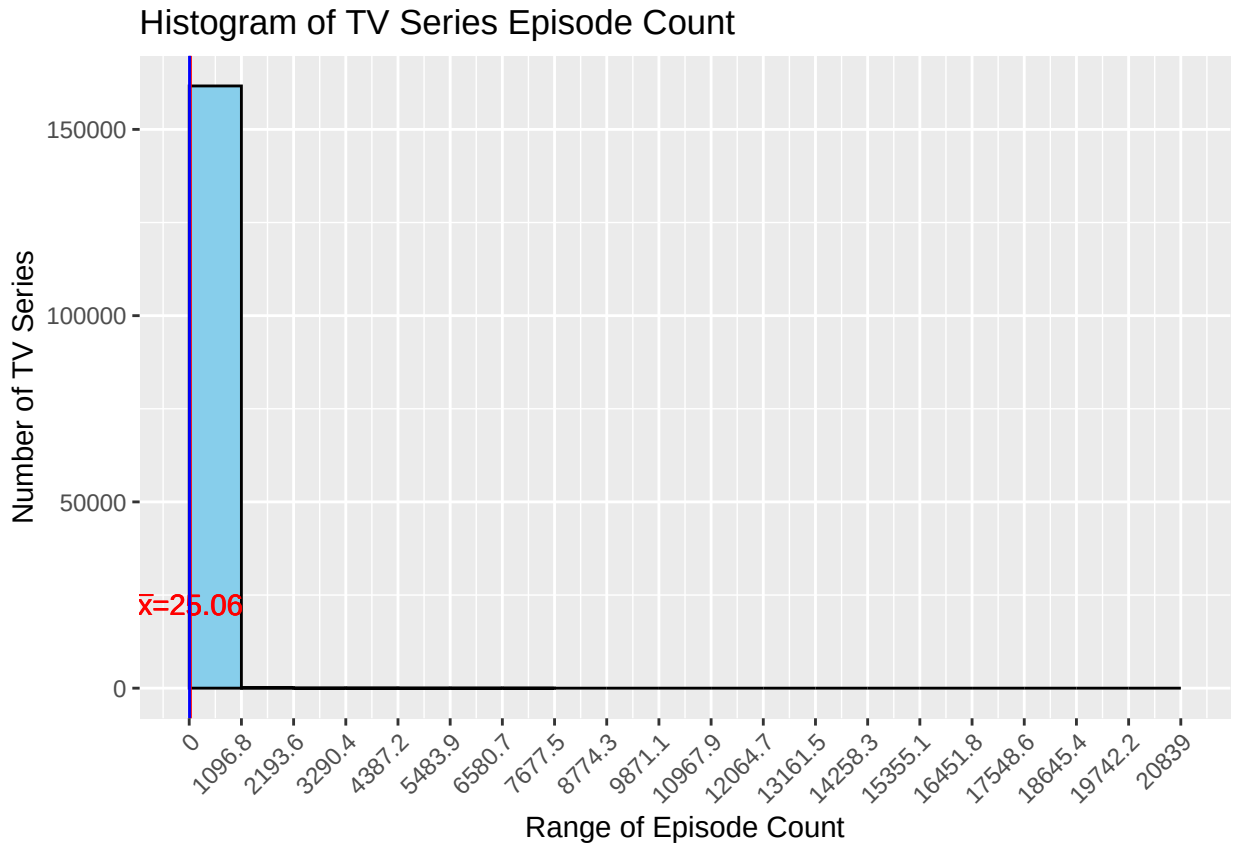


Inference:

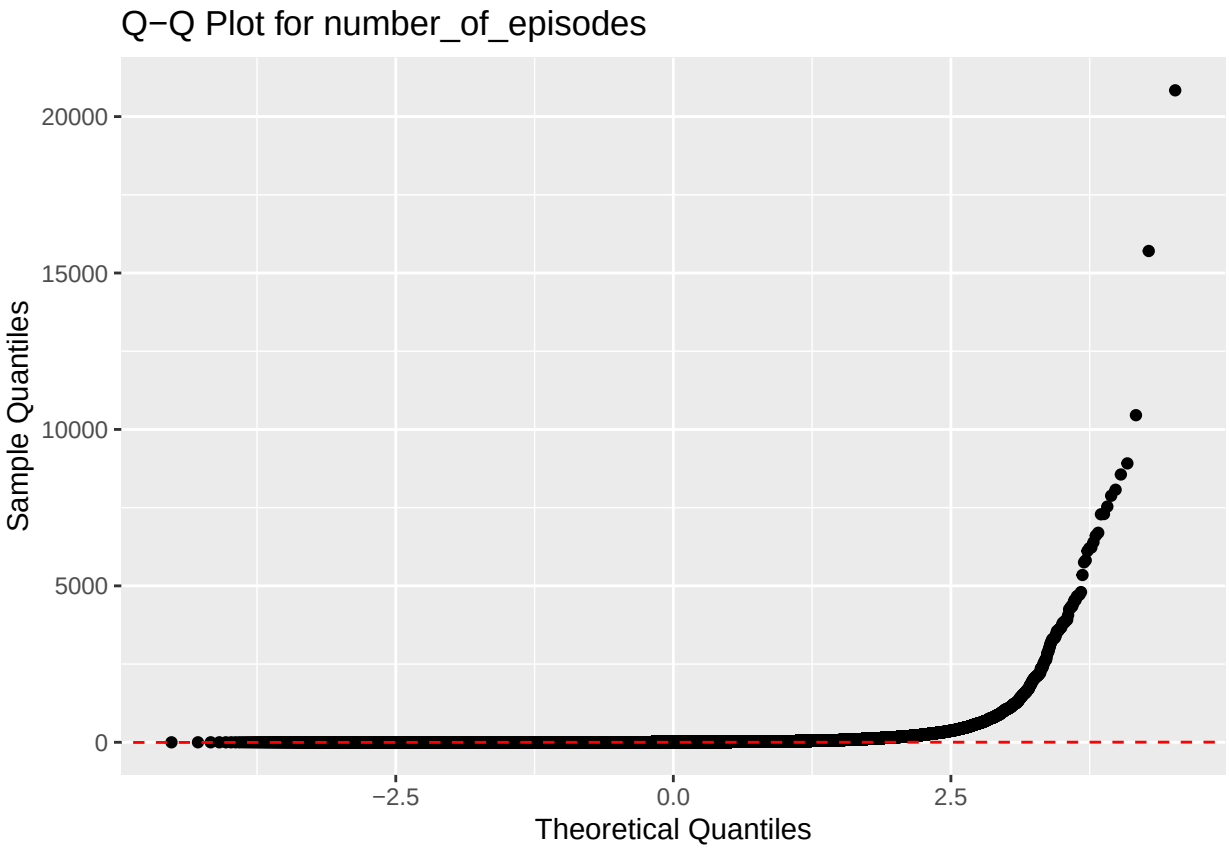
The minimum and maximum values being 0 and 20839 indicate a wide range, suggesting significant variability in the number of episodes. The mean of 25.06 suggests average number of episodes but the presence of high outliers seem to influence the value. Median of 6 is considerably lower than the mean, suggesting positively skewed distribution due to the influence of high values. Skewness of 54.38 indicates significant positive skewness. The 10% trimmed mean of 10.51, calculated by removing extreme values, is lower than the mean and likely reflects a more representative average in the presence of outliers.

```
#HISTOGRAM
plot_histogram(
  data = sample_data,
  column_name = "number_of_episodes",
  x_label = "Range of Episode Count",
  y_label = "Number of TV Series",
  title = "Histogram of TV Series Episode Count",
```

```
  type = 0,
  fillColor = "skyblue")
```



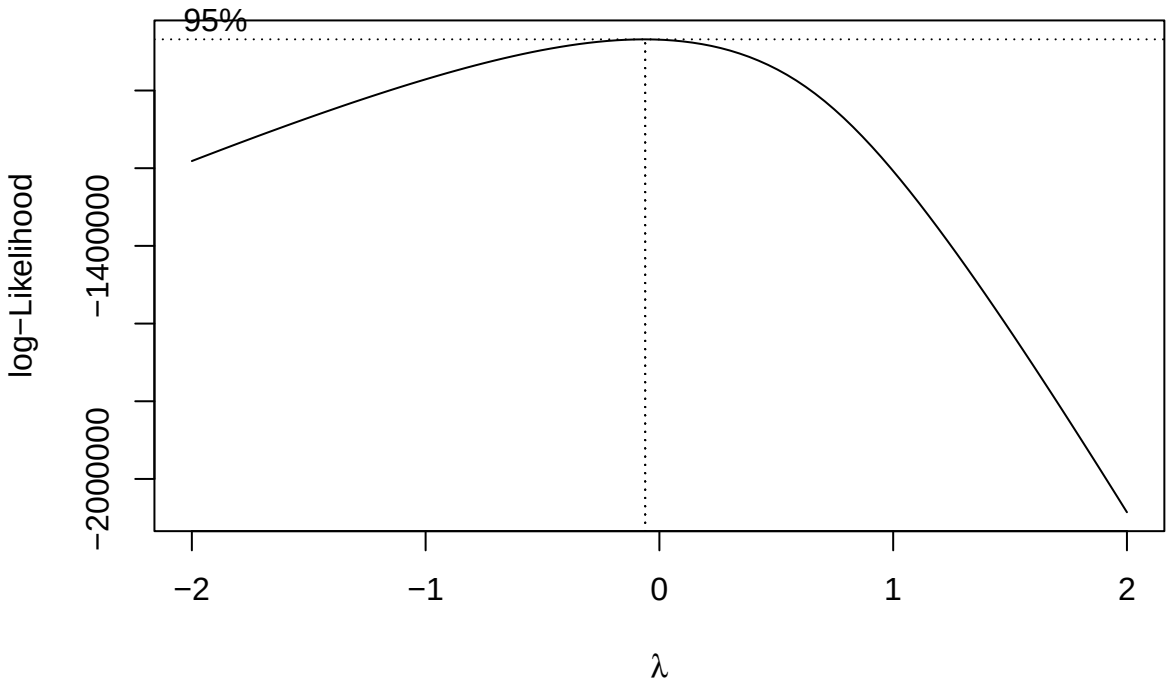

```
#QQ PLOT
plot_qq(data = sample_data, column_name = "number_of_episodes")
```



As further evident from the histogram and qq-plot, the number_of_episodes column is highly left skewed and non-normal and therefore, we can apply the box-cox transformation to make it closer to normal.

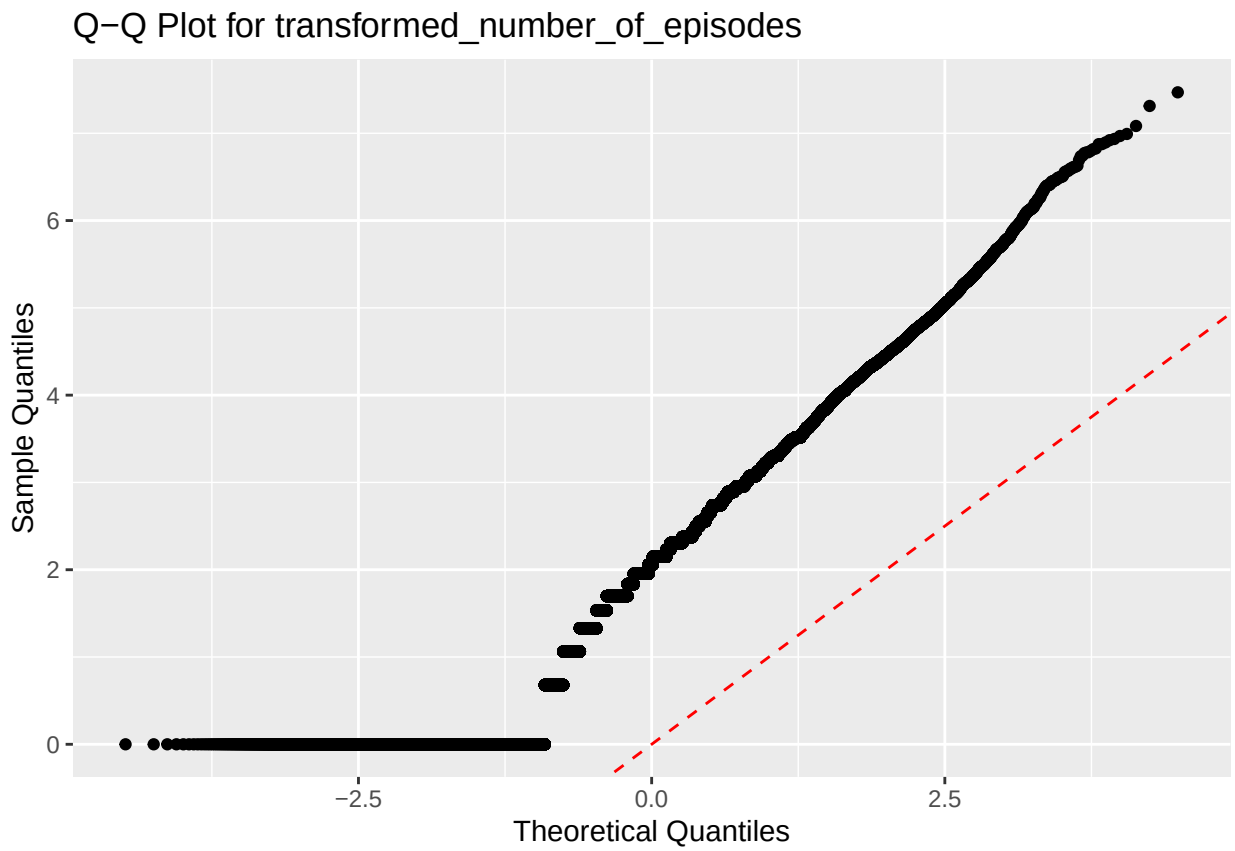
BOXCOX Transformation:

```
#BOX-COX TRANSFORMATION
transformed_col_df <- apply_boxcox_transformation(sample_data, "number_of_episodes")
```




```
## Lambda for optimal transformation: -0.06060606
```

```
plot_qq(data = transformed_col_df, column_name = "transformed_number_of_episodes")
```



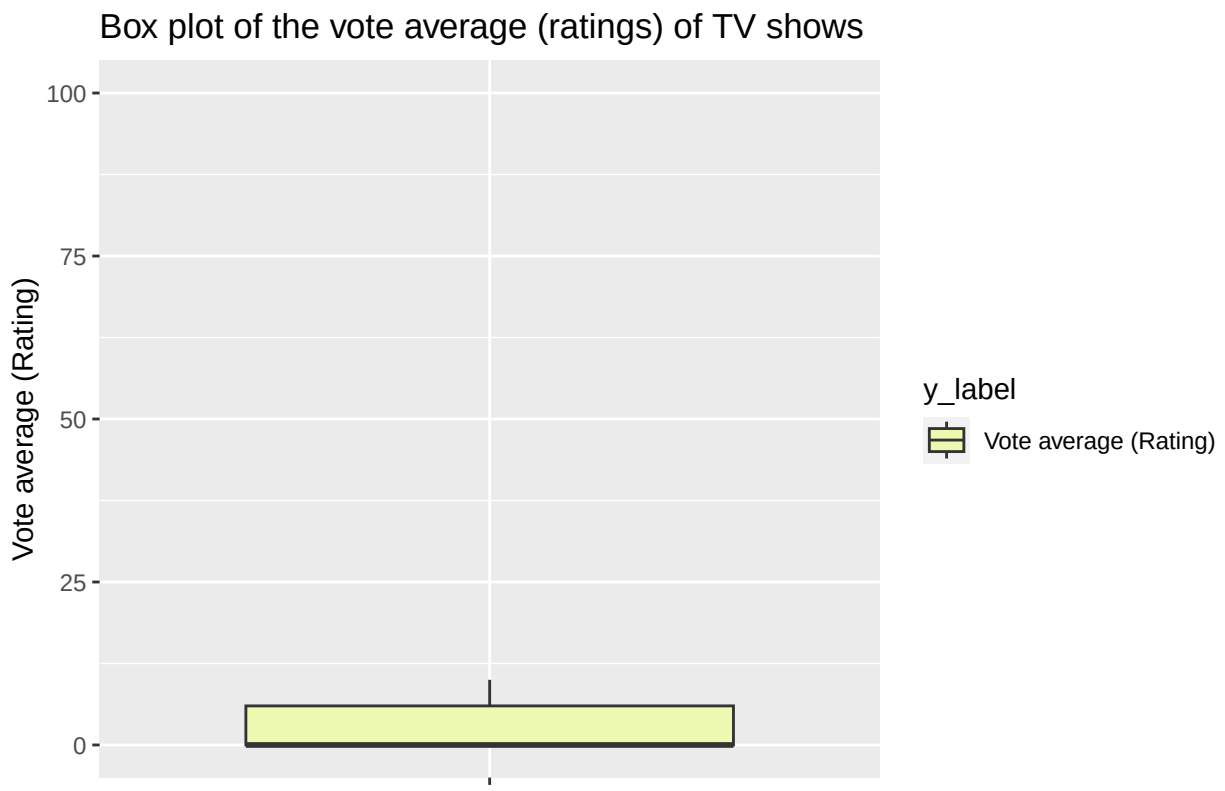
As we can see from quantile plot, the data for number_of_episodes is now much closer to a normal distribution. Further preprocessing of the variable (example taking >0 or >1 or exluding outliers) can be done depending upon the test required.

Statistics on Vote Average

```
sample_data <- transformed_col_df

#VOTE AVERAGE
col_stats(sample_data,"vote_average")
```

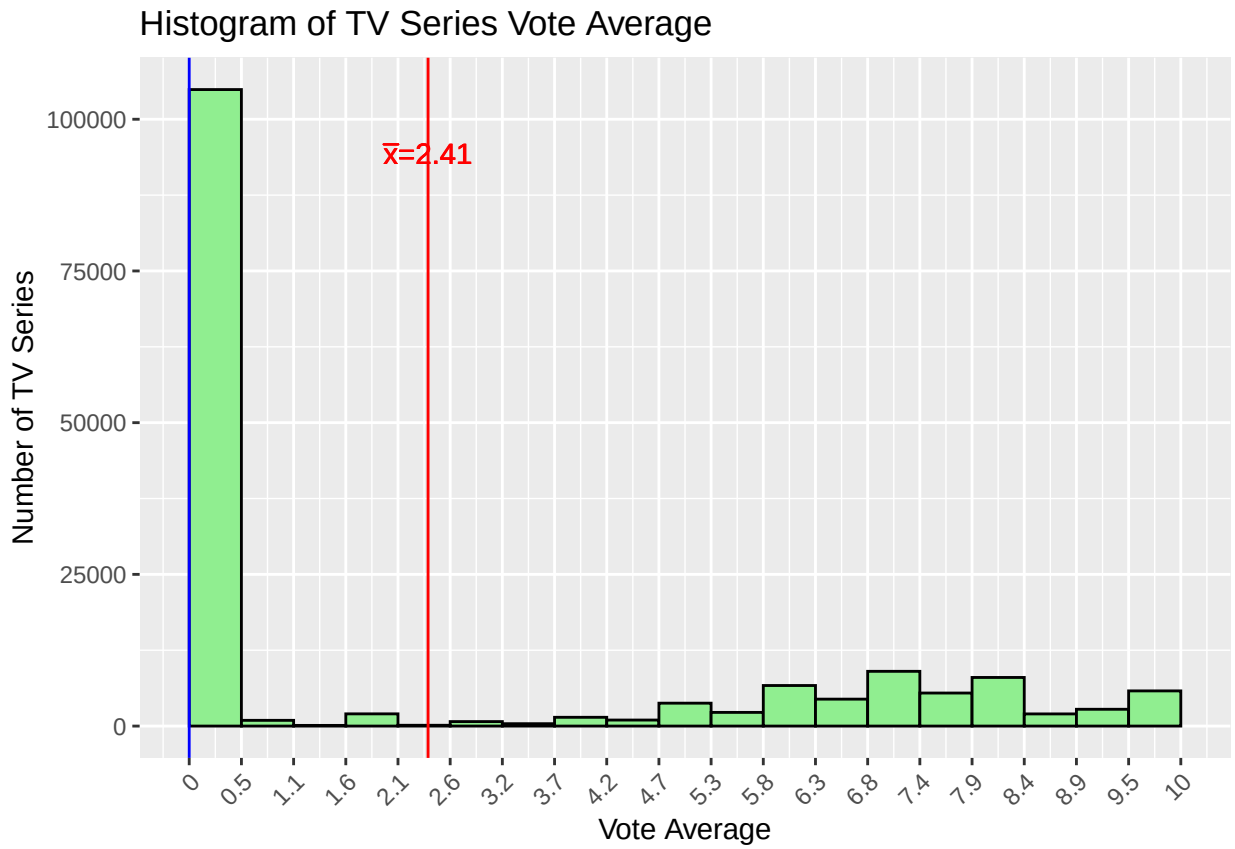
```
## Table for  vote_average
##
## Numeric Summary:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.000   0.000   2.409   6.000  10.000
##
## Coefficient of Variance= 1.4447649276922
## Variance= 12.1113908075565
## Skewness= 0.913651230814089
## IQR= 6
## 10% Trimmed Mean 1.88791270233535
## --------------------_-------------_-----------------------
```

```
generate_box_plot(
  data = sample_data,
  y_col = "vote_average",
  y_label = "Vote average (Rating)",
  title = "Box plot of the vote average (ratings) of TV shows",
  ylim = c(0, 100)
)
```



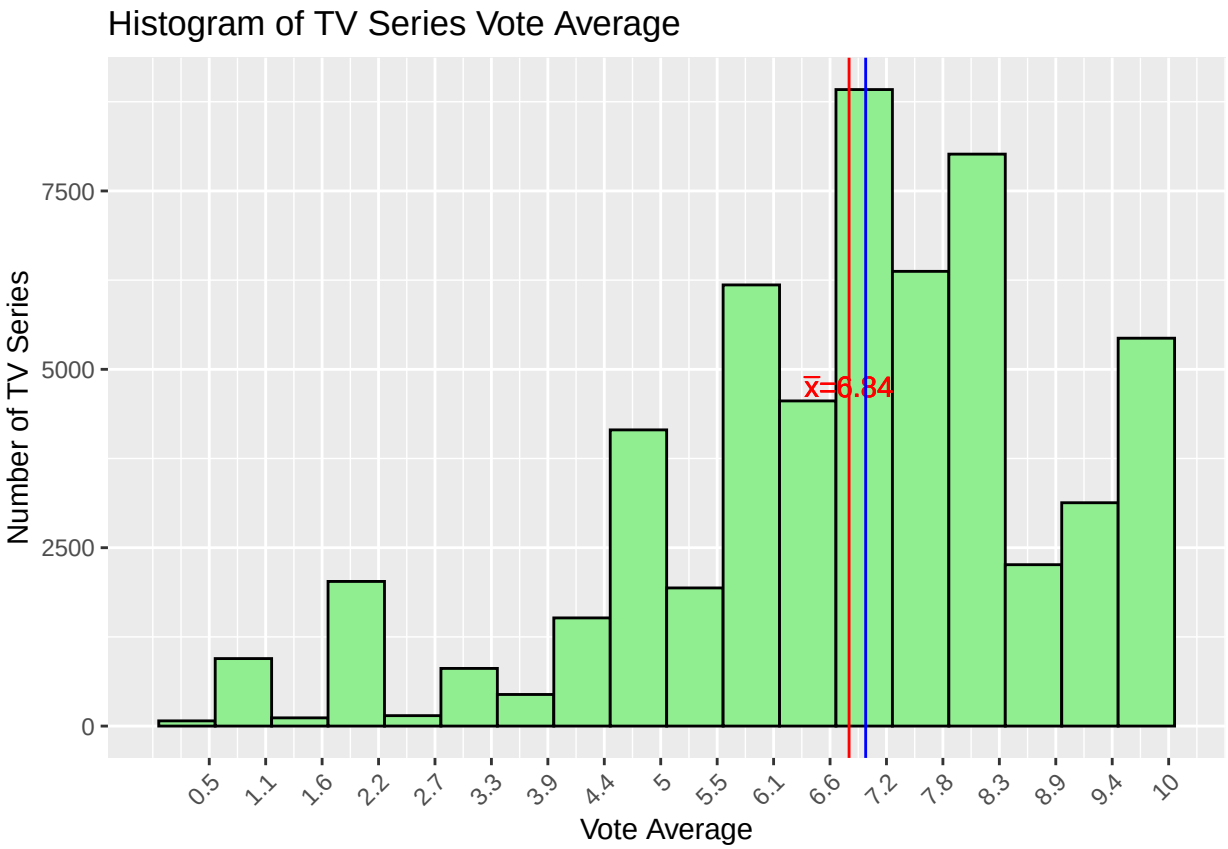
The minimum value of 0 and maximum of 10 indicate that vote_average represents ratings on a scale of 0 to 10. The mean of 2.409 is notably lower than the median of 0.000, suggesting a right-skewed distribution with potential outliers pulling the mean upwards. The median of 0 suggests that a considerable portion of the dataset consists of values at or close to zero. A skewness value of 0.914 indicates a moderately positive skewness, suggesting a tail on the right side of the distribution. The IQR of 6 demonstrates the range of the middle 50% of the data, which spans from 0 to 6. The 10% trimmed mean of 1.888, calculated by removing extreme values, is lower than the mean and might offer a more representative average in the presence of outliers.

```
#HISTOGRAM
plot_histogram(
  data = sample_data,
  column_name = "vote_average",
  x_label = "Vote Average",
  y_label = "Number of TV Series",
  title = "Histogram of TV Series Vote Average",
  type = 0,
  fillColor = "light green")
```

