The relationship of ranking on IMDB website and Netflix data

Registration Number : 2110374

Nuttarun Kunratvej

Contents

Introduction 2

The comparison between Netflix and IMDB 4

Chi-square test 4

linear regression 5

Conclusion: 7

Appendix: 8

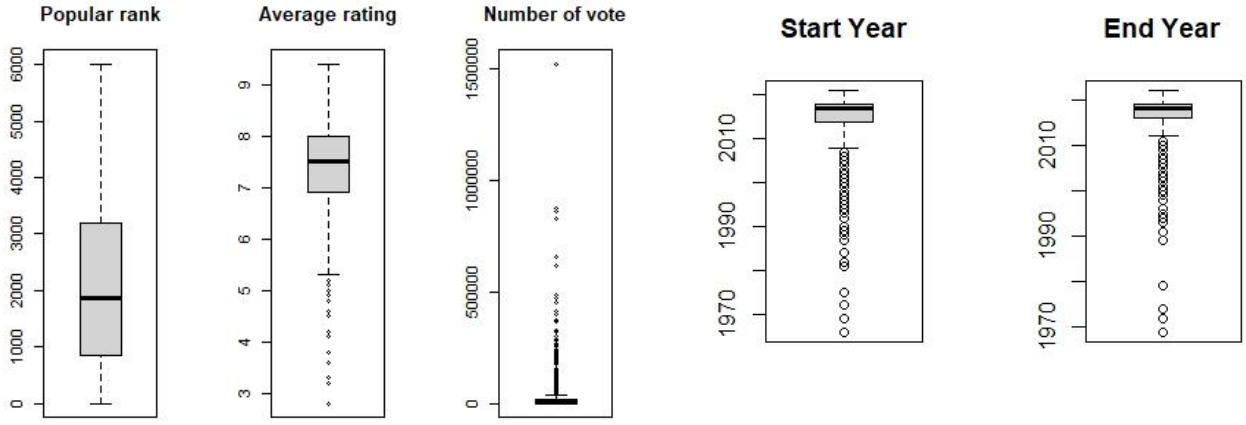
Introduction

The dataset contains a list of the movies and TV series on Netflix which are rated on the IMDB website. These consist of many variables, but in this report will be using some of them as follow;

- title : Title of the show.
- popular_rank : Ranking as given by IMDB when filtered by popularity.
- certificate : Contains the age certifications received by the show. Many null values.
- startYear : When the show was first broadcasted.
- endYear : Year of show ending
- type : The type of the show i.e TV series, Movies, TV mini-series.
- language : Language of the show.
- rating : Average rating given to the show.
- numVotes : Number of votes received by the show.

The dataset come from Kaggle website which can be access by using this url
https://www.kaggle.com/datasets/snehaanbhawal/netflix-tv-shows-and-movie-list

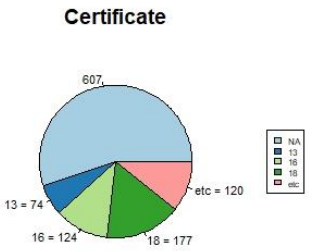
Continuous variables:

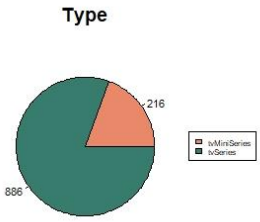


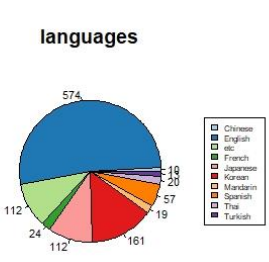
The boxplot above shows the summary of continuous variables; popular rank, rating and number of votes on Netflix. The data have a maximum of 7008, but the choosing data was only a thousand due to the NA (non-available) data and this caused the mean to be 2114.8, not the actual mean before taking NA outs. While rating has almost the same value of mean and median; this means it is the normal distribution graph with some of the outliers. The last column is the number of votes on Netflix's website, the values seem to be skewed and have lots of differences

between minimum and maximum, also the outliers that might affect the response variables as there are lots of outliers that may come from the NA value that has been avoided in the first place and also the human error since the number of votes came from Netflix own customers which they can decide to vote the show after watching or not some might skip this part even they were attracted by the show. The start and end years are similar to each other since movies and series would end in the same year or nearby, but there are lots of outliers which means lots of shows were rearranged and broadcast again.

Categories variables:



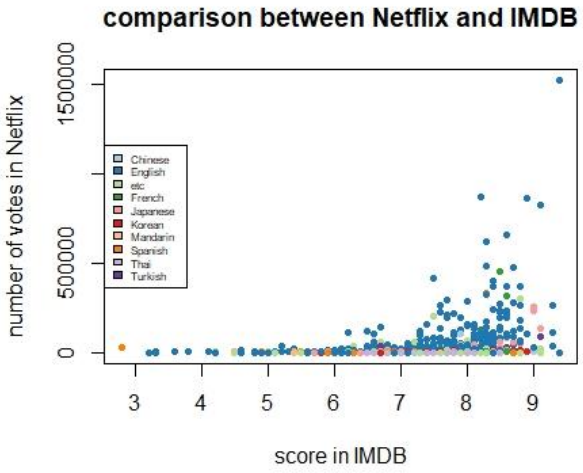




While others are category variables with lots of levels so this report will show only some part of it. As the pie chart shows above, lots of the shows did not show the certificate as some of the shows were released a long time ago so some of them might not have the certificate to suggest to the audience which can be seen in the year start boxplot, about a half was published before 2014. As they are many types of shows on Netflix, but the NA value occurs and was taken out so the result was left out only series and mini-series plus lots of the shows were in English languages and this might be the main prediction of the ranking on IMDB website which the report will do the regression to find the answer.

As the information shown above, the main point of interest is finding the prediction variables that cause the ranking of the IMDB website and the relationship between the rating, ranking on IMDB, the number of votes on Netflix and language whether consistent or not.

The comparison between Netflix and IMDB



The graph above demonstrates the relationship between the number of votes on Netflix and IMDB which divide into the languages. Apparently, the relationship between the two platforms seems to go in the same pattern, but there are some low number of votes on Netflix with the high score on the IMDB website as there can be an error that refers to the explanation in boxplot. The language that has a high value on the two websites is English due to the English language is the universal language. Consequently, it is possible to have a high score in the English language and can pull in people interested.

However, ranking in IMDB website is different column to the average score of the shows so chi-square test would be use to predicted whether there is some relationship between language and ranking or not.

Chi-square test

	Chinese	English	etc	French	Japanese
high	20.00	69.69	33.93	62.50	44.64
low	80.00	30.31	66.07	37.50	55.36

	Korean	Mandarin	Spanish	Thai	Turkish	Sum
high	35.40	26.32	50.88	5.00	84.62	55.17
low	64.60	73.68	49.12	95.00	15.38	44.83

If there is no relationship between ranking and language, every high ranking should be around 55.17% same as low ranking should