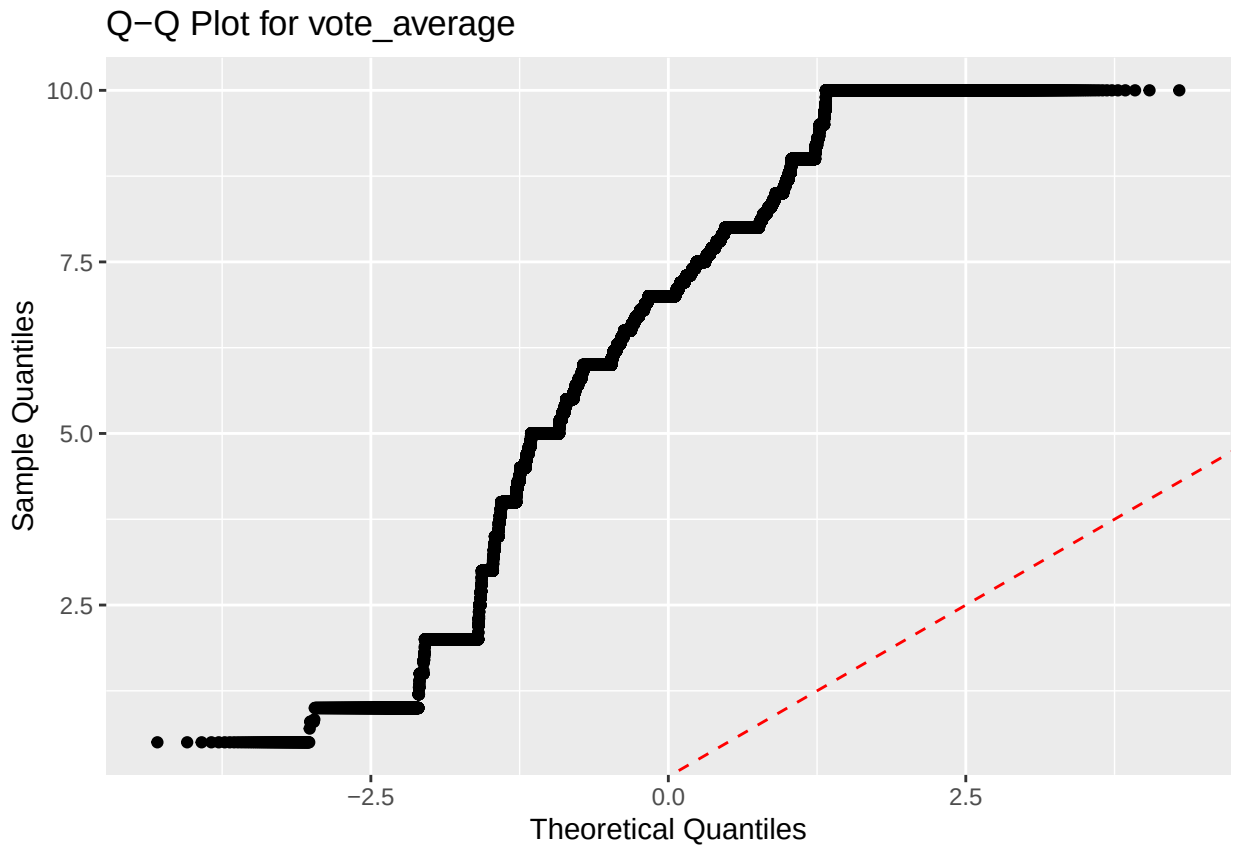


From the histogram and summary statistics it is evident that majority of the data has a vote_average value of 0 which seems like a null value and not part of the metric for rating. Therefore, assuming that vote_average (or the rating) ranges from 1 to 10. We will drop the rows with 0 as the vote_average.

```
sample_data <- subset(sample_data, vote_average > 0)

#HISTOGRAM
plot_histogram(
  data = sample_data,
  column_name = "vote_average",
  x_label = "Vote Average",
  y_label = "Number of TV Series",
  title = "Histogram of TV Series Vote Average",
  type = 0,
  fillColor = "light green")
```



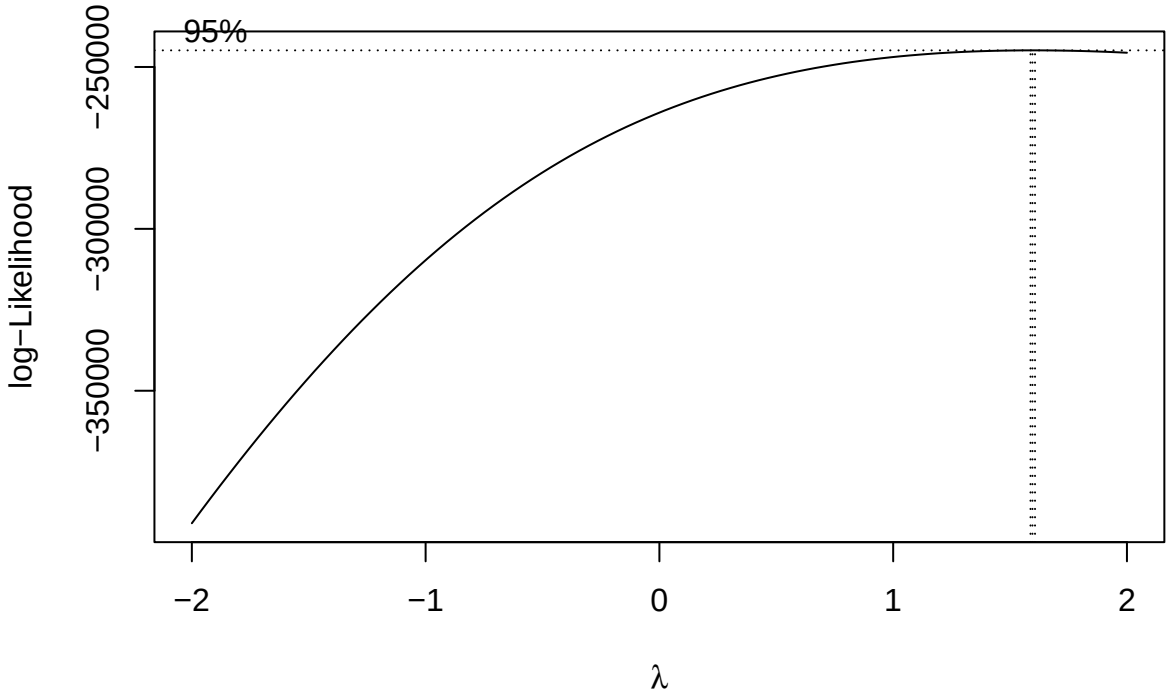

```
#QQ PLOT
plot_qq(data = sample_data, column_name = "vote_average")
```



The histogram shows negative skewness and the quantile plot doesn't show the distribution to be normal, therefore, we can apply the boxcox transformation to normalize the distribution of vote_average

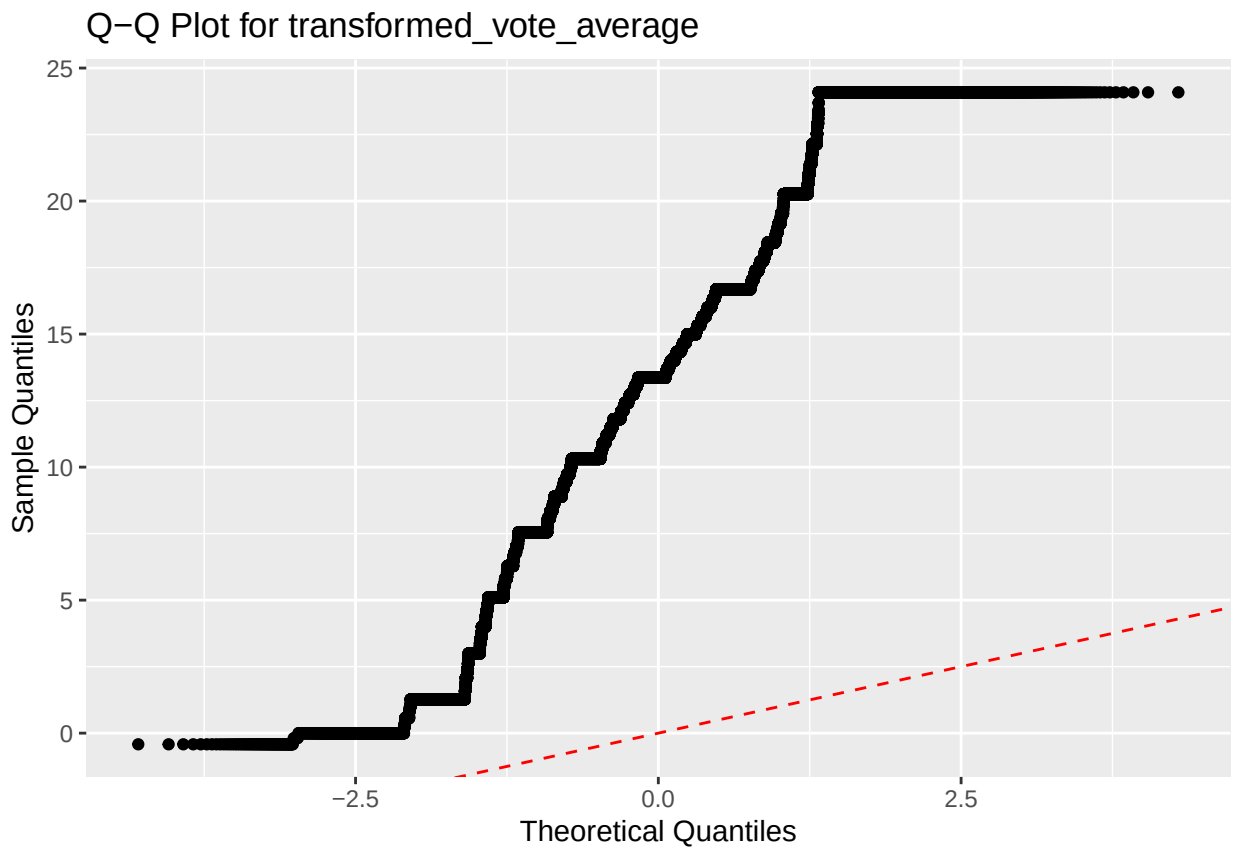
BOXCOX TRANSFORMATION

```
#BOX-COX TRANSFORMATION
transformed_col_df <- apply_boxcox_transformation(sample_data, "vote_average")
```

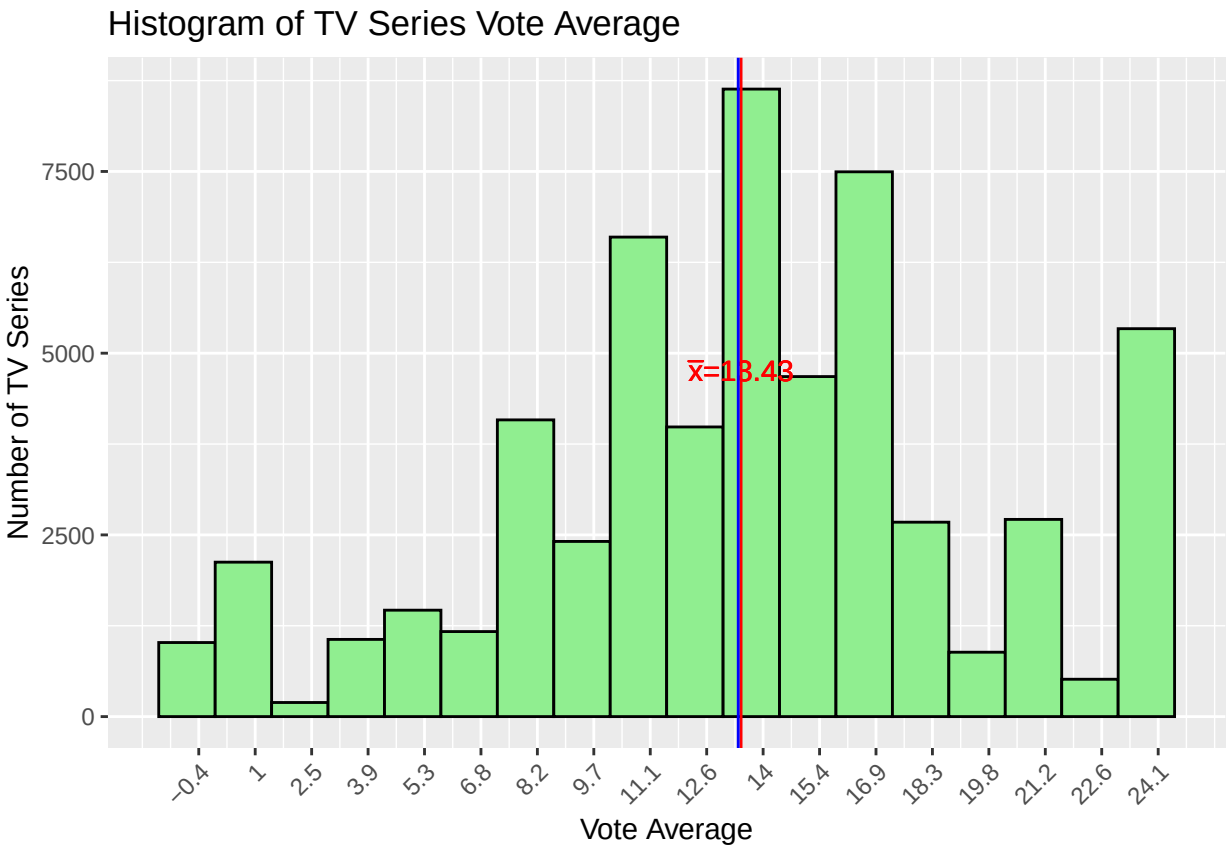



```
## Lambda for optimal transformation: 1.59596
```

```
plot_qq(data = transformed_col_df, column_name = "transformed_vote_average")
```




```
sample_data <- transformed_col_df

#HISTOGRAM AFTER TRANSFORMATION
plot_histogram(
  data = sample_data,
  column_name = "transformed_vote_average",
  x_label = "Vote Average",
  y_label = "Number of TV Series",
  title = "Histogram of TV Series Vote Average",
  type = 0,
  fillColor = "light green")
```



The histogram looks approximately normal, and therefore vote_average can be used a metric for analysing the TV shows data in combination with other variables.

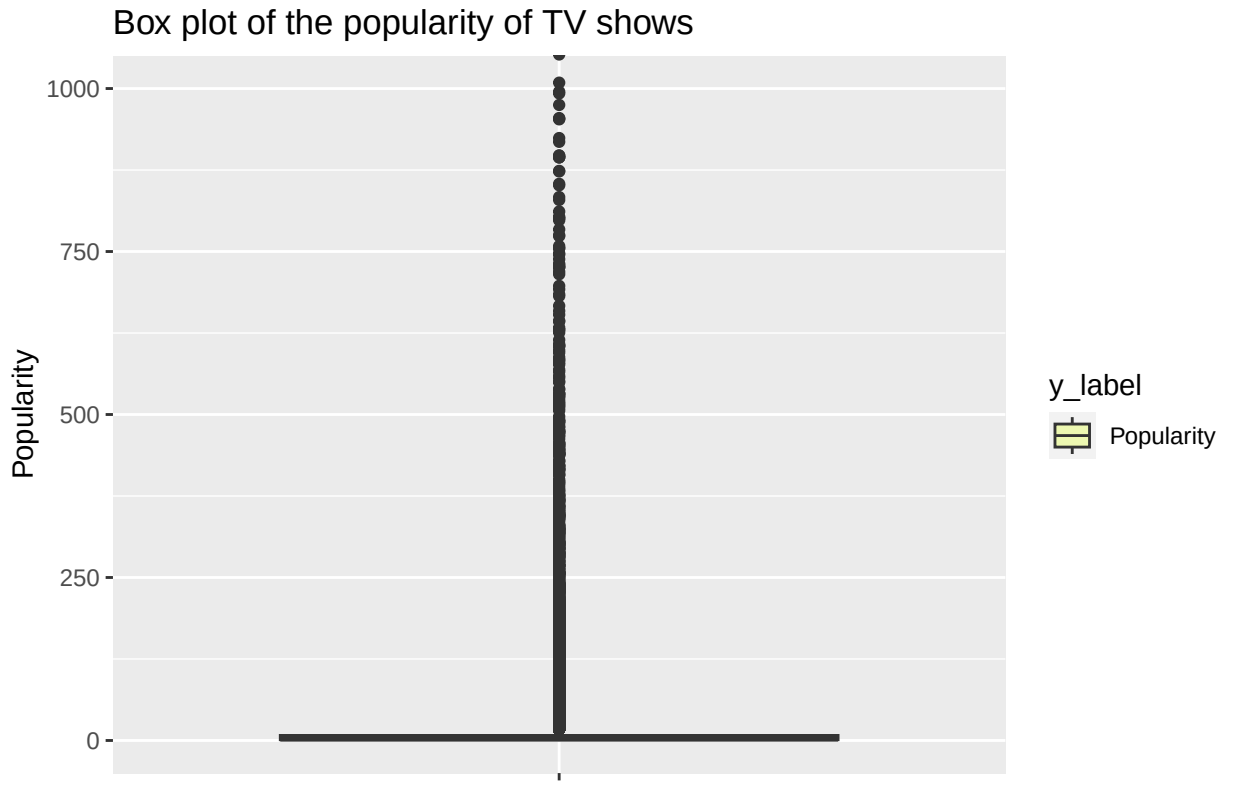
Statistics on Popularity

```
#POPULARITY
col_stats(sample_data,"popularity")
```

```
## Table for  popularity
##
## Numeric Summary:
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##    0.000    1.200    2.726   13.418    7.731 3707.008
##
## Coefficient of Variance= 4.86925444985702
## Variance= 4268.84547884508
## Skewness= 23.2561529508164
## IQR= 6.531
## 10% Trimmed Mean 4.6551524992878
## ____________________-_____________-_______________________
```

```
generate_box_plot(
  data = sample_data,
  y_col = "popularity",
```

```
  y_label = "Popularity",
  title = "Box plot of the popularity of TV shows",
  ylim = c(0, 1000)
)
```



The boxplot shows a heavily right skewed distribution with most of the outliers above the upper limit. The summary statistics indicate that a few items or instances have very high popularity scores (example the max value - 3707), which significantly impact the mean value. The majority of items might have lower popularity scores, while some outliers possess extremely high scores, widening the range (high variance) and skewing the distribution to the right.

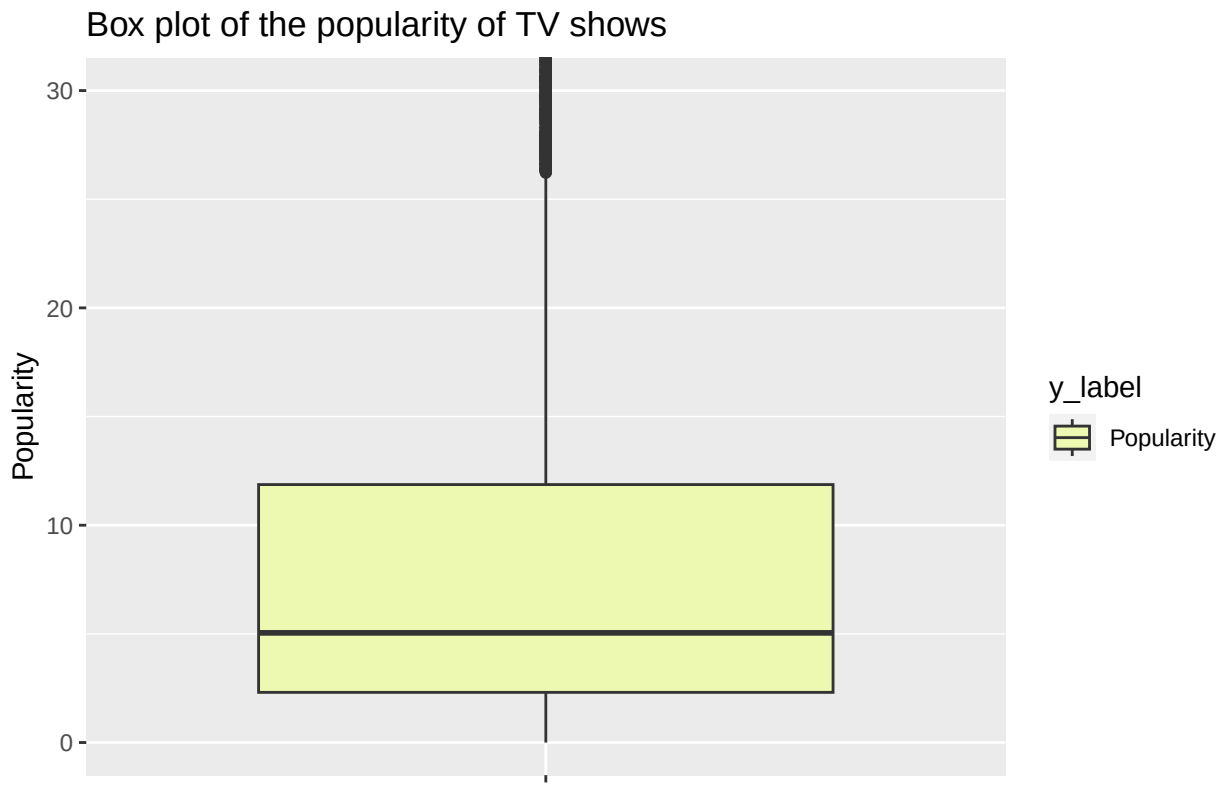
To be able to use popularity for analyses, we have to preprocess the attribute for outliers.

```
data <- na.omit(sample_data)

Q1 <- quantile(data$popularity, 0.25)
Q3 <- quantile(data$popularity, 0.75)
IQR <- Q3 - Q1

lower_limit <- Q1 - 1.5 * IQR
upper_limit <- Q3 + 1.5 * IQR

data <- subset(data, popularity > lower_limit & popularity < upper_limit)

data <- data.frame(data)
```

```r
col_stats(data,"popularity")
```

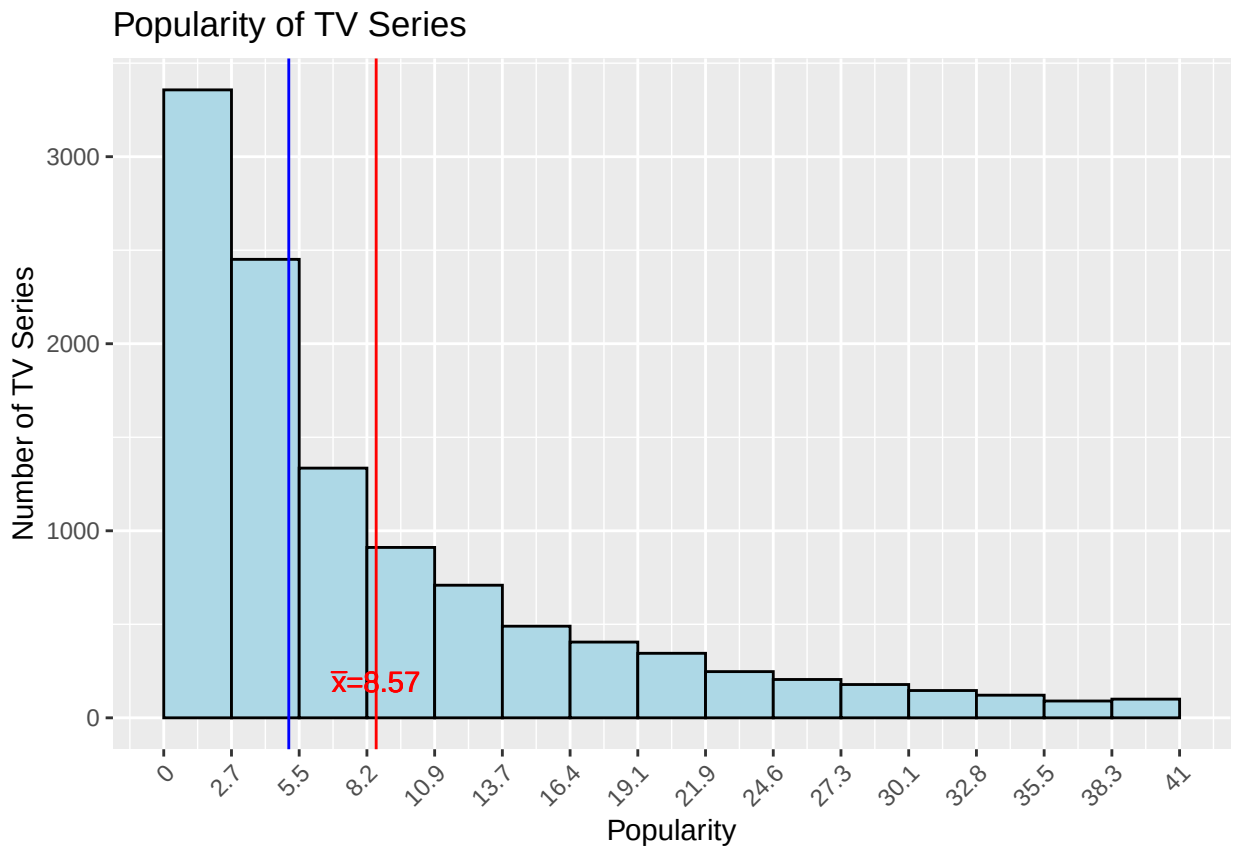
```
## Table for  popularity
##
## Numeric Summary:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   2.309   5.046   8.574  11.870  41.002
##
## Coefficient of Variance= 1.01961857412437
## Variance= 76.43076775096
## Skewness= 1.58040284437901
## IQR= 9.5615
## 10% Trimmed Mean 6.9365637962128
## ____________________-____________-_______________________
```

```r
generate_box_plot(
  data = data,
  y_col = "popularity",
  y_label = "Popularity",
  title = "Box plot of the popularity of TV shows",
  ylim = c(0, 30)
)
```

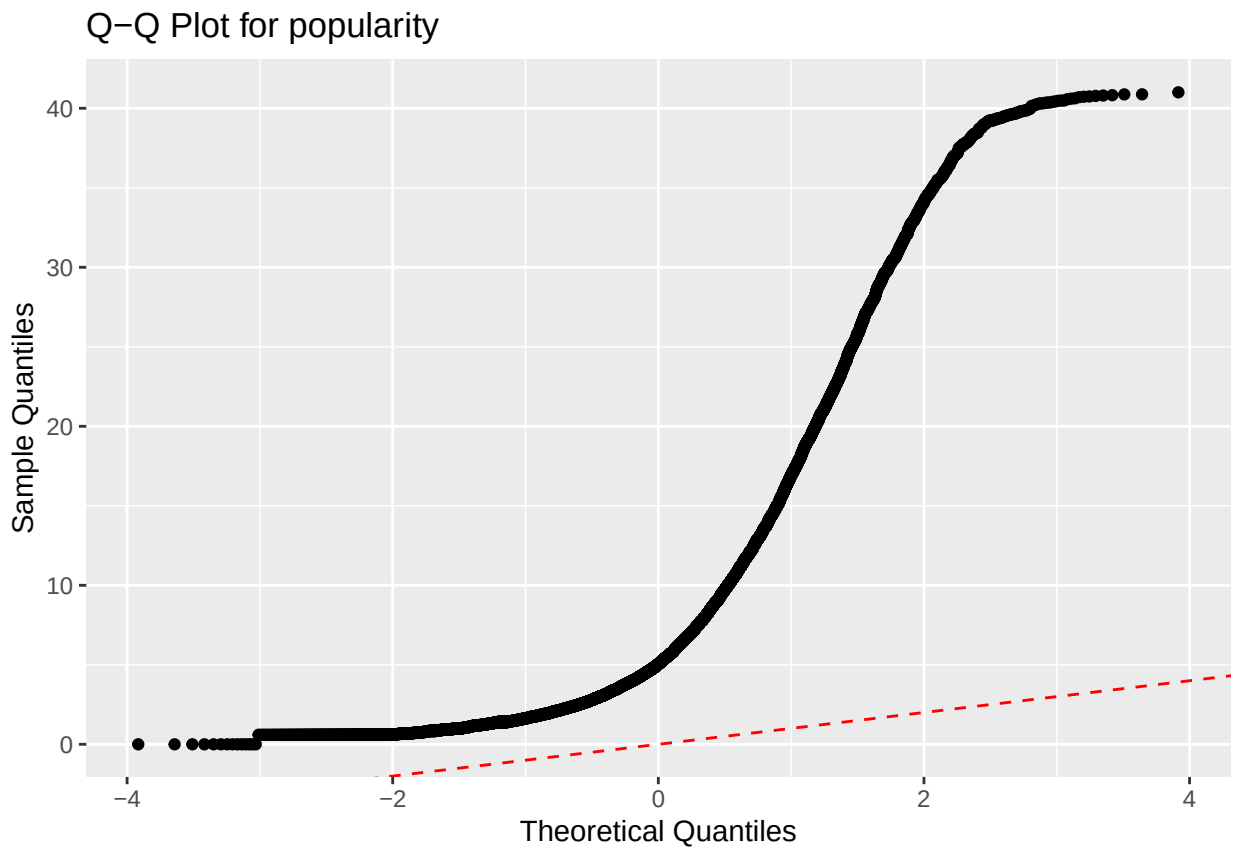


```r
sample_data <- data.frame(data)
```

```r
# HISTOGRAM
plot_histogram(
  data = sample_data,
  column_name = "popularity",
  x_label = "Popularity",
  y_label = "Number of TV Series",
  title = "Popularity of TV Series",
  type = 0,
  fillColor = "light blue")
```




```r
# QQ PLOT
plot_qq(data = sample_data, column_name = "popularity")
```



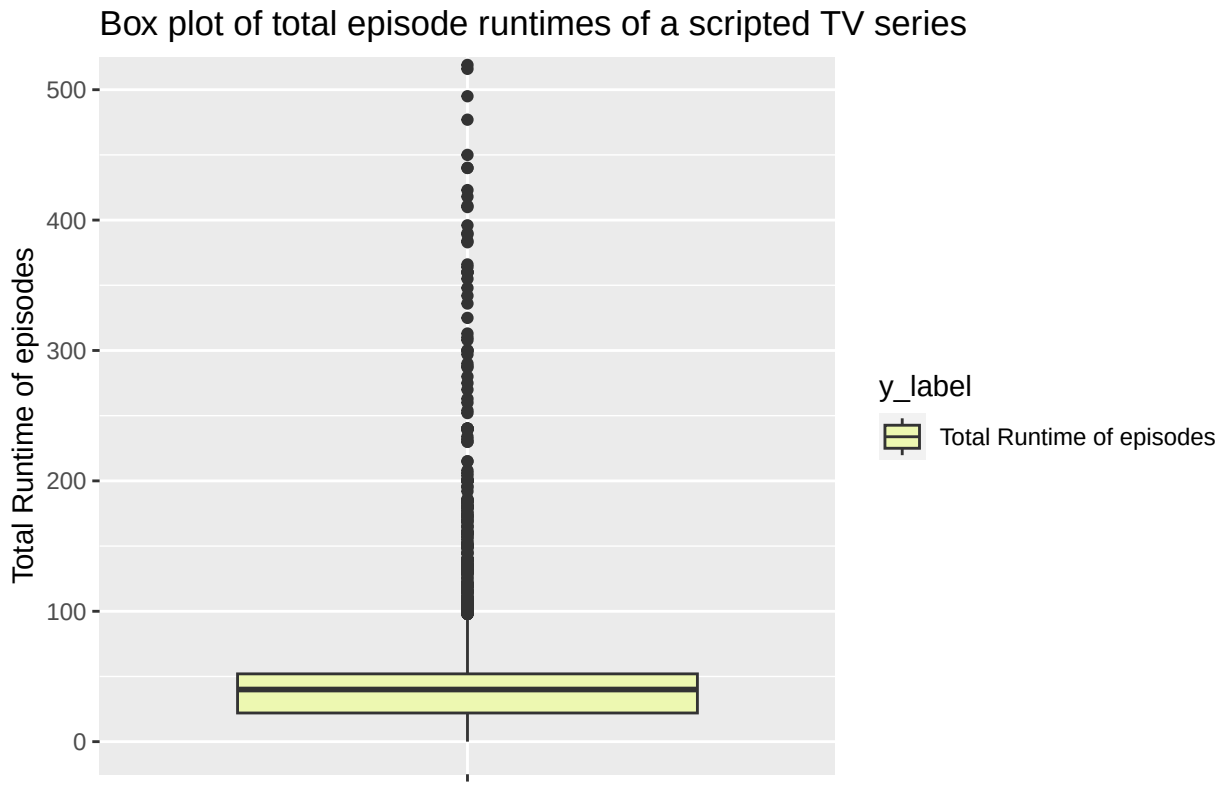
After removing the outliers, the histogram and quantile plots indicate a right skewed distribution, however, applying the box-cox transformation here gives inappropriate values for popularity including negative and null values, therefore we do not apply transformation to popularity data.

Statistics on Episode Run Time

```
#EPISODE RUN-TIME
col_stats(sample_data,"episode_run_time")
```

```
## Table for  episode_run_time
##
## Numeric Summary:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00   22.00   40.00   41.54   52.00 2280.00
##
## Coefficient of Variance= 1.27458161763159
## Variance= 2803.91725185057
## Skewness= 16.5139340846257
## IQR= 30
## 10% Trimmed Mean 36.2222723174031
## --------------------_-------------_-----------------------
```

```
generate_box_plot(
  data = sample_data,
  y_col = "episode_run_time",
  y_label = "Total Runtime of episodes",
  title = "Box plot of total episode runtimes of a scripted TV series",
  ylim = c(0, 500)
)
```



Due to the large variance, it is evident that some of the values in episode_run_time has been given in seconds and some in minutes. We therefore have to preprocess the attribute before analysis. Therefore, a

This function iterates through the "episode_run_time" column of a data frame and performs the following steps: 1. Remove rows with missing values in "episode_run_time". 2. Update values outside the specified range directly. 3. Update values by dividing by "number_of_episodes" where applicable. 4. Update values by dividing by 60 where applicable. 5. Assign a default value of -1 for values that don't meet the conditions. 6. Remove rows with the default value. 7. Discard entire rows for values that don't meet the conditions.

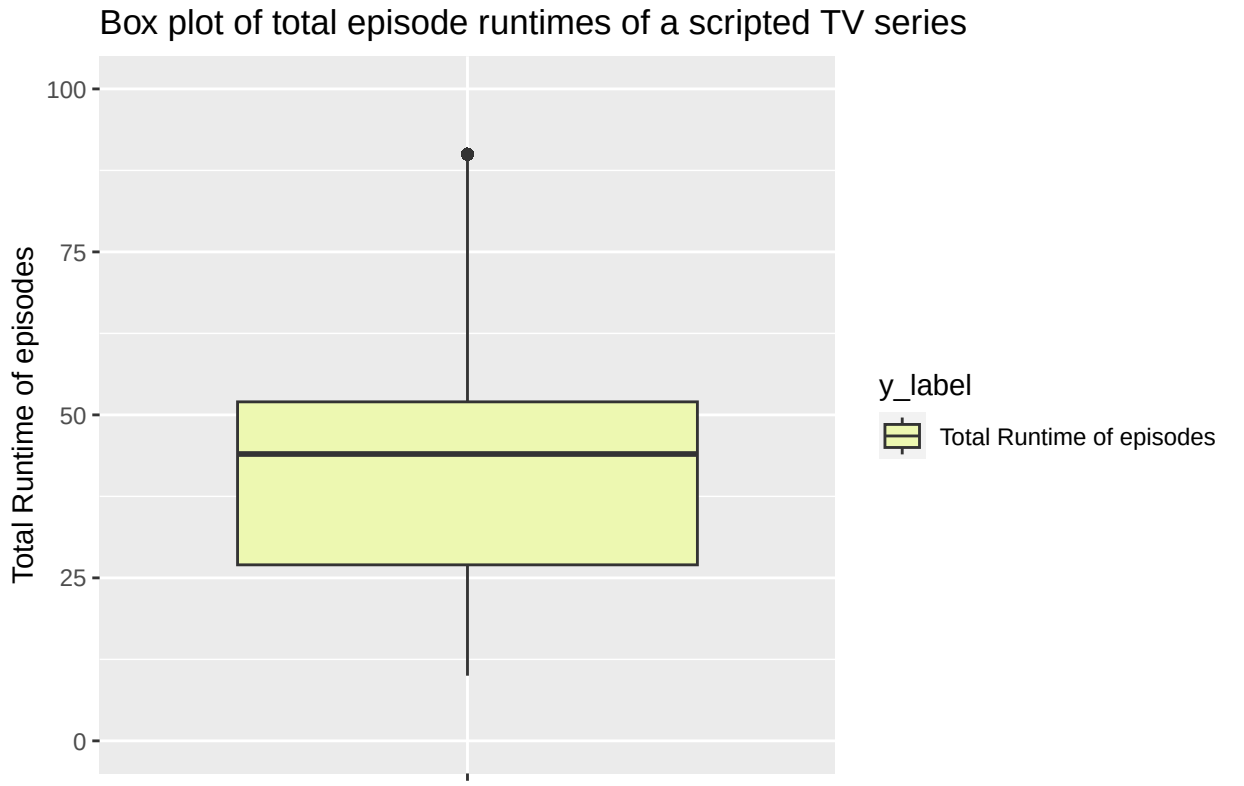
We are going to take the limit for episode length in minutes to be a minimum of 10 minutes and a maximum of 90 minutes.

```
sample_data <- process_episode_data(sample_data,10,90)

col_stats(sample_data,"episode_run_time")
```

```
## Table for  episode_run_time
##
```

```
## Numeric Summary:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   10.00   27.00   44.00   42.24   52.00   90.00
##
## Coefficient of Variance= 0.416853115043644
## Variance= 310.041952359884
## Skewness= 0.554700174510282
## IQR= 25
## 10% Trimmed Mean 41.1773264188995
## --------------------_-------------_-----------------------
```

```r
generate_box_plot(
  data = sample_data,
  y_col = "episode_run_time",
  y_label = "Total Runtime of episodes",
  title = "Box plot of total episode runtimes of a scripted TV series",
  ylim = c(0, 100)
)
```



Inference:

The mean of 42.24 minutes represents the average episode run time. The median of 44 minutes suggests that half of the episodes have a duration of less than equal to 44 indicating a moderately symmetric distribution. Q1 = 27 and Q3 = 52 indicates that middle 50% of the episode run times lie between these values. With a coefficient of variation of 0.42, the standard deviation relative to the mean is relatively low, indicating

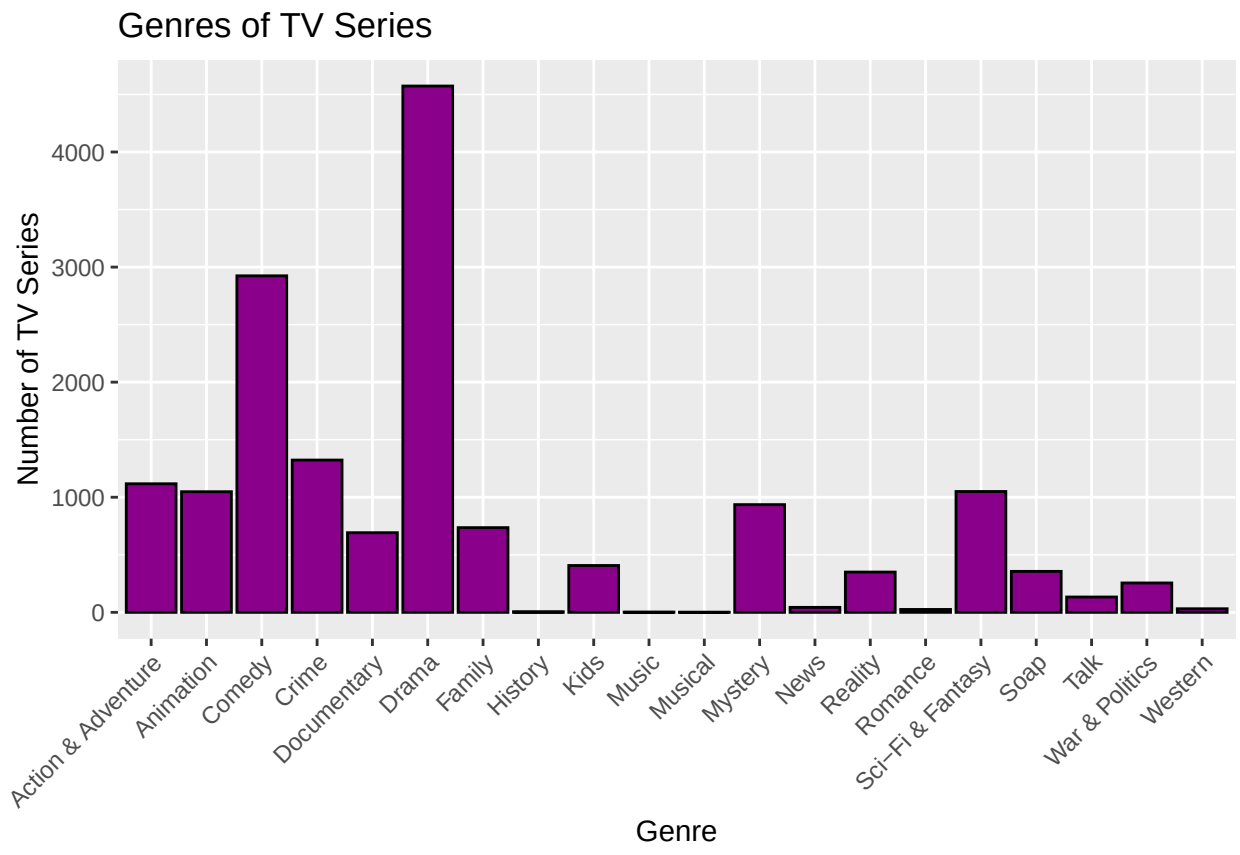
moderate variability around the mean episode run time. A skewness value of 0.55 indicates a moderate positive skewness, suggesting a slightly longer tail on the right side of the distribution compared to a perfectly symmetric distribution. Overall, the summary statistics and the box plot suggest a distribution of episode run times that is moderately skewed to the right, but nearly normal with most episodes falling within the 27 to 52-minute range. While the mean and median are relatively close, the positive skewness indicates that some episodes might have durations longer than the median, contributing to the longer tail on the right side of the distribution.

Exploring the categorical variables and their usability

1. Exploring genres in TV shows data

Since the genres column has multiple categories mentioned on same observations, we can expand the distribution based on comma separated values. This is done internally in the function call for plotting by calling the expand_nested_data function (defined in the utils.R file)

```
#GENRES BAR PLOT
plot_histogram(
  data = sample_data,
  column_name = "genres",
  x_label = "Genre",
  y_label = "Number of TV Series",
  title = "Genres of TV Series",
  type = 1,
  fillColor = "dark magenta")
```



The bar plot of different genres has been plotted above and it is evident from the plot that the most common genre among TV shows is “Drama”, followed by “Comedy”. In contrast, genres like “History”, “Music”, “Musical”, etc. are the least common.

```
#ABSOLUTE FREQUENCY TABLE
df_expanded_genres <- expand_nested_data(sample_data,"genres")
colnames(df_expanded_genres)[colnames(df_expanded_genres) == "cat_col"] <- "genre"

for (i in 1:nrow(df_expanded_genres)) {
  print(paste(df_expanded_genres[i, ], sep = " "))
}
```

```
## [1] "Drama" "4573"
## [1] "Comedy" "2924"
## [1] "Crime" "1323"
## [1] "Action & Adventure" "1117"
## [1] "Sci-Fi & Fantasy" "1050"
## [1] "Animation" "1048"
## [1] "Mystery" "936"
## [1] "Family" "736"
## [1] "Documentary" "692"
## [1] "Kids" "407"
## [1] "Soap" "356"
## [1] "Reality" "350"
## [1] "War & Politics" "255"
## [1] "Talk" "133"
## [1] "News" "43"
## [1] "Western" "31"
## [1] "Romance" "25"
## [1] "History" "5"
## [1] "Music" "2"
## [1] "Musical" "1"
```

```
#RELATIVE FREQUENCY TABLE
total_frequency <- sum(df_expanded_genres$frequency)

for (i in 1:nrow(df_expanded_genres)) {
  genre <- df_expanded_genres[i, "genre"]
  freq <- df_expanded_genres[i, "frequency"]
  percentage <- (freq / total_frequency) * 100

  # Print genre and relative frequency
  print(paste("Genre:", genre, "- RF :", round(percentage, 3), "%"))
}
```

```
## [1] "Genre: Drama - RF : 28.569 %"
## [1] "Genre: Comedy - RF : 18.267 %"
## [1] "Genre: Crime - RF : 8.265 %"
## [1] "Genre: Action & Adventure - RF : 6.978 %"
## [1] "Genre: Sci-Fi & Fantasy - RF : 6.56 %"
## [1] "Genre: Animation - RF : 6.547 %"
## [1] "Genre: Mystery - RF : 5.847 %"
## [1] "Genre: Family - RF : 4.598 %"
```

```
## [1] "Genre: Documentary - RF : 4.323 %"
## [1] "Genre: Kids - RF : 2.543 %"
## [1] "Genre: Soap - RF : 2.224 %"
## [1] "Genre: Reality - RF : 2.187 %"
## [1] "Genre: War & Politics - RF : 1.593 %"
## [1] "Genre: Talk - RF : 0.831 %"
## [1] "Genre: News - RF : 0.269 %"
## [1] "Genre: Western - RF : 0.194 %"
## [1] "Genre: Romance - RF : 0.156 %"
## [1] "Genre: History - RF : 0.031 %"
## [1] "Genre: Music - RF : 0.012 %"
## [1] "Genre: Musical - RF : 0.006 %"
```

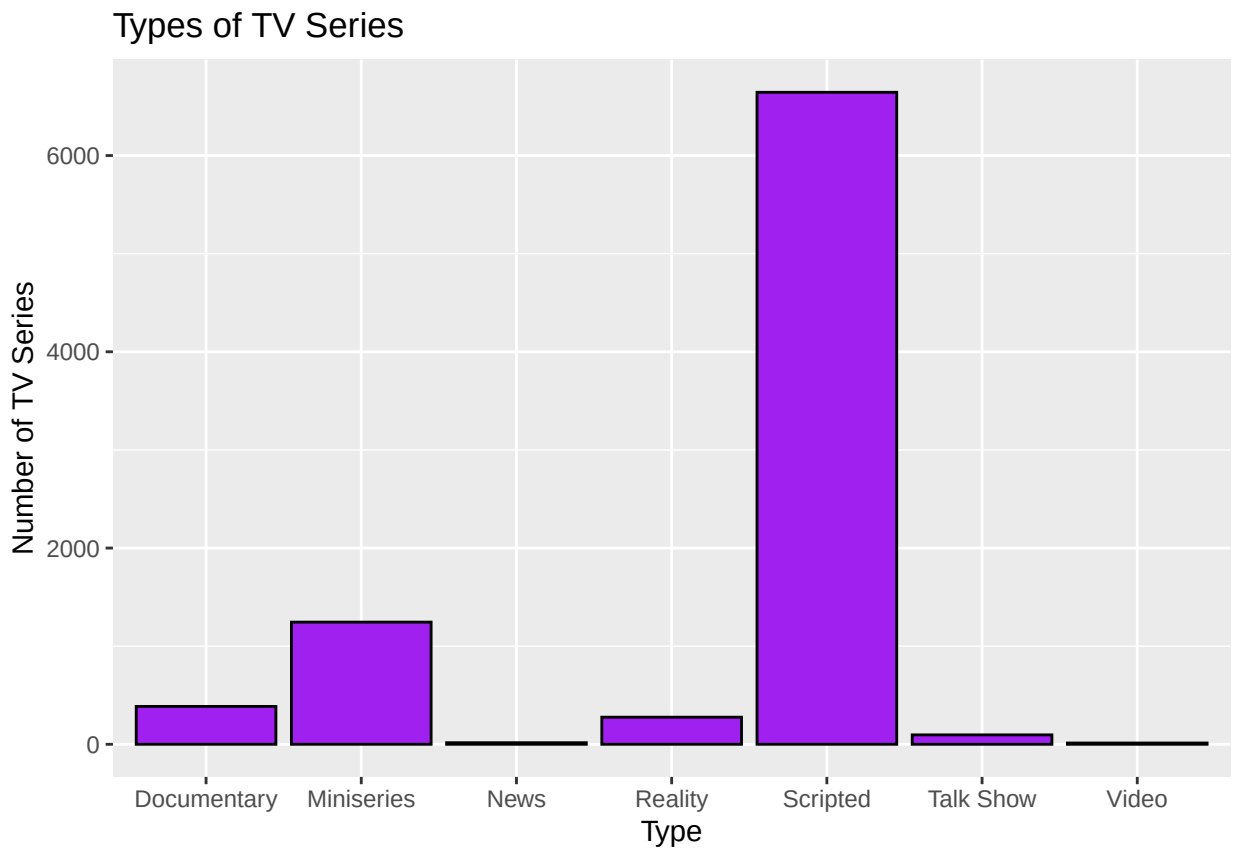
2. Exploring "type" variable

```
print(table(sample_data$type))
```

```
##
## Documentary  Miniseries        News     Reality    Scripted   Talk Show
##         386        1245          16         276        6644          96
##       Video
##          13
```

Since the categories are not in comma separated, repeating format, there is no requirement of expansion, we can therefore get the frequency tables and bar plot directly

```
#TYPE BAR PLOT
plot_histogram(
  data = sample_data,
  column_name = "type",
  x_label = "Type",
  y_label = "Number of TV Series",
  title = "Types of TV Series",
  type = 2,
  fillColor = "purple")
```



From the barplot of "type" categorical variable, we can observe that most of the TV shows listed in the data are of "Scripted" type. It is followed by "Miniseries", "Documentary" and "Reality". "News" and "Video" are the least common type categories.

Frequency tables for categorical variable type:

```
#ABSOLUTE FREQUENCY TABLE
freq_table_type <- sort(table(sample_data$type), decreasing = TRUE)
for (val in unique(sample_data$type)) {
  print(paste(val, " ", freq_table_type[val]))
}
```

```
## [1] "Miniseries   1245"
## [1] "Scripted   6644"
## [1] "Documentary   386"
## [1] "Talk Show   96"
## [1] "Reality   276"
## [1] "Video   13"
## [1] "News   16"
```

```
#RELATIVE FREQUENCY TABLE
relative_freq_type <- prop.table(freq_table_type) * 100

for (val in names(relative_freq_type)) {
  print(paste(val, " ", round(relative_freq_type[val], 2), "%"))
}
```

```
## [1] "Scripted   76.58 %"
## [1] "Miniseries   14.35 %"
## [1] "Documentary   4.45 %"
## [1] "Reality   3.18 %"
## [1] "Talk Show   1.11 %"
## [1] "News   0.18 %"
## [1] "Video   0.15 %"
```

3. Exploring "original_language"

```
print(unique(sample_data$original_language))
```

```
##  [1] "en" "es" "ko" "de" "sv" "it" "ja" "zh" "da" "fr" "pt" "tr" "is" "he" "no"
## [16] "th" "pl" "ru" "nl" "hi" "tl" "fi" "lb" "gl" "uk" "la" "ar" "bg" "hu" "cy"
## [31] "el" "sr" "cs" "cn" "ta" "fa" "ca" "ro" "id" "ur" "sh" "sk" "te" "hr" "bn"
## [46] "si" "et" "ms" "ka" "sq" "lv" "af" "vi" "eu" "nb" "az" "kn" "zu" "lt" "jv"
```

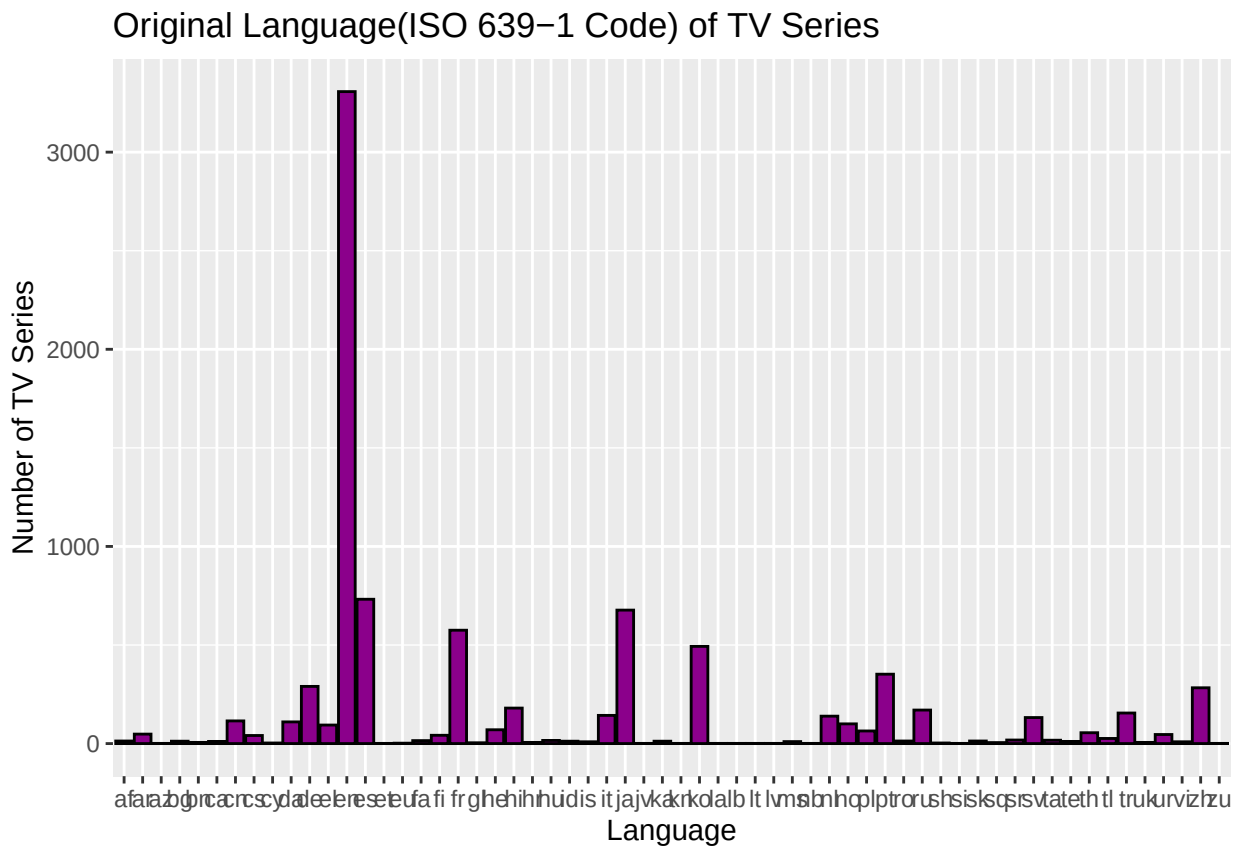
The original languages of the TV shows are in abbreviated format, therefore we can map them to their full versions, as close as possible with information available on the internet:

```
# Abbreviations list
abbreviations <- c("en", "es", "ko", "de", "ja", "sv", "th", "it", "zh", "da", "fr", "pt", "tr", "ru",

# Full names list corresponding to abbreviations
full_names <- c("English", "Spanish", "Korean", "German", "Japanese", "Swedish", "Thai", "Italian", "Ch


language_map <- setNames(full_names, abbreviations)
print(language_map)
```

```
##                en                es                ko                de
##         "English"         "Spanish"          "Korean"          "German"
##                ja                sv                th                it
##        "Japanese"         "Swedish"           "Thai"         "Italian"
##                zh                da                fr                pt
##         "Chinese"          "Danish"          "French"      "Portuguese"
##                tr                ru                is                he
##         "Turkish"         "Russian"       "Icelandic"          "Hebrew"
##                no                pl                nl                hi
##       "Norwegian"          "Polish"          "Dutch"          "Hindi"
##                tl                fi                lb                gl
##         "Tagalog"         "Finnish"   "Luxembourgish"        "Galician"
##                uk                la                ar                cs
##       "Ukrainian"          "Latin"          "Arabic"          "Czech"
##                bg                hu                cy                el
##       "Bulgarian"       "Hungarian"          "Welsh"          "Greek"
##                ta                sr                cn                fa
##          "Tamil"         "Serbian"         "Chinese"         "Persian"
##                bn                ca                ro                id
##         "Bengali"         "Catalan"        "Romanian"      "Indonesian"
##                xx                ur                sk                ms
```

```
##           "Unknown"              "Urdu"           "Slovak"            "Malay"
##                  sh                  af                te                hr
##    "Serbo-Croatian"       "Afrikaans"          "Telugu"       "Croatian"
##                  vi                  si                ga                et
##        "Vietnamese"       "Sinhalese"           "Irish"       "Estonian"
##                  ka                  sq                az                lv
##          "Georgian"         "Albanian"     "Azerbaijani"         "Latvian"
##                  eu                  nb                sl                kn
##           "Basque" "Norwegian Bokmål"       "Slovenian"        "Kannada"
##                  mk                  bs                zu                lt
##        "Macedonian"          "Bosnian"           "Zulu"      "Lithuanian"
##                  jv                  so                pa                ne
##          "Javanese"          "Somali"         "Punjabi"          "Nepali"
```

```
#BAR PLOT
plot_histogram(
  data = sample_data,
  column_name = "original_language",
  x_label = "Language",
  y_label = "Number of TV Series",
  title = "Original Language(ISO 639-1 Code) of TV Series",
  type = 2,
  fillColor = "darkmagenta")
```



From the bar plot of "original_language" of TV shows data, we can observe that majority of the TV shows available in the dataset are in "English", followed by "French" and "Spanish"

```r
#ABSOLUTE FREQUENCY TABLE
freq_table_ol <- sort(table(sample_data$original_language), decreasing = TRUE)
for (val in unique(sample_data$original_language)) {
  print(paste(language_map[[val]], " ", freq_table_ol[val]))
}
```

```
## [1] "English   3307"
## [1] "Spanish   732"
## [1] "Korean   493"
## [1] "German   290"
## [1] "Swedish   132"
## [1] "Italian   143"
## [1] "Japanese   677"
## [1] "Chinese   283"
## [1] "Danish   110"
## [1] "French   575"
## [1] "Portuguese   352"
## [1] "Turkish   155"
## [1] "Icelandic   9"
## [1] "Hebrew   70"
## [1] "Norwegian   100"
## [1] "Thai   55"
## [1] "Polish   64"
## [1] "Russian   170"
## [1] "Dutch   139"
## [1] "Hindi   180"
## [1] "Tagalog   26"
## [1] "Finnish   42"
## [1] "Luxembourgish   1"
## [1] "Galician   4"
## [1] "Ukrainian   6"
## [1] "Latin   1"
## [1] "Arabic   48"
## [1] "Bulgarian   12"
## [1] "Hungarian   16"
## [1] "Welsh   3"
## [1] "Greek   94"
## [1] "Serbian   18"
## [1] "Czech   41"
## [1] "Chinese   115"
## [1] "Tamil   17"
## [1] "Persian   15"
## [1] "Catalan   11"
## [1] "Romanian   13"
## [1] "Indonesian   12"
## [1] "Urdu   46"
## [1] "Serbo-Croatian   3"
## [1] "Slovak   13"
## [1] "Telugu   11"
## [1] "Croatian   6"
## [1] "Bengali   6"
## [1] "Sinhalese   1"
## [1] "Estonian   1"
```

```
## [1] "Malay   10"
## [1] "Georgian   12"
## [1] "Albanian   5"
## [1] "Latvian   1"
## [1] "Afrikaans   13"
## [1] "Vietnamese   9"
## [1] "Basque   2"
## [1] "Norwegian Bokmål   1"
## [1] "Azerbaijani   1"
## [1] "Kannada   1"
## [1] "Zulu   1"
## [1] "Lithuanian   1"
## [1] "Javanese   1"
```

```
#RELATIVE FREQUENCY TABLE
relative_freq_ol <- prop.table(freq_table_ol) * 100

for (val in names(relative_freq_ol)) {
  print(paste(language_map[[val]], " ", round(relative_freq_ol[val], 2), "%"))
}
```

```
## [1] "English   38.12 %"
## [1] "Spanish   8.44 %"
## [1] "Japanese   7.8 %"
## [1] "French   6.63 %"
## [1] "Korean   5.68 %"
## [1] "Portuguese   4.06 %"
## [1] "German   3.34 %"
## [1] "Chinese   3.26 %"
## [1] "Hindi   2.07 %"
## [1] "Russian   1.96 %"
## [1] "Turkish   1.79 %"
## [1] "Italian   1.65 %"
## [1] "Dutch   1.6 %"
## [1] "Swedish   1.52 %"
## [1] "Chinese   1.33 %"
## [1] "Danish   1.27 %"
## [1] "Norwegian   1.15 %"
## [1] "Greek   1.08 %"
## [1] "Hebrew   0.81 %"
## [1] "Polish   0.74 %"
## [1] "Thai   0.63 %"
## [1] "Arabic   0.55 %"
## [1] "Urdu   0.53 %"
## [1] "Finnish   0.48 %"
## [1] "Czech   0.47 %"
## [1] "Tagalog   0.3 %"
## [1] "Serbian   0.21 %"
## [1] "Tamil   0.2 %"
## [1] "Hungarian   0.18 %"
## [1] "Persian   0.17 %"
## [1] "Afrikaans   0.15 %"
## [1] "Romanian   0.15 %"
## [1] "Slovak   0.15 %"
```

```
## [1] "Bulgarian   0.14 %"
## [1] "Indonesian   0.14 %"
## [1] "Georgian   0.14 %"
## [1] "Catalan   0.13 %"
## [1] "Telugu   0.13 %"
## [1] "Malay   0.12 %"
## [1] "Icelandic   0.1 %"
## [1] "Vietnamese   0.1 %"
## [1] "Bengali   0.07 %"
## [1] "Croatian   0.07 %"
## [1] "Ukrainian   0.07 %"
## [1] "Albanian   0.06 %"
## [1] "Galician   0.05 %"
## [1] "Welsh   0.03 %"
## [1] "Serbo-Croatian   0.03 %"
## [1] "Basque   0.02 %"
## [1] "Azerbaijani   0.01 %"
## [1] "Estonian   0.01 %"
## [1] "Javanese   0.01 %"
## [1] "Kannada   0.01 %"
## [1] "Latin   0.01 %"
## [1] "Luxembourgish   0.01 %"
## [1] "Lithuanian   0.01 %"
## [1] "Latvian   0.01 %"
## [1] "Norwegian Bokmål   0.01 %"
## [1] "Sinhalese   0.01 %"
## [1] "Zulu   0.01 %"
```

4. Exploring "networks" variable for TV shows

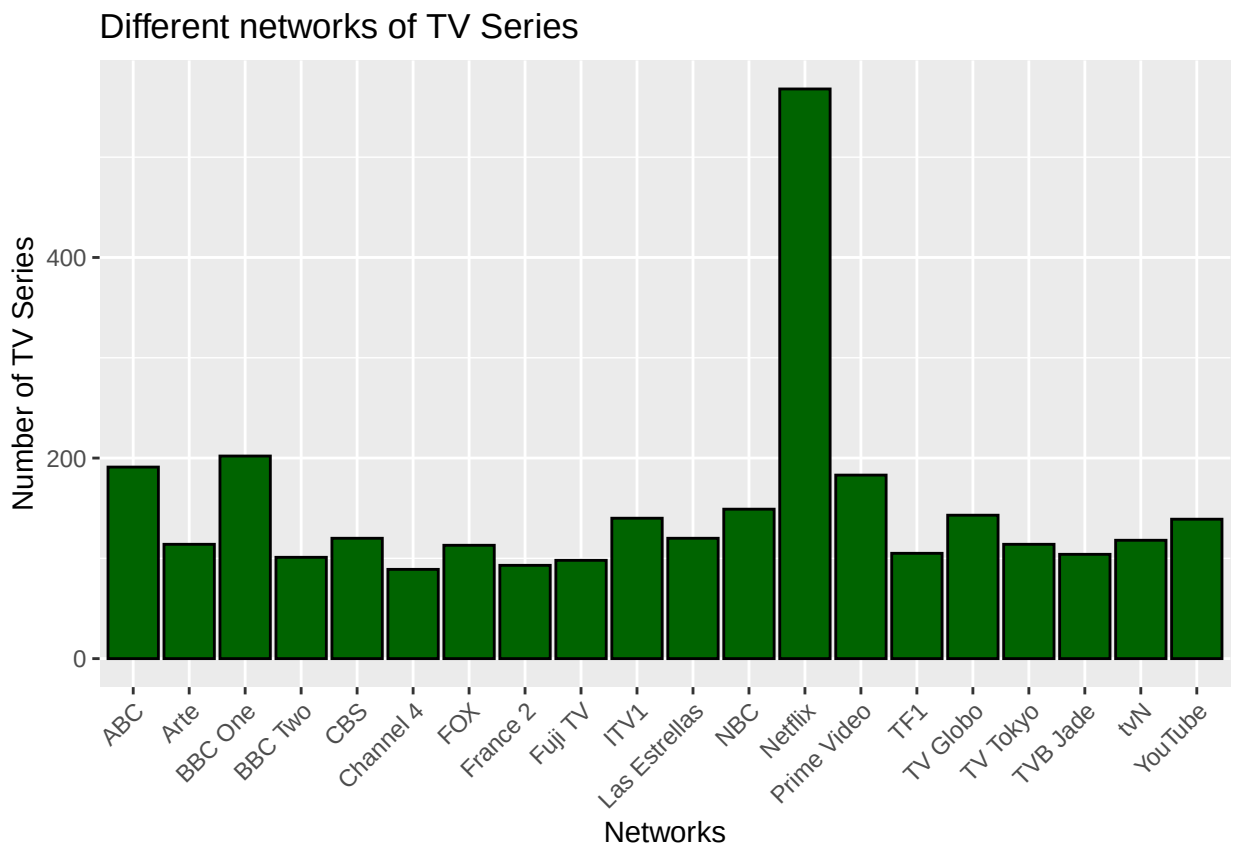
```
print(unique(sample_data$networks)[1:20])
```

```
##  [1] "Disney+"                  "Hulu"
##  [3] "USA Network"              "Channel 4"
##  [5] "Netflix"                  "FOX EspaÃ±a"
##  [7] "Pantaya"                  "ABC"
##  [9] "FOX"                      "KBS2, VIU"
## [11] "Syfy"                     "Hulu, Disney+"
## [13] "Blim"                     "Starz"
## [15] "The CW"                   "Showtime"
## [17] "FOX, National Geographic" "NBC"
## [19] "Atresplayer Premium"      "Apple TV+"
```

Since some of the unique values in "networks" column contains multiple networks separated by "," we can expand in the same way as done in genres. This is done internally in the function call for plotting by calling the expand_nested_data function (defined in the utils.R file)

```
#NETWORKS BAR PLOT
plot_histogram(
  data = sample_data,
  column_name = "networks",
```

```
  x_label = "Networks",
  y_label = "Number of TV Series",
  title = "Different networks of TV Series",
  type = 1,
  fillColor = "darkgreen")
```



From the barplot of networks, we can observe that most common network in the TV shows dataset is "Netflix", followed by "Youtube","TV Globo" and "Prime Video"

```
#ABSOLUTE FREQUENCY TABLE
df_expanded_networks <- expand_nested_data(sample_data,"networks")

colnames(df_expanded_networks)[colnames(df_expanded_networks) == "cat_col"] <- "network"

for (i in 1:nrow(df_expanded_networks)) {
  print(paste(df_expanded_networks[i, ], sep = " "))
}
```

```
## [1] "Netflix" "568"
## [1] "BBC One" "202"
## [1] "ABC" "191"
## [1] "Prime Video" "183"
## [1] "NBC" "149"
## [1] "TV Globo" "143"
## [1] "ITV1" "140"
```

```
## [1] "YouTube" "139"
## [1] "CBS" "120"
## [1] "Las Estrellas" "120"
## [1] "tvN" "118"
## [1] "Arte" "114"
## [1] "TV Tokyo" "114"
## [1] "FOX" "113"
## [1] "TF1" "105"
## [1] "TVB Jade" "104"
## [1] "BBC Two" "101"
## [1] "Fuji TV" "98"
## [1] "France 2" "93"
## [1] "Channel 4" "89"
```

```
#RELATIVE FREQUENCY TABLE
total_frequency_networks <- sum(df_expanded_networks$frequency)

for (i in 1:nrow(df_expanded_networks)) {
  network <- df_expanded_networks[i, "network"]
  freq <- df_expanded_networks[i, "frequency"]
  percentage <- (freq / total_frequency_networks) * 100

  # Print genre and relative frequency
  print(paste("Network:", network, "- RF :", round(percentage, 3), "%"))
}
```

```
## [1] "Network: Netflix - RF : 18.908 %"
## [1] "Network: BBC One - RF : 6.724 %"
## [1] "Network: ABC - RF : 6.358 %"
## [1] "Network: Prime Video - RF : 6.092 %"
## [1] "Network: NBC - RF : 4.96 %"
## [1] "Network: TV Globo - RF : 4.76 %"
## [1] "Network: ITV1 - RF : 4.66 %"
## [1] "Network: YouTube - RF : 4.627 %"
## [1] "Network: CBS - RF : 3.995 %"
## [1] "Network: Las Estrellas - RF : 3.995 %"
## [1] "Network: tvN - RF : 3.928 %"
## [1] "Network: Arte - RF : 3.795 %"
## [1] "Network: TV Tokyo - RF : 3.795 %"
## [1] "Network: FOX - RF : 3.762 %"
## [1] "Network: TF1 - RF : 3.495 %"
## [1] "Network: TVB Jade - RF : 3.462 %"
## [1] "Network: BBC Two - RF : 3.362 %"
## [1] "Network: Fuji TV - RF : 3.262 %"
## [1] "Network: France 2 - RF : 3.096 %"
## [1] "Network: Channel 4 - RF : 2.963 %"
```

5. Exploring “status” variable

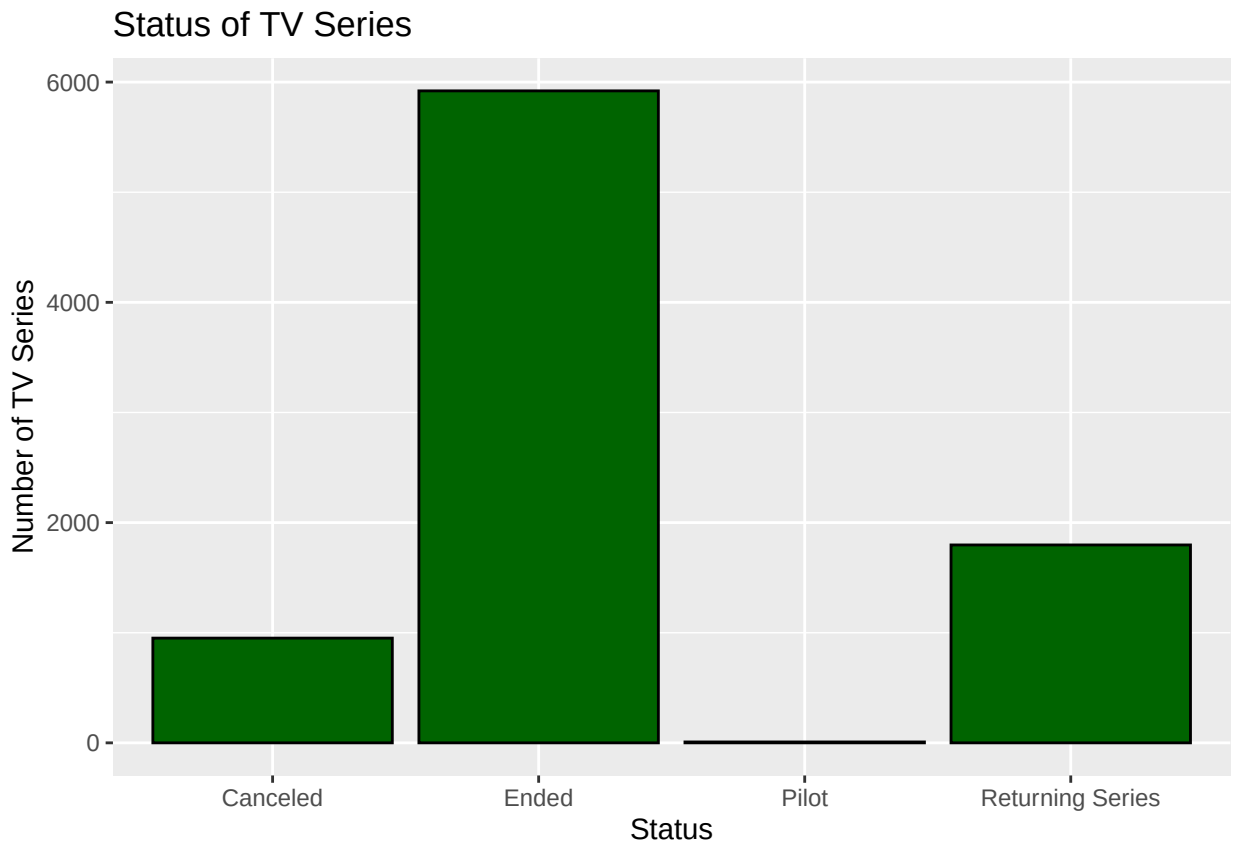
```
print(table(sample_data$status))
```

```
##
```

```
##          Canceled             Ended             Pilot Returning Series
##               951              5920                 8              1797
```

Since the categories are not in comma separated, repeating format, there is no requirement of expansion, we can therefore get the frequency tables and bar plot directly.

```r
#STATUS BAR PLOT
plot_histogram(
  data = sample_data,
  column_name = "status",
  x_label = "Status",
  y_label = "Number of TV Series",
  title = "Status of TV Series",
  type = 2,
  fillColor = "darkgreen")
```



Frequency tables for categorical variable 'status':

```r
#ABSOLUTE FREQUENCY TABLE
freq_table_status <- sort(table(sample_data$status), decreasing = TRUE)
for (val in unique(sample_data$status)) {
  print(paste(val, " ", freq_table_status[val]))
}
```

```
## [1] "Ended   5920"
```

```
## [1] "Canceled   951"
## [1] "Returning Series   1797"
## [1] "Pilot   8"
```

```
#RELATIVE FREQUENCY TABLE
relative_freq_status <- prop.table(freq_table_status) * 100

print("Status  Relative-Frequency")
```

```
## [1] "Status  Relative-Frequency"
```

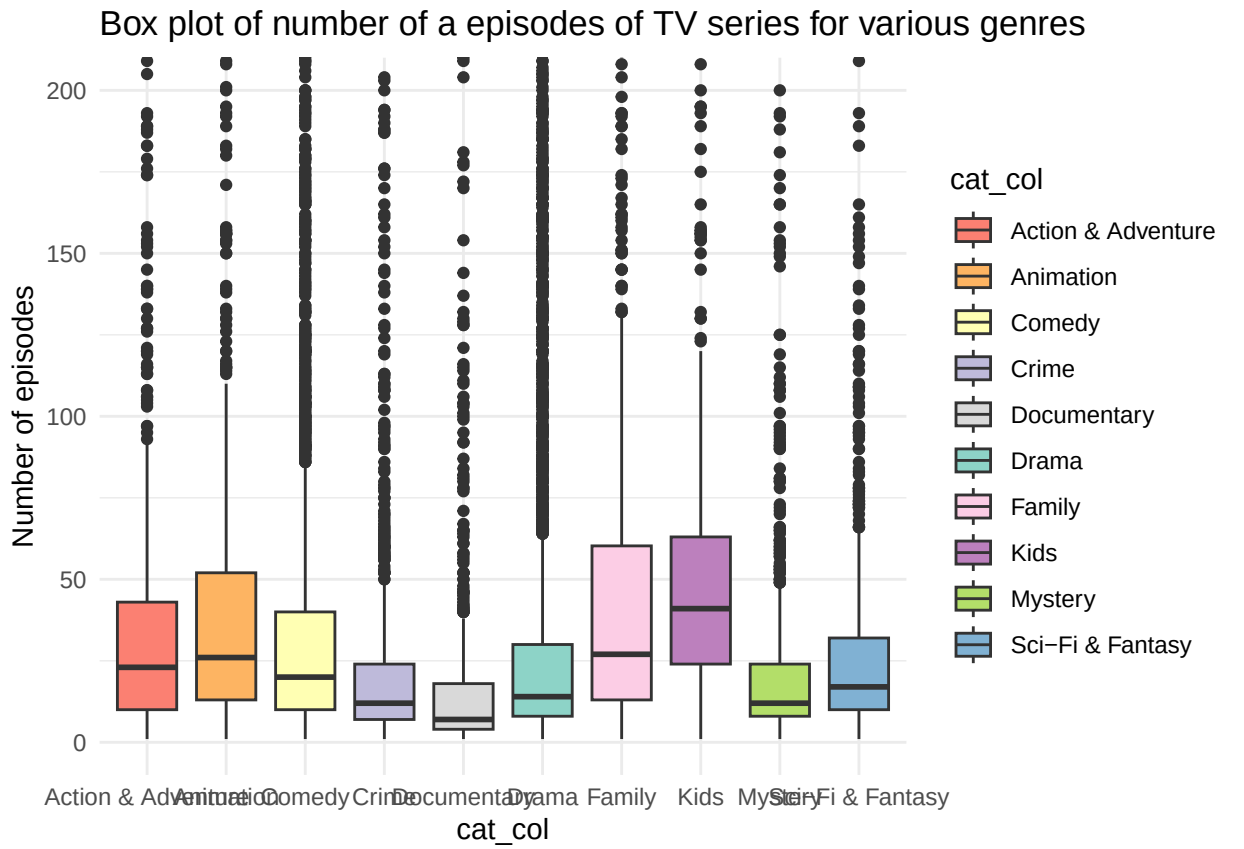
```
for (val in names(relative_freq_status)) {
  print(paste(val, " ", round(relative_freq_status[val], 2), "%"))
}
```

```
## [1] "Ended   68.23 %"
## [1] "Returning Series   20.71 %"
## [1] "Canceled   10.96 %"
## [1] "Pilot   0.09 %"
```

Relationships between variables

1. Number of episodes by Genre

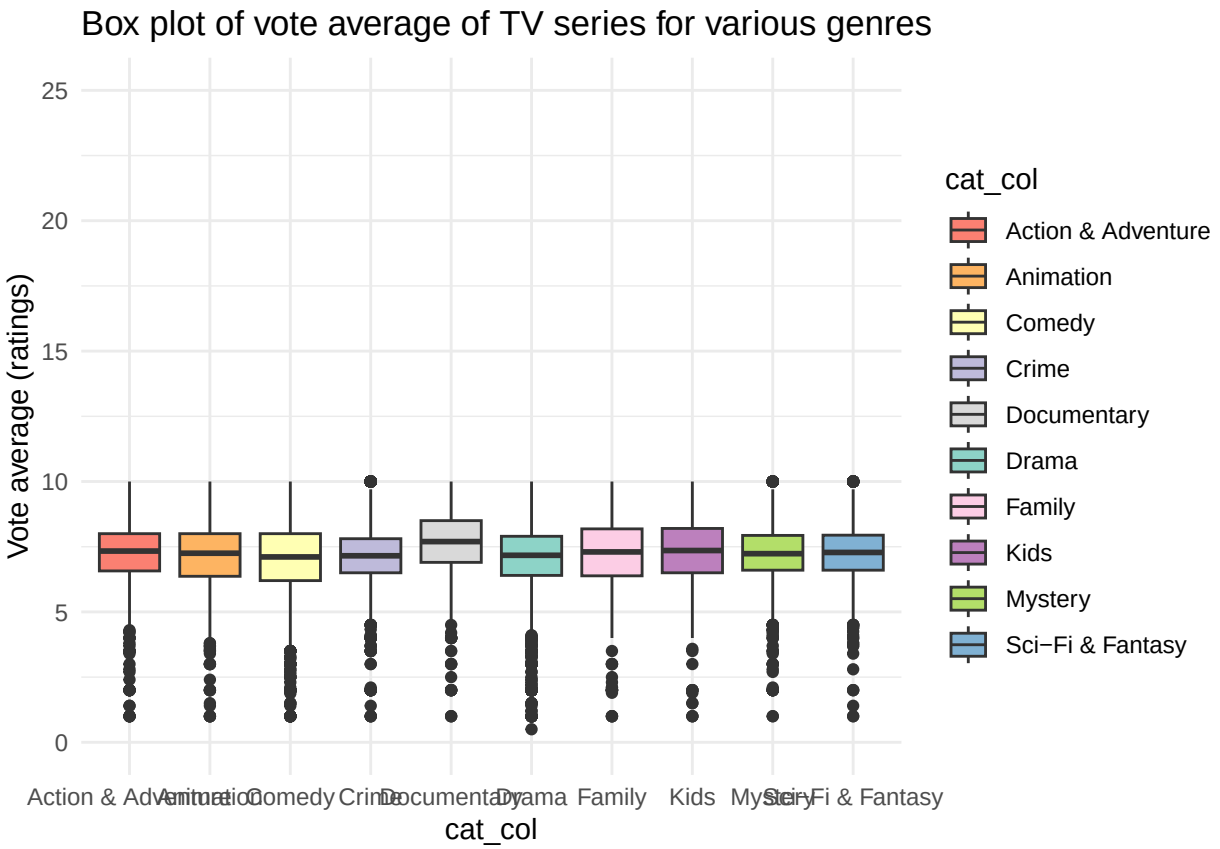
```
# Side-by-side boxplots for number of episodes by genre for TV series

expanded_df <- expand_categorical_cols(sample_data,"genres","number_of_episodes")

generate_box_plot(
  data = expanded_df,
  fill_column = "cat_col",
  y_col = "num_col",
  y_label = "Number of episodes",
  title = "Box plot of number of a episodes of TV series for various genres",
  ylim = c(0, 200)
)
```



Inference: From the boxplots above, it can be said that the variability in the number of episodes is comparable between the different genres. The "Family" and "Kids" genres tend to have higher medians for number of episodes whereas "Documentary" tends to have the least.

We can now check side-by-side box plots for vote_average (ratings) for different genres

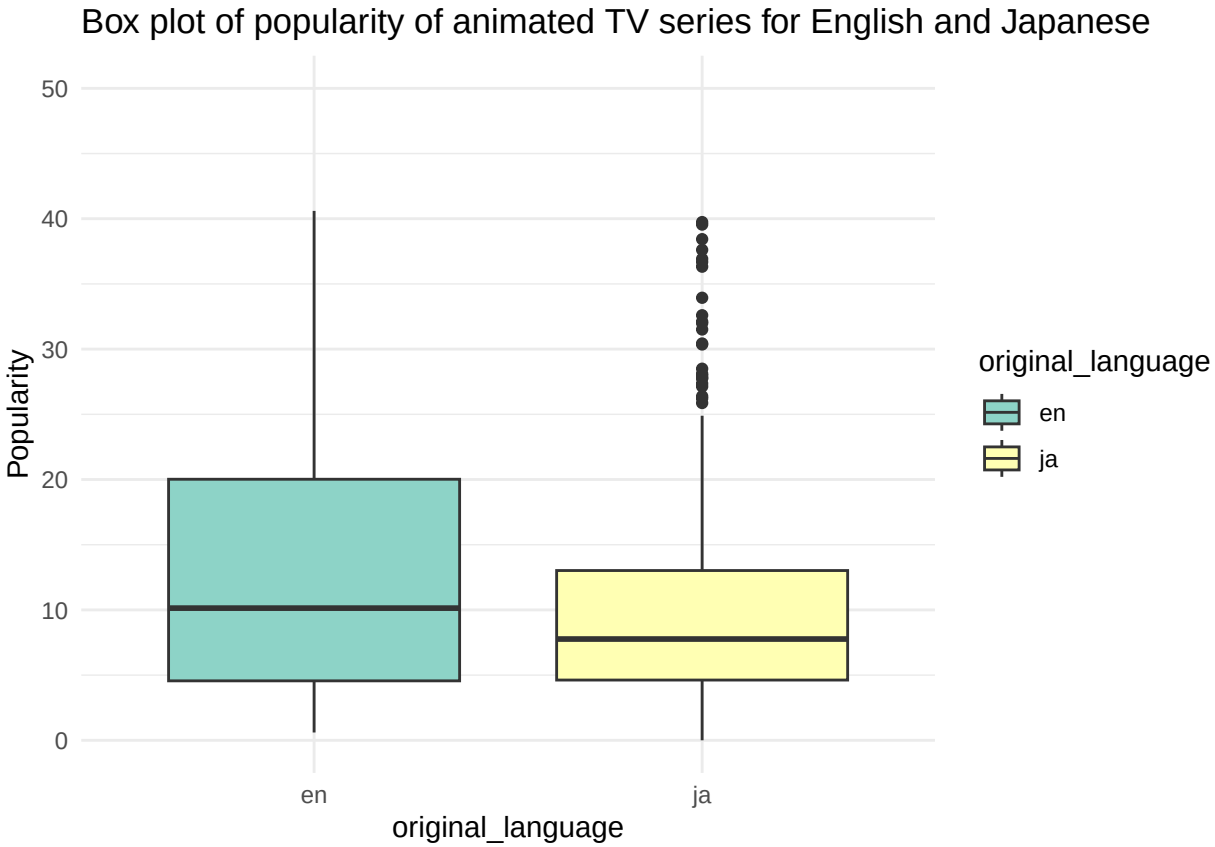
```
# Side-by-side boxplots for vote_average by genre for TV series

expanded_df <- expand_categorical_cols(sample_data,"genres","vote_average")

generate_box_plot(
  data = expanded_df,
  fill_column = "cat_col",
  y_col = "num_col",
  y_label = "Vote average (ratings)",
  title = "Box plot of vote average of TV series for various genres",
  ylim = c(0, 25)
)
```



From the side-by-side boxplots it is evident that the variability of ratings (vote_average) in different genres is comparable. However there appears to be many outliers beyond the lower whisker. The plot is ideal for ANOVA test for comparison of means, which will be further explored in the inferential statistics section.

Example: Compare the variability in popularity of "Japanese" vs "English" animated tv series, we can use side-by-side box plots

```
# Side-by-side boxplots for popularity by original language for animated  TV series



generate_box_plot(
  data = subset(sample_data, grepl("Animation", genres)),
  fill_column = "original_language",
  categories = c("en", "ja"),
  y_col = "popularity",
  y_label = "Popularity",
  title = "Box plot of popularity of animated TV series for English and Japanese",
  ylim = c(0, 50)
)
```



From the box-plots above, we can infer that the variablity in the distribution between animated TV shows in English is more than the variability of animated TV shows in Japanese. The median of animated shows in English is also higher (around 10) than the median of animated shows in Japanese (around 7).

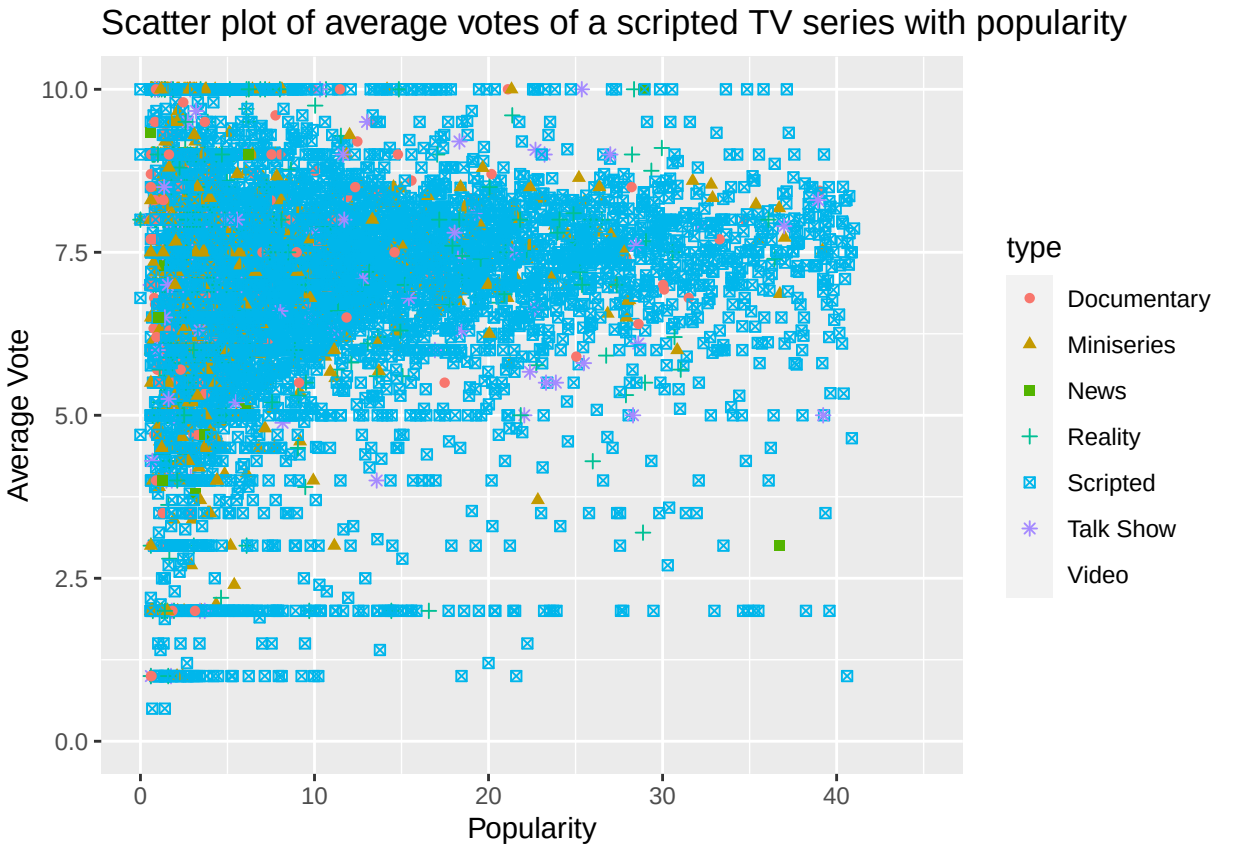
2. relationships between numerical variables.

```
# SCATTER PLOT BETWEEN POPULARITY AND VOTE_AVERAGE

# COVARIANCE
covariance <- cov(sample_data$popularity, sample_data$vote_average)
print(covariance)
```

```
## [1] 0.8482083
```

```
display_scatter_plot(
  data = sample_data,
  x_col = "popularity",
  y_col = "vote_average",
  z_col = "type",
  x_label = "Popularity",
  y_label = "Average Vote",
  title = "Scatter plot of average votes of a scripted TV series with popularity",
  xlim = c(0, 45),
```

```
  ylim = c(0, 10)
  )
```



The scatter plot of popularity vs vote_average depicts that for a wide range of "popularity" values the "vote_average" for many points remain the same, i.e., vote_average is constant for different values of Popularity in multiple cases. This indicates that the two variables are independent of each other. The covariance between the two variables is 0.848 (approximately). The color coding indicates that majority of the tv shows fall in the "scripted" type.

3. Number of seasons and Duration

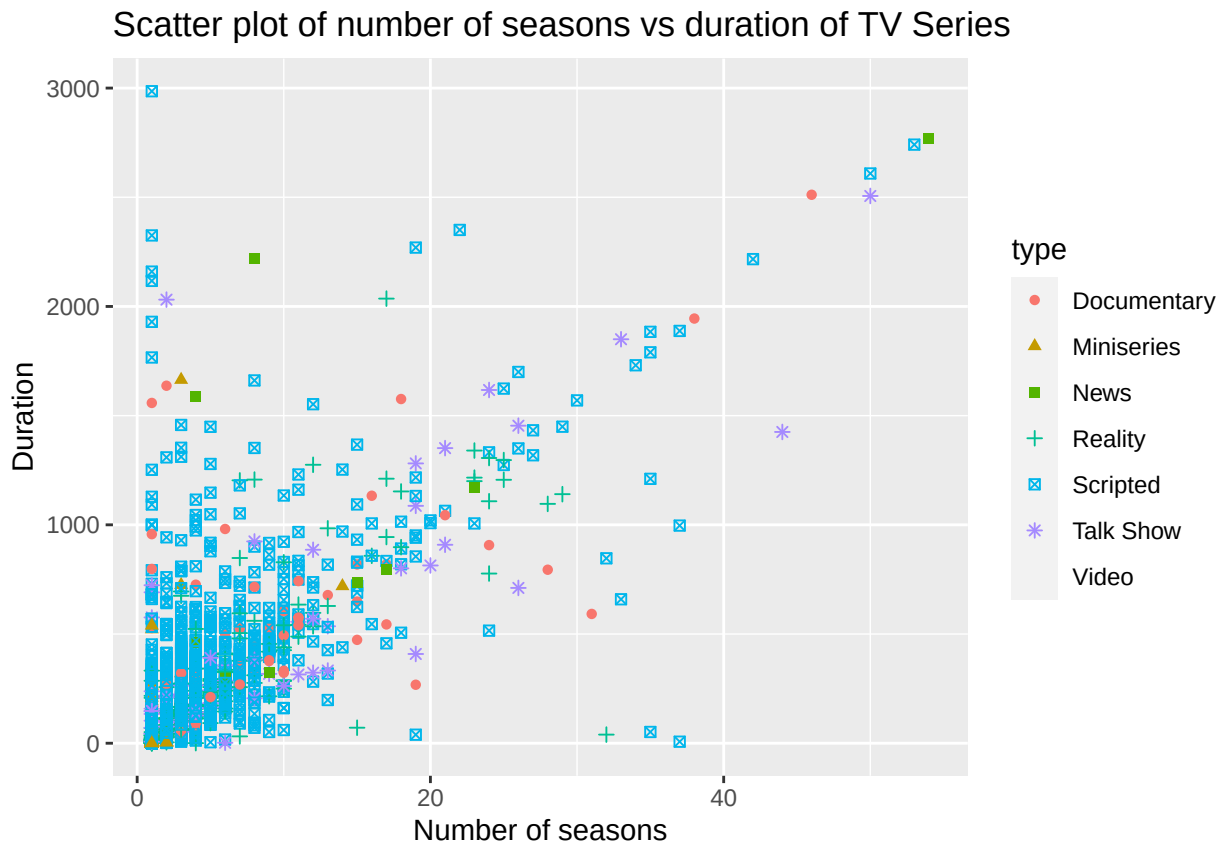
```
# SCATTER PLOT BETWEEN NUMBER OF SEASONS AND DURATION

# COVARIANCE
new_df <- get_duration(sample_data)
```

```
## Removing rows where first_air_date is greater than last_air_date...
```

```
covariance <- cov(new_df$number_of_seasons, new_df$duration)
print(covariance)
```

```
## [1] 532.1457
```

```
display_scatter_plot(
  data = new_df,
  x_col = "number_of_seasons",
  y_col = "duration",
  z_col = "type",
  x_label = "Number of seasons",
  y_label = "Duration",
  title = "Scatter plot of number of seasons vs duration of TV Series",
  #xlim = c(0, 45),
  #ylim = c(0, 10)
  )
```



The scatter plot above shows that the number of seasons is constant for a wide range of values of duration of TV Series implying that the two variables are independent of each other. A visible pattern of the outliers showing linear relationship cannot be accounted for as majority points do not follow the pattern.

Inferential Statistics

```
sample_data = read_excel("./TMDB_tv_dataset_v3.xlsx")
```

1. Inference about mean

1A. Verify the claim that the average runtime of a documentry show is greater than or equal to the average runtime of a reality show. //

Hypothesis Test

Consider the null hypothesis to be as follows H_0 : μ_1 - $\mu_2 \geq 0$. In the context of the question, null hypothesis states that the average runtime of a documentry show is greater than or equal to the average runtime of a reality show.

As such the alternate hypothesis would be H_1 : μ_1 - $\mu_2 < 0$. In the context of the question, alternate hypothesis states that the average runtime of a documentry show is lesser than the average runtime of a reality show.

Here

μ_1 is the population mean of runtime of documentry TV shows.

μ_2 is the population mean of runtime of reality TV shows.

Significance level of the test $\alpha = 5\%$

```
filter_sample_Data = sample_data[sample_data$episode_run_time > 0, ];
#print(head(filter_sample_Data, 50));
#cat(subset(filter_sample_Data, 1:50), file="", sep="\n")
#cat(capture.output(print(filter_sample_Data[1:50, ])), sep="\n")

documentry_filter_sample = subset(filter_sample_Data, type == "Documentary");
reality_filter_sample = subset(filter_sample_Data, type == "Reality");
```

```
result = t.test(documentry_filter_sample$episode_run_time,
       reality_filter_sample$episode_run_time,
       var.equal = FALSE,
       alternative = "less");
print(result$statistic);
```

```
##         t
## -3.213635
```

```
print(result$p.value);
```

```
## [1] 0.0006574883
```

Result:

t_statistic = -3.213635 and p_value = 0.0006574883

Since p_value is less than α (0.05) we reject the null hypothesis H_0.

We have enough evidence to claim that the average runtime of a documentry show is not greater than or equal to the average runtime of a reality show.

Confidence intevals:

```
mean_documentry_smaple = mean(documentry_filter_sample$episode_run_time);
sd_documentry_sample = sd(documentry_filter_sample$episode_run_time);
mean_reality_sample = mean(reality_filter_sample$episode_run_time);
sd_reality_sample = sd(reality_filter_sample$episode_run_time);

size_documentry_sample = nrow(documentry_filter_sample);
size_reality_sample = nrow(reality_filter_sample);
confidence_level <- 0.95;
z_score = qnorm((1 + confidence_level) / 2);

margin_of_error = z_score * sqrt((sd_documentry_sample^2 / size_documentry_sample) + (sd_reality_sample
confidence_interval = list(
  lower_bound = mean_documentry_smaple - mean_reality_sample - margin_of_error,
  upper_bound = mean_documentry_smaple - mean_reality_sample + margin_of_error
);

print (confidence_interval);
```

```
## $lower_bound
## [1] -6.062504
##
## $upper_bound
## [1] -1.469071
```

This variable confidence_interval contains the lower and upper bounds of the confidence interval for the difference in means between documentary and reality show's episode runtime. Our null hypothesis H_0 states that the value (of the differece in runtimes) should be at leaat 0.

Since 0 doesn't lie in the range of our confidence interval (-6.062504, -1.469071) we can say that the average runtime of a documentry show is not greater than or equal to the average runtime of a reality show.

1B. Average episodes per season

```
  scripted_sample_Data = subset(sample_data, type == "Scripted");

  filter_scripted_data = scripted_sample_Data[scripted_sample_Data$number_of_episodes > 0, ];
  filter_scripted_data = filter_scripted_data[filter_scripted_data$number_of_seasons > 1, ];
  # TODO: shouldn't scripted be enough as a filter?
  filter_scripted_data = filter_scripted_data %>% mutate(episodes_per_season = number_of_episodes/numbe
```

i. Verify the claim that the average episodes per season of comedy shows is greater than or equal to the average episodes per season of darma shows.

Hypothesis Test:

Consider the null hypothesis to be as follows $H_0 : \mu_1 - \mu_2 \geq 0$. In the context of the question, null hypothesis states that the average episodes per season of comedy shows is greater than or equal to the average episodes per season of darma shows.

As such the alternate hypothesis would be H_1 : μ_1 - μ_2 <0. In the context of the question, alternate hypothesis states that the average episodes per season of comedy shows is lesser than the average episodes per season of darma shows.

Here

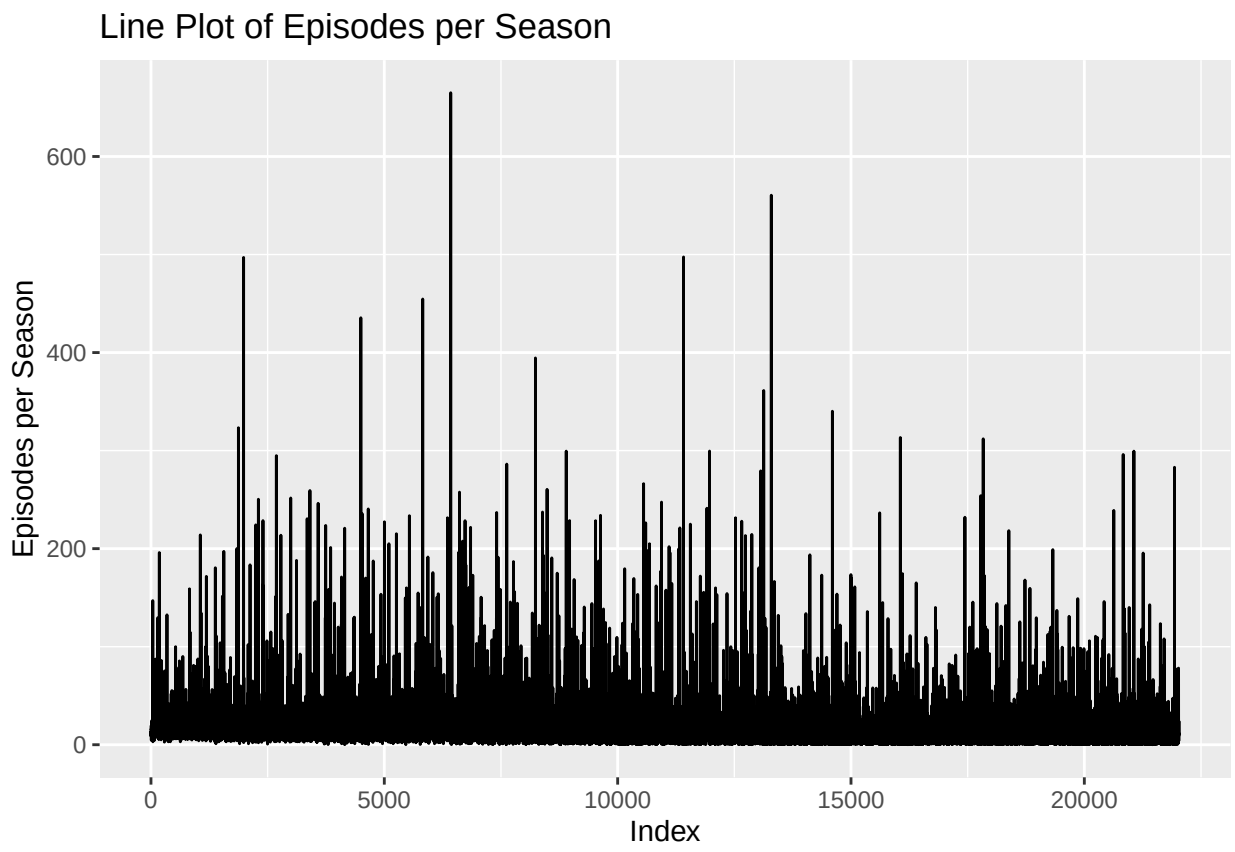
μ_1 is the population mean of episodes per season of comedy shows.

μ_2 is the population mean of episodes per season of drama shows.

Significance level of the test $\alpha = 5\%$

```
  # expanded_filter_data = expand_num_by_cat(filter_scripted_data, "genres", "episodes_per_season"); #
  # comedy_filter_data =  subset(expanded_filter_data, cat_col == "Comedy");
  # drama_filter_data =  subset(expanded_filter_data, cat_col == "Drama");
  # colnames(expanded_filter_data);

  comedy_filter_data =  subset(filter_scripted_data, grepl("Comedy", genres));
  drama_filter_data =  subset(filter_scripted_data, grepl("Drama", genres));
```

```
ggplot(filter_scripted_data, aes(x = 1:nrow(filter_scripted_data), y = episodes_per_season)) +
  geom_line() +
  labs(x = "Index", y = "Episodes per Season", title = "Line Plot of Episodes per Season");
```




```
  result = t.test(comedy_filter_data$episodes_per_season,
         drama_filter_data$episodes_per_season,
         var.equal = FALSE,
```

```
        alternative = "less");
  print(result$statistic);
```

```
##         t
## -3.598296
```

```
  print(result$p.value);
```

```
## [1] 0.000160871
```

Result:

t_statistic = -3.598296 and p_value = 0.000160871

Since p_value is less than α (0.05) we reject the null hypothesis H_0.

We have enough evidence to claim that the average episodes per season of comedy shows is not greater than or equal to the average episodes per season of darma shows.

Confidence Interval

```
mean_comedy_smaple = mean(comedy_filter_data$episodes_per_season);
sd_comedy_sample = sd(comedy_filter_data$episodes_per_season);
mean_drama_sample = mean(drama_filter_data$episodes_per_season);
sd_drama_sample = sd(drama_filter_data$episodes_per_season);

size_comedy_sample = nrow(comedy_filter_data);
size_drama_sample = nrow(drama_filter_data);
confidence_level = 0.95;
z_score = qnorm((1 + confidence_level) / 2);

margin_of_error = z_score * sqrt((sd_comedy_sample^2 / size_comedy_sample) + (sd_drama_sample^2 / size_
confidence_interval = list(
  lower_bound = mean_comedy_smaple - mean_drama_sample - margin_of_error,
  upper_bound = mean_comedy_smaple - mean_drama_sample + margin_of_error
);

print (confidence_interval);
```

```
## $lower_bound
## [1] -2.584698
##
## $upper_bound
## [1] -0.7618559
```

This variable confidence_interval contains the lower and upper bounds of the confidence interval for the difference in means between comedy and drama show's episodes per season. Our null hypothesis H_0 states that the value (of the differece in episodes per season) should be at least 0.

Since 0 doesn't lie in the range of our confidence interval (-2.584698, -0.7618559) we can say that the average episodes per season of comedy shows is not greater than or equal to the average episodes per season of darma shows.

ii. Comparr if average value of episodes_per_season of SciFi is greater than that of Action & Adventure

Consider the null hypothesis to be as follows H_0 : μ_1 - $\mu_2 \geq 0$. In the context of the question, null hypothesis states that the average episodes per season of SciFi shows is greater than or equal to the average episodes per season of Action & Adventure shows.

As such the alternate hypothesis would be H_1 : μ_1 - $\mu_2 < 0$. In the context of the question, alternate hypothesis states that the average episodes per season of SciFi shows is lesser than the average episodes per season of Action & Adventure shows.

Here

μ_1 is the population mean of episodes per season of SciFi shows.

μ_2 is the population mean of episodes per season of Action & Adventure shows.

Significance level of the test $\alpha = 5\%$

```
scifi_filter_data =  subset(filter_scripted_data, grepl("Sci-Fi & Fantasy", genres));
adv_filter_data =  subset(filter_scripted_data, grepl("Action & Adventure", genres));
```

```
result = t.test(scifi_filter_data$episodes_per_season,
       adv_filter_data$episodes_per_season,
       var.equal = FALSE,
       alternative = "less");
print(result$statistic);
```

```
##         t
## -1.460528
```

```
print(result$p.value);
```

```
## [1] 0.0721231
```

Result:

t_statistic = -1.460528 and p_value = 0.0721231

Since p_value is greater than α (0.05) we fail to reject the null hypothesis H_0.

We have enough evidence to claim that the average episodes per season of SciFi shows is greater than or equal to the average episodes per season of Action & Adventure shows.

iii. Comparing average values of episodes_per_season of Family against that of Crime

Consider the null hypothesis to be as follows H_0 : μ_1 - $\mu_2 \leq 0$. In the context of the question, null hypothesis states that the average episodes per season of Family shows is lesser than or equal to the average episodes per season of Crime shows.

As such the alternate hypothesis would be H_1 : μ_1 - $\mu_2 > 0$. In the context of the question, alternate hypothesis states that the average episodes per season of Family shows is greater than the average episodes per season of Crime shows.

Here

μ_1 is the population mean of episodes per season of Family shows.

μ_2 is the population mean of episodes per season of Crime shows.

Significance level of the test $\alpha = 5\%$

```
  family_filter_data =  subset(filter_scripted_data, grepl("Family", genres));
  crime_filter_data =  subset(filter_scripted_data, grepl("Crime", genres));
```

```
  result = t.test(family_filter_data$episodes_per_season,
         crime_filter_data$episodes_per_season,
         var.equal = FALSE,
         alternative = "greater");
  print(result$statistic);
```

```
##        t
## 11.55692
```

```
  print(result$p.value);
```

```
## [1] 1.619423e-30
```

Result:

t_statistic = 11.55692 and p_value = 1.619423e-30

Since p_value is lesser than α (0.05) we reject the null hypothesis H_0.

We have enough evidence to claim that the average episodes per season of Family shows is lesser than or equal to the average episodes per season of Family shows.

1C Independent : we can consider the popularity rating of genre in english (particular language) / country or origin, eg weather the popularity of animation is same in english to that in japanese.

Consider the null hypothesis to be as follows H_0 : μ_1 - $\mu_2 \geq 0$. In the context of the question, null hypothesis states that the average popularity of English Animated shows is greater than or equal to the average popularity of Japanese Animated shows.

As such the alternate hypothesis would be H_1 : μ_1 - $\mu_2 < 0$. In the context of the question, alternate hypothesis states that the average popularity of English Animated shows is lesser than the average popularity of Japanese Animated shows.

Here

μ_1 is the population mean of popularity of English Animated shows.

μ_2 is the population mean of popularity of Japanese Animated shows.

Significance level of the test $\alpha = 5\%$

```
#anime_sample_Data = subset(sample_data, genres == "Animation"); #FALSE, this is strict matching
anime_sample_Data = subset(sample_data, grepl("Animation", genres)); # This is 'contains' matching

q1 = quantile(anime_sample_Data$popularity, probs = 0.25);
q3 = quantile(anime_sample_Data$popularity, probs = 0.75);
iqr = q3 - q1;
```

```
upper_whisker_limit = q3 + (1.5 * iqr);
lower_whisker_limit = q1 - (1.5 * iqr);

anime_sample_Data = subset(anime_sample_Data, popularity > lower_whisker_limit); # This should not be 0
anime_sample_Data = subset(anime_sample_Data, popularity < upper_whisker_limit);

english_anime_data = subset(anime_sample_Data, original_language == "en"); # strict matchin is correct
japanese_samples = subset(anime_sample_Data, original_language == "ja");
```

```
  result = t.test(english_anime_data$popularity,
         japanese_samples$popularity,
         var.equal = FALSE,
         alternative = "less");
  print(result$statistic);
```

```
##         t
## -2.510185
```

```
  print(result$p.value);
```

```
## [1] 0.006047949
```

Result:

t_statistic = 9.019615 and p_value = 1 (In case of strict matching lol)

t_statistic = -2.510185 and p_value = 0.006047949 (In case of contains matching. We are using this.)

Since p_value is less than α (0.05) we reject the null hypothesis H_0.

We have enough evidence to claim that the average popularity of English Animated shows is not greater than or equal to the average popularity of Japanese Animated shows.

1D Comparing Rating (Vote Average) of Animated English and Animated Japanese Shows

Consider the null hypothesis to be as follows $H_0 : \mu_1 - \mu_2 \geq 0$. In the context of the question, null hypothesis states that the average Rating of English Animated shows is greater than or equal to the average Rating of Japanese Animated shows.

As such the alternate hypothesis would be $H_1 : \mu_1 - \mu_2 < 0$. In the context of the question, alternate hypothesis states that the average Rating of English Animated shows is lesser than the average Rating of Japanese Animated shows.

Here

μ_1 is the population mean of rating of English Animated shows.

μ_2 is the population mean of rating of Japanese Animated shows.

Significance level of the test $\alpha = 5\%$

```r
anime_sample_Data = subset(sample_data, grepl("Animation", genres)); # This is 'contains' matching

q1 = quantile(anime_sample_Data$popularity, probs = 0.25);
q3 = quantile(anime_sample_Data$popularity, probs = 0.75);
iqr = q3 - q1;
upper_whisker_limit = q3 + (1.5 * iqr);
lower_whisker_limit = q1 - (1.5 * iqr);

anime_sample_Data = subset(anime_sample_Data, popularity > lower_whisker_limit); # This should not be 0
# anime_sample_Data = subset(anime_sample_Data, popularity < upper_whisker_limit);

english_anime_data = subset(anime_sample_Data, original_language == "en"); # strict matchin is correct
japanese_samples = subset(anime_sample_Data, original_language == "ja");
```

```r
  result = t.test(english_anime_data$vote_average,
         japanese_samples$vote_average,
         var.equal = FALSE,
         alternative = "less");
  print(result$statistic);
```

```
##         t
## -11.27042
```

```r
  print(result$p.value);
```

```
## [1] 1.640778e-29
```

Result:

t_statistic = -7.357608 and p_value = 1.533366e-13 (In case of strict matching)

t_statistic = -14.29599 and p_value = 8.377412e-46 (In case of contains matching. We are using this.)

Since p_value is less than α (0.05) we reject the null hypothesis H_0.

We have enough evidence to claim that the average Rating of English Animated shows is not greater than or equal to the average rating of Japanese Animated shows.

2. Inference about variance.

2A Verify the claim that the variance in rating of shows on Netflix is the same as variance in rating of shows on Amazon Prime.

Consider the null hypothesis to be as follows $H_0 : \sigma_1^2 - \sigma_2^2 = 0$. In the context of the question, null hypothesis states that the variance in rating of shows on Netflix is the same as variance in rating of shows on Amazon Prime

As such the alternate hypothesis would be $H_1 : \sigma_1^2 - \sigma_2^2 \neq 0$. In the context of the question, alternate hypothesis states that the variance in rating of shows on Netflix is significantly different from variance in rating of shows on Amazon Prime

Here

σ_1 is the standard deviation in rating of shows on Netflix.

σ_2 is the standard deviation in rating of shows on Amazon Prime.

Significance level of the test $\alpha = 5\%$

```
filter_sample_Data = sample_data[sample_data$vote_average >0, ];
netflix_sample_Data = subset(sample_data, grepl("Netflix", networks));
amazon_sample_Data = subset(sample_data, grepl("Prime Video", networks));
```

```
f_statistic = var(netflix_sample_Data$vote_average) / var(amazon_sample_Data$vote_average);
df1 = nrow(netflix_sample_Data) - 1;
df2 = nrow(amazon_sample_Data) - 1;
p_value = 2 * pf(f_statistic, df1, df2);

print(f_statistic);
```

```
## [1] 0.5180385
```

```
print(p_value);
```

```
## [1] 6.529343e-32
```

Result:

f_statistic = 0.5180385 and p_value = 6.529343e-32

Since p_value is less than α (0.05) we reject the null hypothesis H_0.

We have enough evidence to claim that the variance in rating of shows on Netflix is not the same as variance in rating of shows on Amazon Prime

Confidence Interval

```
var_netflix_smaple = var(netflix_sample_Data$vote_average);
var_amazon_sample = var(amazon_sample_Data$vote_average);

size_netflix_sample = nrow(netflix_sample_Data);
size_amazon_sample = nrow(amazon_sample_Data);

confidence_level = 0.95;
# df1 = nrow(netflix_sample_Data) - 1; Calculated in earlier block
# df2 = nrow(amazon_sample_Data) - 1;  Calculated in earlier block

f_lower = qf((1 - confidence_level) / 2, df1, df2);
f_upper = qf(1 - (1 - confidence_level) / 2, df1, df2);

confidence_interval <- c((var_netflix_smaple / var_amazon_sample) * (1 / f_upper), (var_netflix_smaple /

print (confidence_interval);
```

```
## [1] 0.4619673 0.5809154
```

This variable confidence_interval contains the lower and upper bounds of the confidence interval for the difference variance in rating of shows on Netflix and Amazpn. Our null hypothesis H_0 states that the value (of the difference in episodes per season) should be at least 0.

Since 0 doesn't lie in the range of our confidence interval (0.4619673, 0.5809154) we can say that the variance in rating of shows on Netflix is not the same as variance in rating of shows on Amazon Prime.

NOTE: The fact that the confidence interval is relatively narrow (0.46 - 0.58) indicates that we can be fairly confident about the range of possible values for the ratio of variances. This provides valuable information about the relative variability of the two groups. In this case we are fairly certain that the difference in variance is not 0 (rather the ratio of the vairances is not equal to 1 but lies somewhere in the interval 0.46 - 0.58)

2C Test the claim that The variance of the number of episodes for Music genere is equal to or greater than the variance of the number of episodes for History genere.

Consider the null hypothesis to be as follows $H_0 : \sigma_1^2 - \sigma_2^2 \geq 0$. In the context of the question, null hypothesis states that the variance of the number of episodes for Music is the greater than or equal to variance of the number of episodes for History genere.

As such the alternate hypothesis would be $H_1 : \sigma_1^2 - \sigma_2^2 < 0$. In the context of the question, alternate hypothesis states that the variance of the number of episodes for Music is the lesser than the variance of the number of episodes for History genere.

Here

σ_1 is the standard deviation of the number of episodes for Music genere.

σ_2 is the standard deviation of the number of episodes for History genere

Significance level of the test $\alpha = 5\%$

```r
data_2c <- subset(sample_data, select = c("genres", "number_of_episodes"))

# Dropping NA values
data_2c <- na.omit(data_2c)

# Filtering rows where "number_of_episodes" is greater than 0
data_2c <- data_2c[data_2c$number_of_episodes > 0, ]

# Resetting the row indices
data_2c <- data.frame(data_2c)  # Ensure data_2c is a dataframe
rownames(data_2c) <- NULL  # Reset row names

expanded_data_2c <- expand_col1_by_col2(data_2c, "number_of_episodes", "genres")
# dim(expanded_data_2c)

unique_value_counts <- table(expanded_data_2c$genres)
# print(sort(unique_value_counts, decreasing = TRUE))

# Subset 'number_of_episodes' for 'genres' equal to "Music"
music_num_epi_samples <- expanded_data_2c[expanded_data_2c$genres == "Music", "number_of_episodes"]

# Subset 'number_of_episodes' for 'genres' equal to "History"
history_num_epi_samples <- expanded_data_2c[expanded_data_2c$genres == "History", "number_of_episodes"]
```

```r
test_results <- var.test(music_num_epi_samples, history_num_epi_samples, alt="less", conf.level = 0.95)
print(test_results$statistic);
```

```
##         F
## 0.3991559
```

```r
print(test_results$p.value);
```

```
## [1] 0.001351561
```

Result:

f_statistic = 0.3991559 and p_value = 0.001351561

Since p_value is less than α (0.05) we reject the the null hypothesis H_0.

We have enough evidence to claim that the that the variance of the number of episodes for Music is not greater than or equal to variance of the number of episodes for History genere.

2D Find out if the claim that the variance of episode runtime for scripted shows is equal to or less than the variance of episode runtime for miniseries is true.

Consider the null hypothesis to be as follows $H_0 : \sigma_1^2 - \sigma_2^2 \leq 0$. In the context of the question, null hypothesis states that the variance of episode runtime for scripted shows is equal to or less than the variance of episode runtime for miniseries.

As such the alternate hypothesis would be $H_1 : \sigma_1^2 - \sigma_2^2 > 0$. In the context of the question, alternate hypothesis states that the variance of episode runtime for scripted shows is greater the variance of episode runtime for miniseries.

Here

σ_1 is the standard deviation of episode runtime for scripted shows

σ_2 is the standard deviation of episode runtime for miniseries

Significance level of the test $\alpha = 5\%$

```r
data_2d <- sample_data[, c("type", "episode_run_time")]

# Drop rows with NA values
data_2d <- na.omit(data_2d)

# Filter rows where 'episode_run_time' is greater than 0
data_2d <- data_2d[data_2d$episode_run_time > 0, ]

# Filter rows where 'episode_run_time' is less than 90
data_2d <- data_2d[data_2d$episode_run_time < 70, ]

# Reset row indices
data_2d <- data.frame(data_2d, row.names = NULL)
```

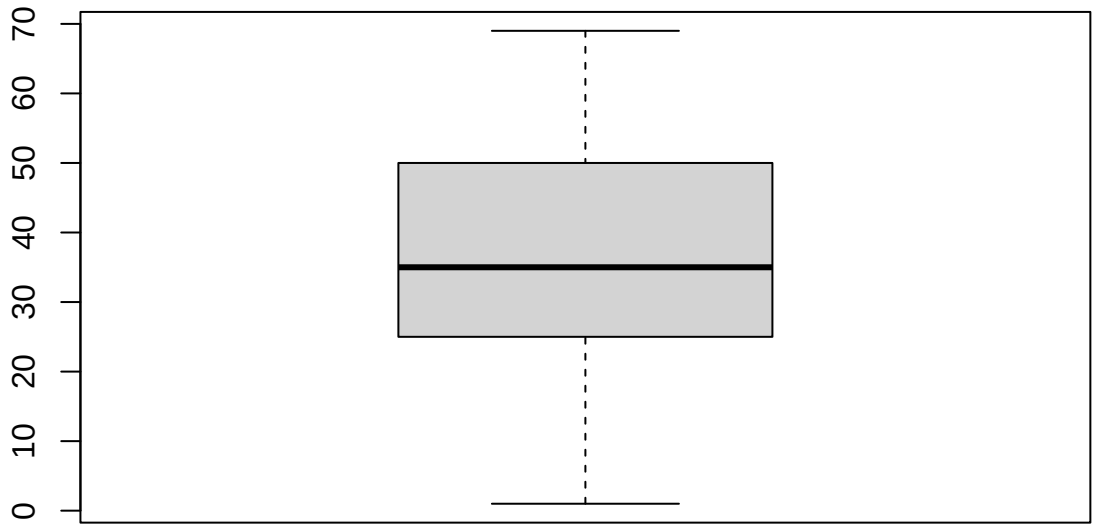
```r
# dim(data_2d)

# Filter 'episode_run_time' for 'type' equal to "Scripted"
scripted_ert_samples <- data_2d[data_2d$type == "Scripted", "episode_run_time"]

# Filter 'episode_run_time' for 'type' equal to "Miniseries"
miniseries_ert_samples <- data_2d[data_2d$type == "Miniseries", "episode_run_time"]

test_results <-var.test(scripted_ert_samples, miniseries_ert_samples, ratio = 1, alt="greater", conf.lev
print(test_results$statistic);
```

```
##        F
## 1.015139
```

```r
print(test_results$p.value);
```

```
## [1] 0.2245131
```

```r
boxplot(data_2d$episode_run_time);
```



Result:

f_statistic = 1.015139 and p_value = 0.2245131

Since p_value is greater than α (0.05) we fail to reject the the null hypothesis H_0.

We have enough evidence to conclude the claim to be true that the variance of episode runtime for scripted shows is equal to or less than the variance of episode runtime for miniseries is true.

3. Inference on Proportions

3C To what extent can we generalize the claim that one-third of Netflix shows are returning series based on our sample data?

Consider the null hypothesis to be as follows $H_0 : p = p_0$. In the context of the question, null hypothesis states that one-third of Netflix shows are returning series

As such the alternate hypothesis would be $H_1 : p \neq p_0$. In the context of the question, alternate hypothesis states that the ratio of Netflix shows that are returning is not one third.

Here

p_o is 1/3

Significance level of the test $\alpha = 5\%$

```
# Subset 'networks' and 'status' columns
data_3c <- sample_data[, c("networks", "status")]

# Drop rows with NA values
data_3c <- na.omit(data_3c)

# Reset row indices
data_3c <- data.frame(data_3c, row.names = NULL)
# head(data_3c)


# dim(data_3c)

expanded_data_3c <- expand_col1_by_col2(data_3c, "status", "networks")
# dim(expanded_data_3c)


# Subset 'status' for 'networks' equal to "Netflix"
netflix_samples <- expanded_data_3c[expanded_data_3c$networks == "Netflix", "status"]

# Count the occurrences of each unique value in 'netflix_samples'
netflix_samples_counts <- table(netflix_samples)

# Display the counts of unique values in 'netflix_samples'
print(netflix_samples_counts)
```

```
## netflix_samples
##         Canceled            Ended    In Production          Planned
##              207              941              133               35
## Returning Series
##              574
```

```
successes <- table(netflix_samples)["Returning Series"]
sample_size <- length(netflix_samples)
#print(successes)
#print(sample_size)
```

```r
pknot <- 1/3

test_results <- prop.test(successes, sample_size, p = pknot, alternative = "two.sided", conf.level = 0.9
print(test_results$statistic)
```

```
## X-squared
##  7.466667
```

```r
print(test_results$p.value)
```

```
## [1] 0.006285182
```

Result:

X-squared = 7.466667 and p_value = 0.006285182

Since p_value is less than α (0.05) we reject the the null hypothesis H_0.

We have enough evidence to conclude that the ratio of Netflix shows that are returning is not one third.

3D Proportion of TV shows that are streamed in English language

```r
# Subset 'type', 'original_language', and 'genres' columns
data_3d <- sample_data[, c("type", "original_language", "genres")]

# Drop rows with NA values
data_3d <- na.omit(data_3d)

# Reset row indices
data_3d <- data.frame(data_3d, row.names = NULL)

#head(data_3d)

type_counts <- table(data_3d$type)
#print(type_counts)

scripted_samples <- data_3d[data_3d$type == "Scripted",]
scripted_samples <- data.frame(scripted_samples, row.names = NULL)

expanded_scripted_samples <- expand_col1_by_col2(scripted_samples, "original_language", "genres")
#dim(expanded_scripted_samples)
```

i. Test the claim that approximately 40% or greater of the total comedy shows are streamed in english

Consider the null hypothesis to be as follows $H_0 : p \geq p_0$. In the context of the question, null hypothesis states that proportion of 40% (or greater) comedy shows are streamed in English language

As such the alternate hypothesis would be $H_1 : p < p_0$. In the context of the question, alternate hypothesis states hat proportion of lesser than 40% of comedy shows are streamed in English language

Here

p_o is 2/5 (or 0.4)

Significance level of the test $\alpha = 5\%$

```
# For comedy

# Filter 'expanded_scripted_samples' for 'genres' equal to "Comedy"
scripted_comedy_samples <- expanded_scripted_samples[expanded_scripted_samples$genres == "Comedy", ]

# Count occurrences of 'en' in 'original_language' for 'Comedy' samples
scripted_comedy_en_cnt <- sum(scripted_comedy_samples$original_language == "en")

# Get the number of 'Comedy' samples
scripted_comedy_cnt <- nrow(scripted_comedy_samples)

# Print p_hat (proportion of 'en' in 'Comedy' samples)
#cat("p_hat:", scripted_comedy_en_cnt/scripted_comedy_cnt, "\n")

# Assumed population proportion under the null hypothesis
population_proportion <- 0.4

#run a one sample proportion test
test_results <- prop.test(scripted_comedy_en_cnt, scripted_comedy_cnt, p = population_proportion, alter

print(test_results$statistic);
```

```
## X-squared
##  4.294033
```

```
print(test_results$p.value);
```

```
## [1] 0.01912317
```

Result:

X-squared = 4.294 and p_value = 0.01912

Since p_value is lesser than α (0.05) we reject the the null hypothesis H_0.

We have enough evidence to conclude that proportion comedy shows streamed in English language is not greater than or ewqual to 40%

ii. Test the claim that approximately 30% or greater of the total action and adventure shows are streamed in english

Consider the null hypothesis to be as follows $H_0 : p \geq p_0$. In the context of the question, null hypothesis states that proportion of 30% (or greater) Action and Adventure shows are streamed in English language

As such the alternate hypothesis would be $H_1 : p < p_0$. In the context of the question, alternate hypothesis states hat proportion of lesser than 30% of Action and Adventure shows are streamed in English language

Here

p_o is 3/10 (or 0.3)

Significance level of the test $\alpha = 5\%$

```r
# For action and adventure
# Filter 'expanded_scripted_samples' for 'genres' equal to "Action & Adventure"
scripted_actadv_samples <- expanded_scripted_samples[expanded_scripted_samples$genres == "Action & Adve

# Count occurrences of 'en' in 'original_language' for 'Action & Adventure' samples
scripted_actadv_en_cnt <- sum(scripted_actadv_samples$original_language == "en")

# Get the number of 'Action & Adventure' samples
scripted_actadv_cnt <- nrow(scripted_actadv_samples)

# Print p_hat (proportion of 'en' in 'Action & Adventure' samples)
#cat("p_hat:", scripted_actadv_en_cnt/scripted_actadv_cnt, "\n")

# Assumed population proportion under the null hypothesis
population_proportion <- 0.3

#run a one sample proportion test
test_results <-prop.test(scripted_actadv_en_cnt, scripted_actadv_cnt, p = population_proportion, alterna
print(test_results$statistic);
```

```
## X-squared
## 0.7220476
```

```r
print(test_results$p.value);
```

```
## [1] 0.8022635
```

Result:

X-squared = 0.7220476 and p_value = 0.8022635

Since p_value is greater than α (0.05) we fail to reject the the null hypothesis H_0.

We have enough evidence to support the calim that proportion of 30% (or greater) Action and Adventure shows are streamed in English language

iii. Test the claim that approximately 35% or greater of the total Sci-Fi & Fantasy shows are streamed in english

Consider the null hypothesis to be as follows $H_0 : p \geq p_0$. In the context of the question, null hypothesis states that proportion of 35% (or greater) shows are streamed in English language

As such the alternate hypothesis would be $H_1 : p < p_0$. In the context of the question, alternate hypothesis states hat proportion of lesser than 35% of shows are streamed in English language

Here

p_o is 7/20 (or 0.35)

Significance level of the test $\alpha = 5\%$

```r
# For Sci-Fi & Fantasy
# Filter 'expanded_scripted_samples' for 'genres' equal to "Sci-Fi & Fantasy"
scripted_scifi_samples <- expanded_scripted_samples[expanded_scripted_samples$genres == "Sci-Fi & Fantas
```

```r
# Count occurrences of 'en' in 'original_language' for 'Sci-Fi & Fantasy' samples
scripted_scifi_en_cnt <- sum(scripted_scifi_samples$original_language == "en")

# Get the number of 'Sci-Fi & Fantasy' samples
scripted_scifi_cnt <- nrow(scripted_scifi_samples)

# Print p_hat (proportion of 'en' in 'Sci-Fi & Fantasy' samples)
# cat("p_hat:", scripted_scifi_en_cnt/scripted_scifi_cnt, "\n")

# Assumed population proportion under the null hypothesis
population_proportion <- 0.35

#run a one sample proportion test
test_results <- prop.test(scripted_scifi_en_cnt, scripted_scifi_cnt, p = population_proportion, alterna

print(test_results$statistic);
```

```
##  X-squared
## 0.01334423
```

```r
print(test_results$p.value);
```

```
## [1] 0.4540176
```

Result:

X-squared = 0.01334423 and p_value = 0.4540176

Since p_value is greater than α (0.05) we fail to reject the the null hypothesis H_0.

We have enough evidence to conclude that the ratio of Netflix shows that proportion of 35% (or greater) scripted scifi shows are streamed in English language

4 Inference about two proportions

4A Is the proportion of animation (genre) series in "in_production" greater in Japan than the US?

Consider the null hypothesis to be as follows H_0 : p_1 - $p_2 \geq 0$. In the context of the question, null hypothesis states that the proportion of animation (genre) series in "in_production" greater (or equal) in Japan than the US

As such the alternate hypothesis would be H_1 : p_1 - $p_2 < 0$. In the context of the question, alternate hypothesis states that proportion of animation (genre) series in "in_production" lesser in Japan than the US

Here

p_1 is the proportion of animation TV shows in Japan

p_2 is the proportion of animation TV shows in USA

Significance level of the test $\alpha = 5\%$

```r
# Subset 'genres', 'in_production', and 'origin_country' columns
data_4a <- sample_data[, c("genres", "in_production", "origin_country")]

# Remove rows with NA values
data_4a <- na.omit(data_4a)

# Reset row indices
row.names(data_4a) <- NULL

# Filter rows where 'in_production' is TRUE
in_prod_samples <- data_4a[data_4a$in_production == TRUE, ]

# Reset row indices
row.names(in_prod_samples) <- NULL

#dim(in_prod_samples)

# Call the function 'expand_col1_by_col2' on 'in_prod_samples'
expanded_in_prod_samples <- expand_col1_by_col2(in_prod_samples, "origin_country", "genres")
#dim(expanded_in_prod_samples)

# Call the function 'expand_col1_by_col2' on 'expanded_in_prod_samples'
expanded_in_prod_samples <- expand_col1_by_col2(expanded_in_prod_samples, "genres", "origin_country")
#dim(expanded_in_prod_samples)

us_in_prod_samples <- expanded_in_prod_samples[expanded_in_prod_samples$origin_country == "US", ]

genre_counts <- table(us_in_prod_samples$genres)
#print(genre_counts)

jp_in_prod_samples <- expanded_in_prod_samples[expanded_in_prod_samples$origin_country == "JP", ]

genre_counts <- table(jp_in_prod_samples$genres)
#print(genre_counts)

in_prod_animation_samples <- expanded_in_prod_samples[expanded_in_prod_samples$genres == "Animation", ]
in_prod_animation_us_cnt <- sum(in_prod_animation_samples$origin_country == "US")
in_prod_us_cnt <- sum(expanded_in_prod_samples$origin_country == "US")
in_prod_animation_jp_cnt <- sum(in_prod_animation_samples$origin_country == "JP")
in_prod_jp_cnt <- sum(expanded_in_prod_samples$origin_country == "JP")

# Calculate proportions
p_hat1 <- in_prod_animation_us_cnt / in_prod_us_cnt
p_hat2 <- in_prod_animation_jp_cnt / in_prod_jp_cnt

# Output proportions
#print(paste("p_hat1:", p_hat1))
#print(paste("p_hat2:", p_hat2))

# Perform two-sample proportion test
test_results <- prop.test(c(in_prod_animation_us_cnt, in_prod_animation_jp_cnt), c(in_prod_us_cnt, in_pr

# Output the results
```

```r
print(test_results$statistic);
```

```
## X-squared
##  496.1043
```

```r
print(test_results$p.value);
```

```
## [1] 3.3464e-110
```

Result:

X-squared = 496.1043 and p_value = 3.3464e-110

Since p_value is less than α (0.05) we reject the the null hypothesis H_0.

We have enough evidence to conclude that the proportion of animation (genre) series in "in_production" is not greater than (or equal to) in Japan when compared to the US

4B Proportion of shows has increased (or decreased) after 2020 (post covid) in major streaming platform named Netflix

Consider the null hypothesis to be as follows $H_0 : p_1 - p_2 \geq 0$. In the context of the question, null hypothesis states that the proportion of shows has increased (or remained the same) after covid.

As such the alternate hypothesis would be $H_1 : p_1 - p_2 < 0$. In the context of the question, alternate hypothesis states that the proportion of shows has decreased after covid

Here

p_1 is the proportion of shows post covid era (started in 2021 or later)

p_2 is the proportion of shows pre covid era (ended in 2019 or earlier)

Significance level of the test $\alpha = 5\%$

```r
data_4b <- subset(sample_data, select = c("first_air_date", "last_air_date", "networks"))

data_4b <- data_4b[complete.cases(data_4b), ]

data_4b <- data.frame(data_4b)  # Ensure data_4b is a dataframe
rownames(data_4b) <- NULL  # Reset row names

add_col_df <- add_pre_post_covid(data_4b)
#head(add_col_df)
#dim(add_col_df)

# colnames(add_col_df)

net_covid_stat_samples <- expand_col1_by_col2(add_col_df, "Covid_Type", "networks")
#dim(net_covid_stat_samples)

#head(net_covid_stat_samples)

# Filter rows where 'networks' is 'Netflix' and select 'Covid Type'
netflix_covid_stats <- net_covid_stat_samples[net_covid_stat_samples$networks == "Netflix", "Covid_Type
```

```r
# Count occurrences of each unique value in 'Covid_Type'
covid_type_counts <- table(netflix_covid_stats)
#print(covid_type_counts)

# Count occurrences of 'Pre Covid' and 'Post Covid'
pre_covid_netflix_cnt <- sum(netflix_covid_stats == "Pre Covid")
post_covid_netflix_cnt <- sum(netflix_covid_stats == "Post Covid")
total_netflix_cnt <- length(netflix_covid_stats)

# Calculate proportions
p_hat1 <- pre_covid_netflix_cnt / total_netflix_cnt
p_hat2 <- post_covid_netflix_cnt / total_netflix_cnt

# Output proportions
# print(paste("p_hat1:", p_hat1))
# print(paste("p_hat2:", p_hat2))

# Perform two-sample proportion test
test_results <- prop.test(c(pre_covid_netflix_cnt, post_covid_netflix_cnt),c(total_netflix_cnt, total_n

# Output the results
print(test_results$statistic);
```

```
## X-squared
##  155.9218
```

```r
print(test_results$p.value);
```

```
## [1] 4.402856e-36
```

Result:

X-squared = 155.9218 and p_value = 4.402856e-36

Since p_value is less than α (0.05) we reject the the null hypothesis H_0.

We have enough evidence to conclude that the proportion of shows has has not increased (or remained the same) after covid.

5 Chi Square Inference - Goodness of fit test

```r
expand_col <- function(col) {
  col_exp_vals = c()
  col_vals <- unlist(strsplit(trimws(col), ","))
  for (val in col_vals) {
      col_exp_vals = append(col_exp_vals, trimws(val))
  }
  return(col_exp_vals)
}
```

```
expand_col1_by_col2_col3 = function(df, col1, col2, col3){
  col1_exp_vals <- c()
  col2_exp_vals <- c()
  col3_exp_vals <- c()

  for (i in 1:nrow(df)){
    df_row <- df[i, , drop = FALSE]
    col1_val <- df_row[[col1]]
    if (nchar(df_row[[col2]]) > 0){
      col2_vals <- unlist(strsplit(df_row[[col2]], ","))
      for (col2_val in col2_vals) {
        if (nchar(df_row[[col3]]) > 0){
          col3_vals <- unlist(strsplit(df_row[[col3]], ","))
          for (col3_val in col3_vals) {
            col1_exp_vals = append(col1_exp_vals, col1_val)
            col2_exp_vals = append(col2_exp_vals, trimws(col2_val))
            col3_exp_vals = append(col3_exp_vals, trimws(col3_val))
          }
        }
      }
    }
  }
  column_names <- c(col1, col2, col3)
  my_dataframe <- setNames(data.frame(col1_exp_vals, col2_exp_vals, col3_exp_vals), column_names)
  return(my_dataframe)
}

test_df2 = data.frame(
  "c1" = c(1     , 2    ,  3    , 4 , 5),
  "c2" = c("a, b", "c"  , "i"   , "x", ""),
  "c3" = c("d"   , "e,f", "g,h ", "", NA)
)
test_df2 = na.omit(test_df2)
test_df2
```

```
##   c1   c2   c3
## 1  1 a, b    d
## 2  2    c  e,f
## 3  3    i g,h
## 4  4    x
```

```
expand_col1_by_col2_col3(test_df2, "c1", "c2", "c3")
```

```
##   c1 c2 c3
## 1  1  a  d
## 2  1  b  d
## 3  2  c  e
## 4  2  c  f
## 5  3  i  g
## 6  3  i  h
```

```r
get_duration2 <- function(data) {
  # Remove rows with missing values in "first_air_date" and "last_air_date"
  data <- data[complete.cases(data$first_air_date) & complete.cases(data$last_air_date), ]
  # Convert date columns to Date objects if needed
  if (!inherits(data$first_air_date, "Date")) {
    data$first_air_date <- as.Date(data$first_air_date, format="%d/%m/%Y")
  }
  if (!inherits(data$last_air_date, "Date")) {
    data$last_air_date <- as.Date(data$last_air_date, format="%d/%m/%Y")
  }

  # Remove rows where first_air_date is greater than last_air_date
  # cat("Removing rows where first_air_date is greater than last_air_date...\n")
  data <- data[data$first_air_date <= data$last_air_date, ]

  # Calculate the duration between the first and last air dates
  data$duration <- as.numeric(difftime(data$last_air_date, data$first_air_date, units = "weeks"))

  # Filter out rows with non-positive duration
  data <- data[data$duration > 0, ]

  return(data)
}
```

```r
expand_col1_by_col2 <- function(df, col1, col2) {
  col1_exp_vals = c()

  col2_exp_vals = c()
  for (i in 1:nrow(df)){
    df_row <- df[i, , drop = FALSE]
    col1_val <- df_row[[col1]]
    if (nchar(df_row[[col2]]) > 1){
      col2_vals <- unlist(strsplit(df_row[[col2]], ","))

      for (exp_val in col2_vals) {

        # print(col2_exp_vals)
        col1_exp_vals = append(col1_exp_vals, col1_val)# append(col1_exp_vals, trimws(col1_val))
        col2_exp_vals = append(col2_exp_vals, trimws(exp_val))# append(col2_exp_vals, trimws(col2_val))
      }
    }
  }
  column_names <- c(col1, col2)

  my_dataframe <- setNames(data.frame(col1_exp_vals, col2_exp_vals), column_names)
  return(my_dataframe)
}
```

5a Let's say a sample drawn randomly from our sample data has the following result of language distribution - English 40%, Japanese 12%, Chinese 9%, Korean 9%, Others 30%. What can you infer from this result?

Consider the null hypothesis to be as follows H_0 : $English = 40\%$, $Japanese = 12\%$, $Chinese = 9\%$, $Korean = 9\%$, $Others = 30\%$

As such the alternate hypothesis would be H_1 : H_0^c (At least one in H_0 is incorrect).

Here

Significance level of the test $\alpha = 5\%$

```
original_lang_data = sample_data$original_language
total_count = length(original_lang_data)
language_counts <- table(original_lang_data)
english_count = language_counts["en"]
japanese_count = language_counts["ja"]
chinese_count = language_counts["zh"]
korean_count = language_counts["ko"]
other_count = total_count - english_count - japanese_count - chinese_count - korean_count
cat("eng cnt:", english_count, "\njap cnt:", japanese_count, "\nchi cnt:", chinese_count, "\nkor cnt:",
```

```
## eng cnt: 74619
## jap cnt: 13588
## chi cnt: 12940
## kor cnt: 7489
## oth cnt: 53222
```

```
chisq.test(c(english_count, japanese_count, chinese_count, korean_count, other_count), p=c(0.4, 0.12, 0
```

```
##
##  Chi-squared test for given probabilities
##
## data:  c(english_count, japanese_count, chinese_count, korean_count,     other_count)
## X-squared = 7328.5, df = 4, p-value < 2.2e-16
```

Result:

t_statistic = 7328.525 and p_value = (Very close to 0)

Since p_value is less than α (0.05) we reject the null hypothesis H_0.

We have enough evidence to conclude that atleast one value in H_0 is in accurate.

Q5b Conside the claim with respect to distribution of genere. Expected Genre distribution - Drama 30%, Comedy 20 %, Documentary 15%, Animation 10%, Others 25%. Can you conclude if this claim is what our sample supports?

Consider the null hypothesis to be as follows H_0 : $Drama = 30\%$, $Comedy = 20\%$, $Documentary = 15\%$, $Animation = 10\%$, $Others = 25\%$

As such the alternate hypothesis would be H_1 : H_0^c (At least one in H_0 is incorrect).

Here

Significance level of the test $\alpha = 5\%$

```
genre_data = sample_data$genres
clean_genre_data = na.omit(genre_data)

exp_genre_data = expand_col(clean_genre_data)
```

```
genres_counts <- table(exp_genre_data)

total_genres_dcount <- length(exp_genre_data)

drama_count <- genres_counts["Drama"]
comedy_count <- genres_counts["Comedy"]
documentary_count <- genres_counts["Documentary"]
actadv_count <- genres_counts["Action & Adventure"]
other_genre_count <- total_genres_dcount - drama_count - comedy_count - documentary_count - actadv_coun

cat("dra cnt:", drama_count, "\ncom cnt:", comedy_count, "\ndoc cnt:", documentary_count, "\nact cnt:",
```

```
## dra cnt: 32775
## com cnt: 22757
## doc cnt: 20998
## act cnt: 7132
## oth cnt: 59972
```

```
chisq.test(c(drama_count, comedy_count, documentary_count, actadv_count, other_genre_count), p=c(0.3, 0
```

```
##
##  Chi-squared test for given probabilities
##
## data:  c(drama_count, comedy_count, documentary_count, actadv_count,     other_genre_count)
## X-squared = 23490, df = 4, p-value < 2.2e-16
```

Result:

t_statistic = 23490 and p_value = (Very close to 0)

Since p_value is less than α (0.05) we reject the null hypothesis H_0.

We have enough evidence to conclude that atleast one value in H_0 is in accurate.

5c Consider the claim Expected Network distribution - BBC One 8%, Netflix 7%, Fuji TV 3.5%, TV Tokyo 3.5%, Prime Video 3%, Others 75% for the population. Can we conclude if this claim is accurate?

Consider the null hypothesis to be as follows H_0 : $BBCOne = 8\%$, $Netflix = 7\%$, $Fuji\ TV = 3.5\%$, $TV\ Tokyo = 3.5\%$, $Prime\ Video = 3\%$, $Others = 75\%$

As such the alternate hypothesis would be H_1 : H_0^c (At least one in H_0 is incorrect).

Here

Significance level of the test $\alpha = 5\%$

```r
networks_data = sample_data$networks
clean_networks_data = na.omit(networks_data)

  exp_networks_data = expand_col(clean_networks_data)
```

```r
networks_counts <- table(exp_networks_data)

total_networks_dcount <- length(exp_networks_data)

bbc_one_count <- networks_counts["BBC One"]
netflix_count <- networks_counts["Netflix"]
fuji_tv_count <- networks_counts["Fuji TV"]
tv_tokyo_count <- networks_counts["TV Tokyo"]
prime_video_count <- networks_counts["Prime Video"]
total_networks_dcount <- total_networks_dcount - bbc_one_count - netflix_count - fuji_tv_count - tv_toky

cat("bbc 1 cnt:", bbc_one_count, "\nnetfl cnt:", netflix_count, "\nfujtv cnt:", fuji_tv_count, "\ntvtok
```

```
## bbc 1 cnt: 2252
## netfl cnt: 1890
## fujtv cnt: 1054
## tvtok cnt: 1026
## privd cnt: 877
## other cnt: 101056
```

```r
chisq.test(c(bbc_one_count, netflix_count, fuji_tv_count, tv_tokyo_count, prime_video_count, total_netwo
```

```
##
##  Chi-squared test for given probabilities
##
## data:  c(bbc_one_count, netflix_count, fuji_tv_count, tv_tokyo_count,     prime_video_count, total_n
## X-squared = 19609, df = 5, p-value < 2.2e-16
```

```r
head(sort(table(exp_genre_data), decreasing = T), 20)
```

```
## exp_genre_data
##              Drama             Comedy        Documentary          Animation
##              32775              22757              20998              11730
##            Reality Action & Adventure              Crime             Family
##              10572               7132               6698               6333
##   Sci-Fi & Fantasy            Mystery              Kids               Talk
##               6252               5132               4200               3093
##               Soap     War & Politics               News            Western
##               2005               1725               1599                335
##            Romance              Music            History            Musical
##                183                 70                 44                  1
```

Result:

Z-squared_statistic = 19609 and p_value = Very close to 0

Since p_value is less than α (0.05) we reject the null hypothesis H_0.

We have enough evidence to conclude thaat at least one condition in H_0 is incorrect

6 Chi squared inference

6A Is there a possible relationship between In-production column and Status column

Consider the null hypothesis to be as follows H_0 : Columns have a possible relation between them

As such the alternate hypothesis would be H_1 : H_0^c (No relation)

Here

Significance level of the test $\alpha = 5\%$

```
in_prod_data = sample_data$in_production
status_data = sample_data$status
```

```
contingency_table <- table(in_prod_data, status_data)
print(contingency_table)
```

```
##             status_data
## in_prod_data Canceled Ended In Production Pilot Planned Returning Series
##        FALSE     4375 91738             0     0       0                 0
##        TRUE         0     0          2030   228     546             62941
```

```
result <- chisq.test(contingency_table)
print(result)
```

```
##
##  Pearson's Chi-squared test
##
## data:  contingency_table
## X-squared = 161858, df = 5, p-value < 2.2e-16
```

Result:

t_statistic = 161858 and p_value = Very close to 0

Since p_value is less than α (0.05) we reject the null hypothesis H_0.

We have enough evidence to conclude that there is no relation between the two columns

6b If there is degree of assosiation between Production_companies vs network

Consider the null hypothesis to be as follows H_0 : Columns have a possible relation between them

As such the alternate hypothesis would be H_1 : H_0^c (No relation)

Here

Significance level of the test $\alpha = 5\%$

```
data_6b <- sample_data[, c("production_companies", "networks")]
data_6b <- na.omit(data_6b)
```

```
exp_net_data <- expand_col1_by_col2(data_6b, "production_companies", "networks")
exp_prod_data <- expand_col1_by_col2(exp_net_data, "networks", "production_companies")
```

```r
contingency_table <- table(exp_prod_data$networks, exp_prod_data$production_companies)
result <- chisq.test(contingency_table)
print(result)
```

```
##
##  Pearson's Chi-squared test
##
## data:  contingency_table
## X-squared = 100713155, df = 49464020, p-value < 2.2e-16
```

Result:

X_square_statistic = 100713155 and p_value = Very close to 0

Since p_value is less than α (0.05) we reject the null hypothesis H_0.

We have enough evidence to conclude that there is no relation between the two columns

6e To check if there is independence between the columns production_country and country of origin

Consider the null hypothesis to be as follows H_0 : Columns have a possible relation between them

As such the alternate hypothesis would be H_1 : H_0^c (No relation)

Here

Significance level of the test $\alpha = 5\%$

```r
data_6e <- sample_data[, c("production_countries", "origin_country")]
data_6e <- na.omit(data_6e)
```

```r
exp_prod_count <- expand_col1_by_col2(data_6e, "production_countries", "origin_country")
```

```r
exp_origin_count <- expand_col1_by_col2(exp_prod_count, "origin_country", "production_countries")
```

```r
contingency_table <- table(exp_origin_count$origin_country, exp_origin_count$production_countries)
result <- chisq.test(contingency_table)
print(result)
```

```
##
##  Pearson's Chi-squared test
##
## data:  contingency_table
## X-squared = 6790303, df = 26026, p-value < 2.2e-16
```

```r
#table(exp_origin_count$production_countries)
# exp_origin_count
```

Result:

X_square_statistic = 6790303 and p_value = Very close to 0

Since p_value is less than α (0.05) we reject the null hypothesis H_0.

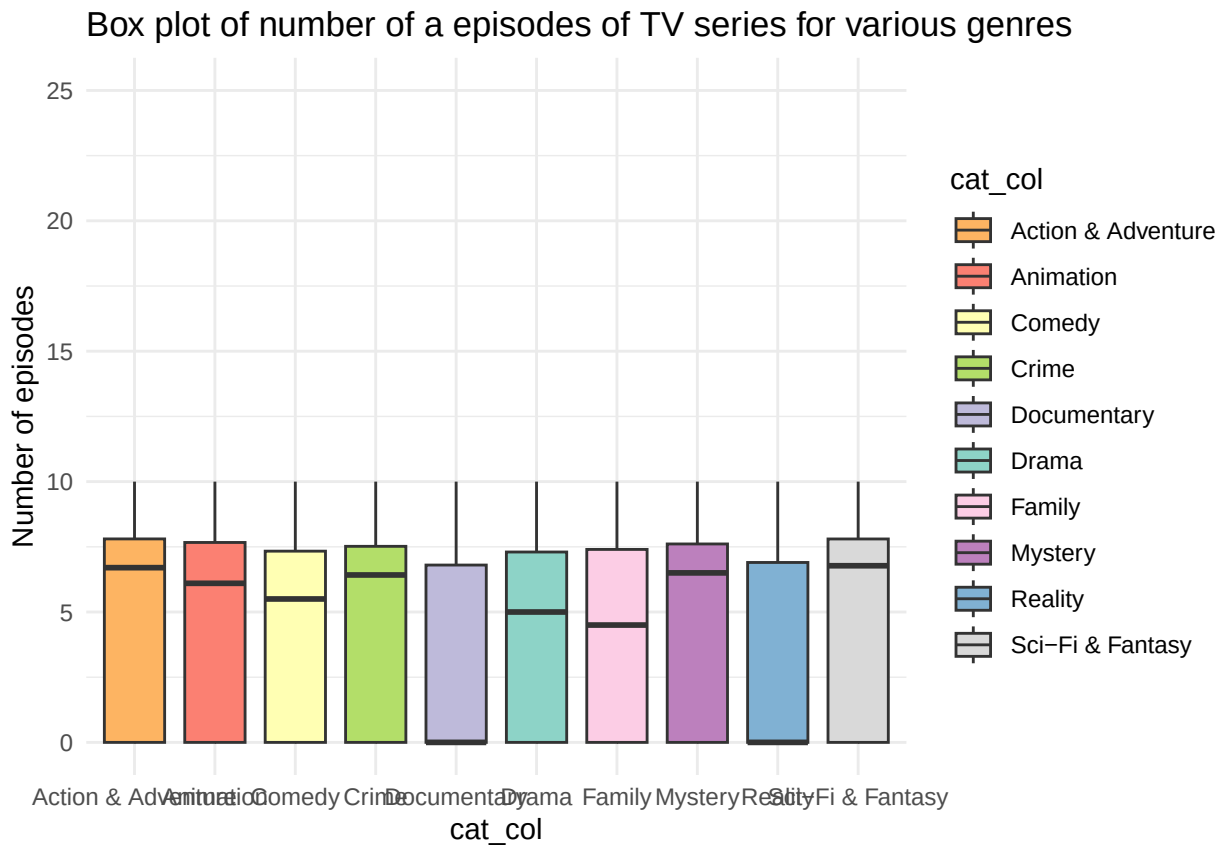
We have enough evidence to conclude that there is no relation between the two columns

7 ANOVA

7a Running ANOVA for VOte_AVERAGE by Genere for over all networks.

Demonstrating equality of variance for vote_Average and genere:

```
expanded_df <- expand_categorical_cols(sample_data,"genres","vote_average")

generate_box_plot(
  data = expanded_df,
  fill_column = "cat_col",
  y_col = "num_col",
  y_label = "Number of episodes",
  title = "Box plot of number of a episodes of TV series for various genres",
  ylim = c(0, 25)
)
```




```
data_7a <- sample_data[, c("vote_average", "type", "networks")]
data_7a <- data_7a[data_7a$vote_average > 0, ]
data_7a <- na.omit(data_7a);
# data_7a <- data_7a[data_7a$type == "Scripted", ]
# data_7a
```

```
# exp_net_avg
```

```r
exp_net_avg <- expand_col1_by_col2(data_7a, "vote_average", "networks")
# exp_net_avg
```

```r
anova_model <- aov(exp_net_avg$vote_average ~ exp_net_avg$networks)
summary(anova_model)
```

```
##                          Df Sum Sq Mean Sq F value Pr(>F)
## exp_net_avg$networks   2340  32307  13.807   4.391 <2e-16 ***
## Residuals             54235 170540   3.144
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

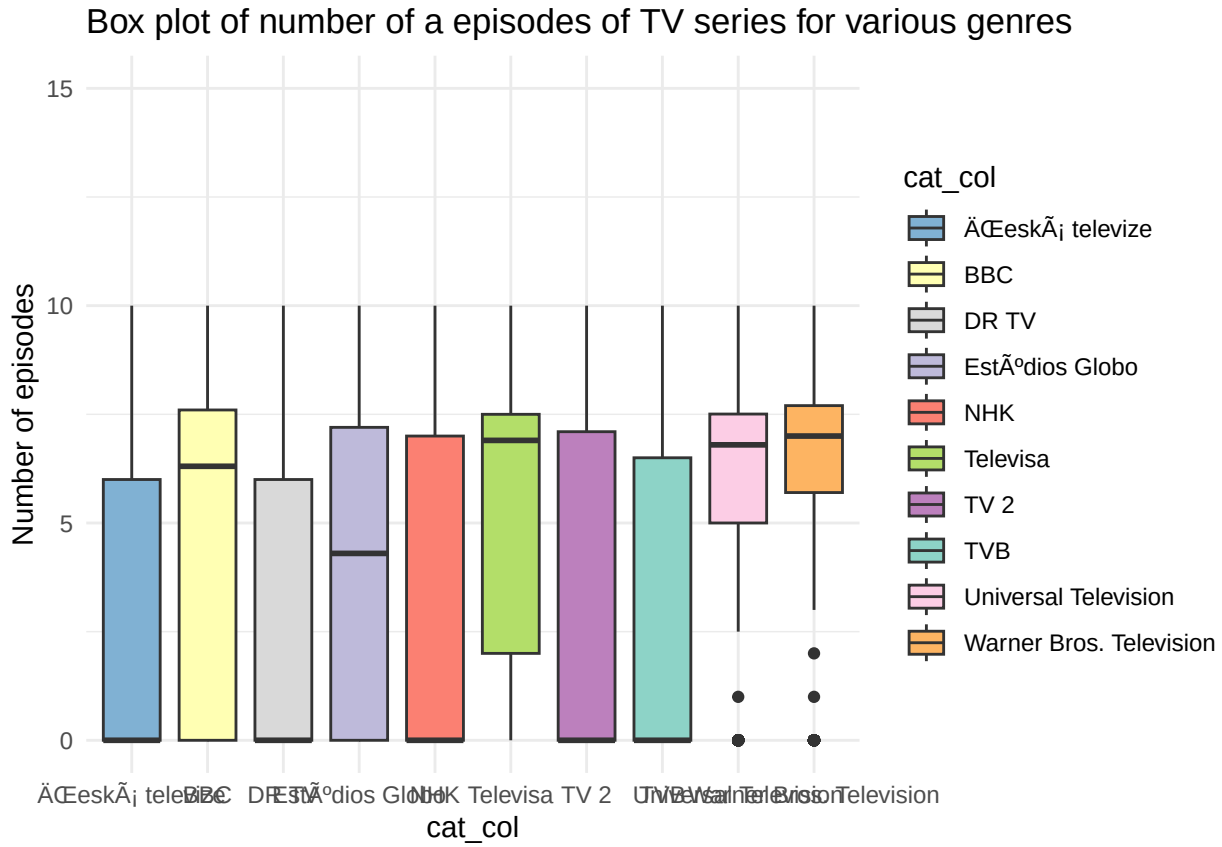
7b Two - way ANOVA - Vote_average by production_company and network

Consider null hypothesis as H_0: $\mu_1 = \mu_2 = \mu_3 \ldots$.

As such our alternalte hypothesis is H_1: H_o^c (Atleast one of the means is not equal)

Demonstrating equality of variance for vote_Average and genere:

```r
expanded_df <- expand_categorical_cols(sample_data,"production_companies","vote_average")

generate_box_plot(
  data = expanded_df,
  fill_column = "cat_col",
  y_col = "num_col",
  y_label = "Number of episodes",
  title = "Box plot of number of a episodes of TV series for various genres",
  ylim = c(0, 15)
)
```




```
data_7b <- sample_data[, c("vote_average", "production_companies", "networks")]
data_7b <- data_7b[data_7b$vote_average > 0, ]
data_7b=na.omit(data_7b)
# data_7b
```

```
exp_vavg = expand_col1_by_col2_col3(data_7b, "vote_average", "production_companies", "networks")
typeof(exp_vavg)
```

```
## [1] "list"
```

```
typeof(exp_vavg[1,1])
```

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

```
anova2way_model <- aov(vote_average ~ production_companies * networks, data = exp_vavg[1:325, ])
summary(anova2way_model)
```

```
##                               Df Sum Sq Mean Sq F value Pr(>F)
## production_companies          121 20.125 0.16632   0.627  0.924
## networks                       83  2.446 0.02947   0.111  1.000
## production_companies:networks 103  0.515 0.00500   0.019  1.000
## Residuals                      17  4.509 0.26525
```

Result:

X_square_statistic = 6790303 and p_value = Very close to 0

Since p_values are greater than $\alpha/3$ (0.01666) we fail to reject the null hypothesis H_0.

We so not have enough evidence to reject the claim that the means are equal.

8 Inference about correlation

8a Duration v/s Number of seasons (for scripted shows)

Consider the null hypothesis to be as follows H_0: $\rho = 0$. Considering no co relation between Duration and number of seasons

As such the alternate hypothesis would be H_1 : H_0^c (co relation exists to some degree)

Here

Significance level of the test $\alpha = 5\%$

```
data_8a = sample_data[, c("last_air_date", "first_air_date", "number_of_seasons")]
data_8a = na.omit(data_8a)
data_8a
```

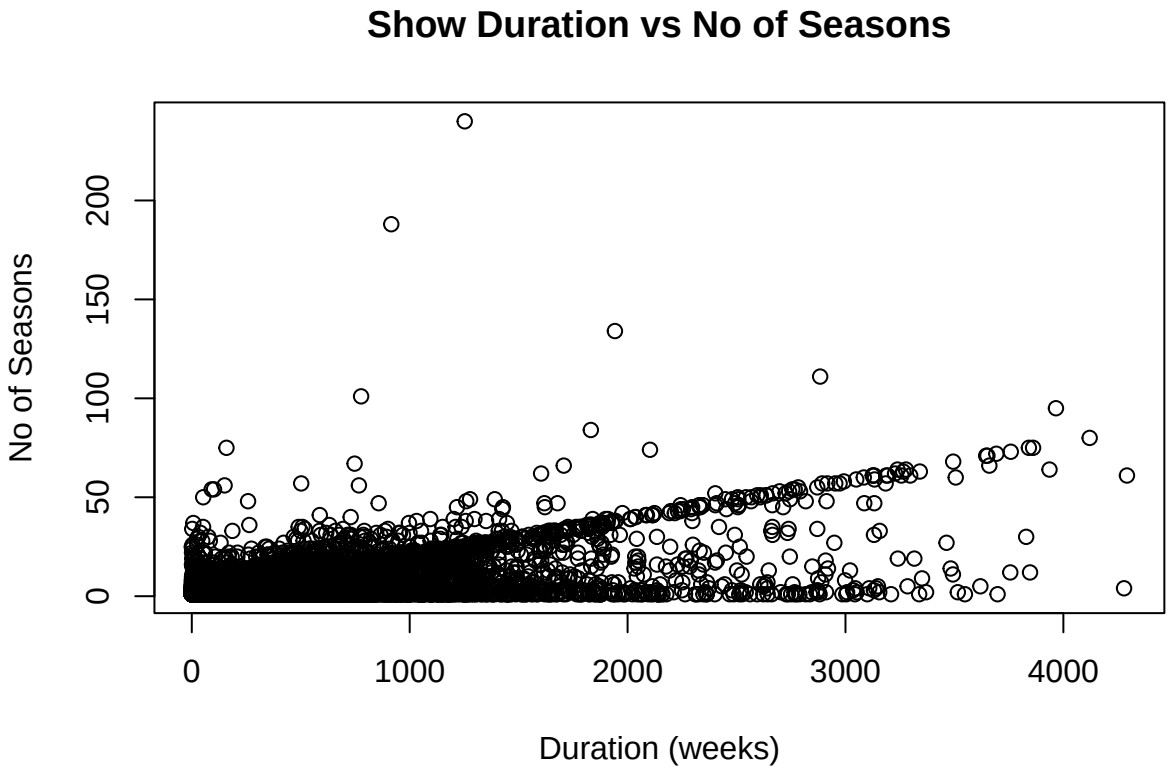
```
## # A tibble: 130,355 x 3
##    last_air_date       first_air_date      number_of_seasons
##    <dttm>              <dttm>                          <dbl>
##  1 2019-05-19 00:00:00 2011-04-17 00:00:00                 8
##  2 2021-12-03 00:00:00 2017-05-02 00:00:00                 3
##  3 2022-07-01 00:00:00 2016-07-15 00:00:00                 4
##  4 2022-11-20 00:00:00 2010-10-31 00:00:00                11
##  5 2021-09-10 00:00:00 2016-01-25 00:00:00                 6
##  6 2023-08-23 00:00:00 2017-01-26 00:00:00                 7
##  7 2021-09-17 00:00:00 2021-09-17 00:00:00                 2
##  8 2013-09-29 00:00:00 2008-01-20 00:00:00                 5
##  9 2023-05-01 00:00:00 2017-09-25 00:00:00                 6
## 10 2021-03-05 00:00:00 2021-01-15 00:00:00                 1
## # i 130,345 more rows
```

```
data_8a_new = get_duration2(data_8a)
data_8a_new = na.omit(data_8a_new)
data_8a_new
```

```
## # A tibble: 98,859 x 4
##    last_air_date first_air_date number_of_seasons duration
##    <date>        <date>                    <dbl>    <dbl>
##  1 2019-05-19    2011-04-17                    8     422
##  2 2021-12-03    2017-05-02                    3     239.
##  3 2022-07-01    2016-07-15                    4     311
##  4 2022-11-20    2010-10-31                   11     629
##  5 2021-09-10    2016-01-25                    6     294.
##  6 2023-08-23    2017-01-26                    7     343.
##  7 2013-09-29    2008-01-20                    5     297
##  8 2023-05-01    2017-09-25                    6     292
```

```
##  9 2021-03-05    2021-01-15                    1       7
## 10 2023-05-24    2014-10-07                    9     450.
## # i 98,849 more rows
```

```
plot(data_8a_new$duration, data_8a_new$number_of_seasons, main="Show Duration vs No of Seasons", xlab="
```




```
#d_Q1 = quantile(data_8a_new$duration, 0.25)
#d_Q3 = quantile(data_8a_new$duration, 0.75)
#d_IQR = d_Q3 - d_Q1
#d_upper = d_Q3 + 1.5 * d_IQR

#data_8a_new = data_8a_new[data_8a_new$duration <= d_upper, ]

#s_Q1 = quantile(data_8a_new$number_of_seasons, 0.25)
#s_Q3 = quantile(data_8a_new$number_of_seasons, 0.75)
#s_IQR = s_Q3 - s_Q1
#s_upper = s_Q3 + 1.5 * s_IQR
#s_lower = s_Q1 - 1.5 * s_IQR

#data_8a_new = data_8a_new[data_8a_new$number_of_seasons <= s_upper, ]
#data_8a_new = data_8a_new[data_8a_new$number_of_seasons >= s_lower, ]

#data_8a_new <- data_8a
```

```r
cor.test(data_8a_new$duration, data_8a_new$number_of_seasons, exact=FALSE, method = "spearman")
```

```
##
##  Spearman's rank correlation rho
##
## data:  data_8a_new$duration and data_8a_new$number_of_seasons
## S = 4.7427e+13, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.7054705
```

Result:

s_statistic = 4.7427e+13 and p_value = Very close to 0

Since p_value is less than α (0.05) we reject the null hypothesis H_0.

We have enough evidence to conclude that there is a co relation between the two columns

9 - Regression on number of seasons with duration.

Consider the null hypothesis to be as follows $\beta_1 = 0$. Considering not linearly related

As such the alternate hypothesis would be $\beta_1 \neq 0$.

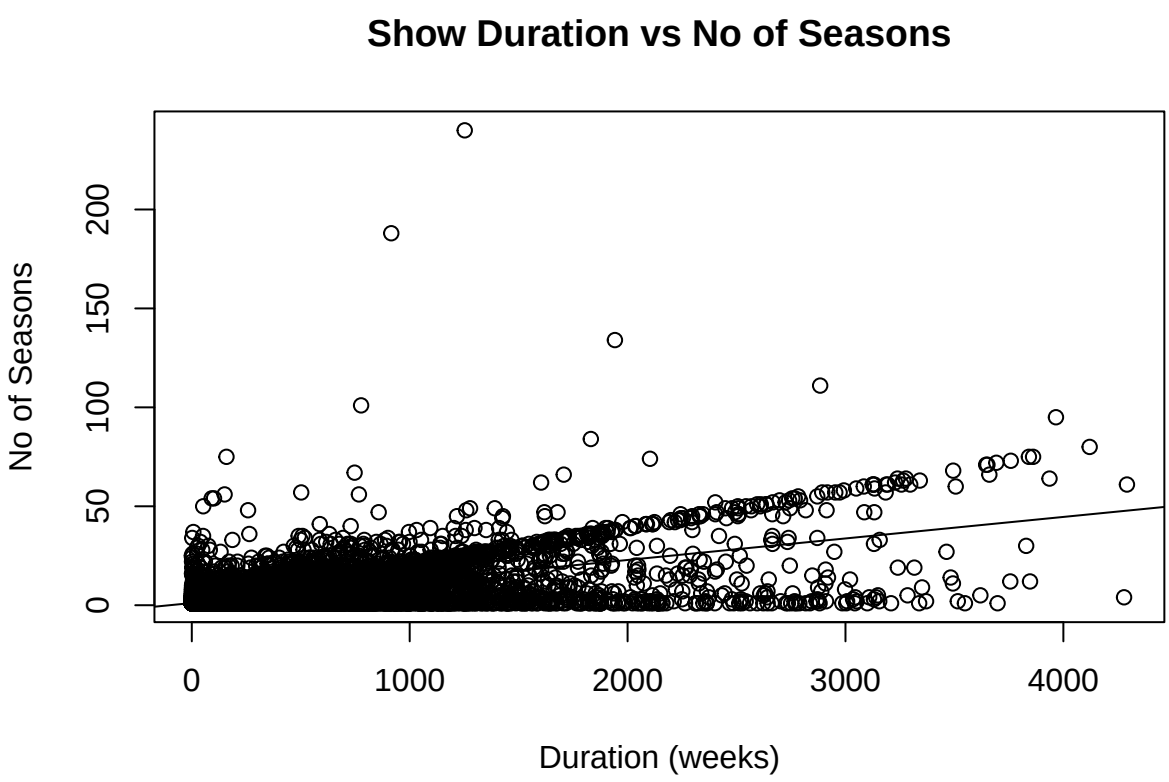
Here

Significance level of the test $\alpha = 5\%$

```r
linear_model <- lm(data_8a_new$number_of_seasons ~ data_8a_new$duration)
linear_model
```

```
##
## Call:
## lm(formula = data_8a_new$number_of_seasons ~ data_8a_new$duration)
##
## Coefficients:
##          (Intercept)  data_8a_new$duration
##              1.11444               0.01088
```

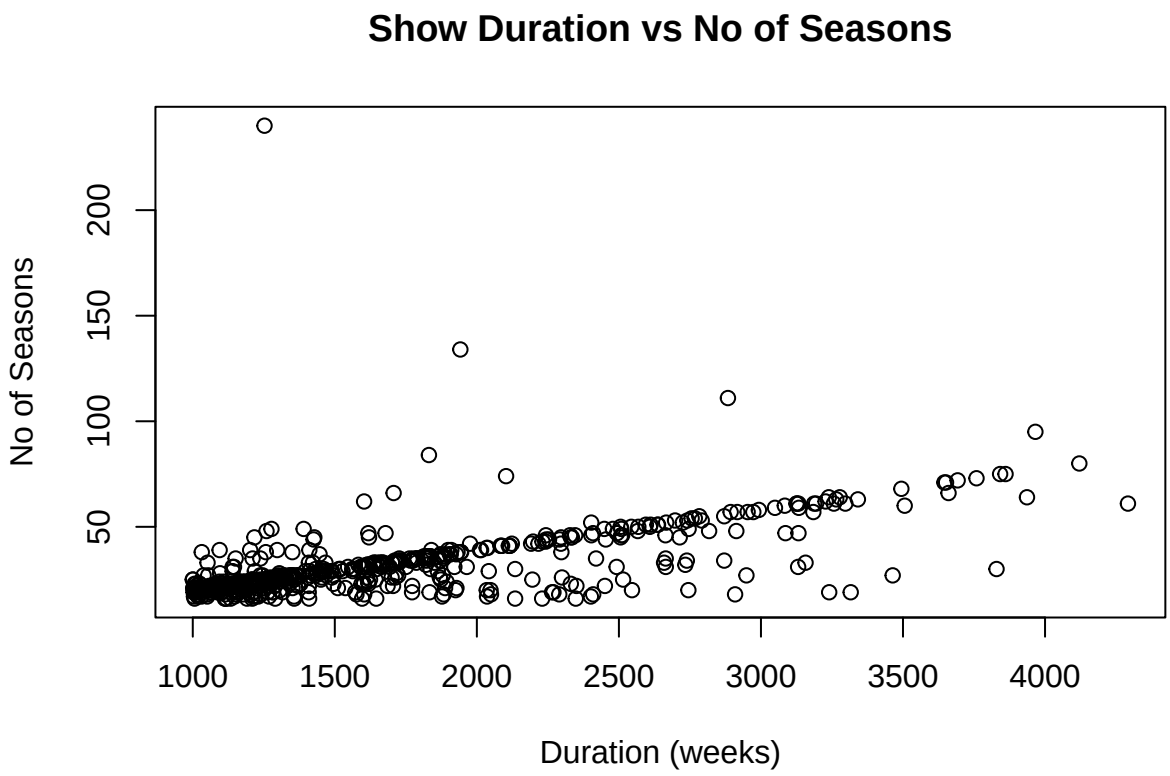
```r
plot(data_8a_new$duration, data_8a_new$number_of_seasons, main="Show Duration vs No of Seasons", xlab="
abline(linear_model)
```




```
# Filtering data with 10+ seasons and duration more than 1000 to capture the visible the linearity in t
test_data_8a = data_8a_new[data_8a_new$duration > 1000, ]
test_data_8a = test_data_8a[test_data_8a$number_of_seasons > 15, ]
test_data_8a
```

```
## # A tibble: 574 x 4
##    last_air_date first_air_date number_of_seasons duration
##    <date>        <date>                    <dbl>    <dbl>
##  1 2023-10-01    1989-12-17                   35    1763
##  2 2023-10-01    1999-10-20                   21    1250.
##  3 2023-10-01    1999-01-31                   22    1287
##  4 2023-03-29    1997-08-13                   26    1337
##  5 2023-05-18    1999-09-20                   24    1234.
##  6 2023-05-22    2003-09-23                   20    1026.
##  7 2023-03-24    1997-04-01                   25    1355.
##  8 2023-09-29    1993-08-28                   30    1570.
##  9 2022-12-18    2002-10-20                   33    1052
## 10 2015-04-15    1979-04-02                   27    1880.
## # i 564 more rows
```

```r
plot(test_data_8a$duration, test_data_8a$number_of_seasons, main="Show Duration vs No of Seasons", xlab=
```



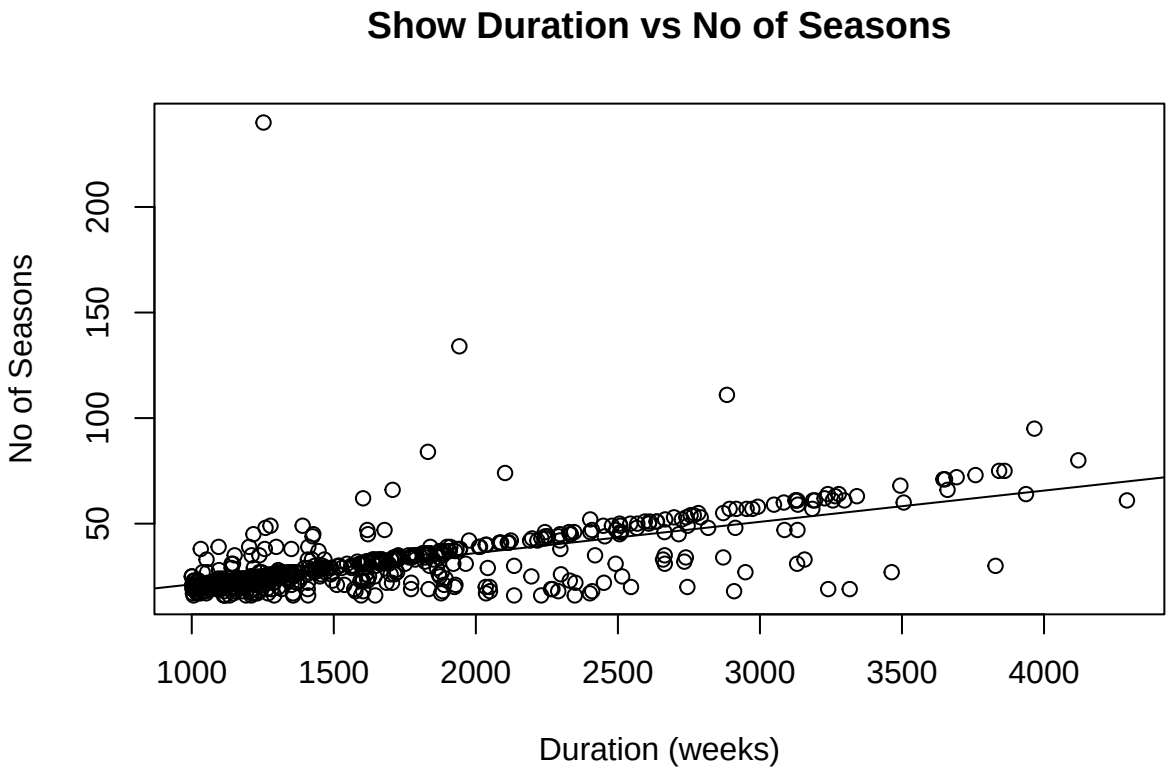

```r
cor.test(test_data_8a$duration, test_data_8a$number_of_seasons, exact=FALSE, method = "spearman")
```

```
##
##  Spearman's rank correlation rho
##
## data:  test_data_8a$duration and test_data_8a$number_of_seasons
## S = 9871265, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.6868231
```

```r
linear_model2 <- lm(test_data_8a$number_of_seasons ~ test_data_8a$duration)
linear_model2
```

```
##
## Call:
## lm(formula = test_data_8a$number_of_seasons ~ test_data_8a$duration)
##
## Coefficients:
##           (Intercept)  test_data_8a$duration
##                6.4505                 0.0148
```

```r
plot(test_data_8a$duration, test_data_8a$number_of_seasons, main="Show Duration vs No of Seasons", xlab=
abline(linear_model2)
```



Result:

s_statistic = 9871265 and p_value = Very close to 0

Since p_value is less than α (0.05) we reject the null hypothesis H_0.

We have enough evidence to conclude that there is (to a certain extent) a relationship between duration and number of seasons.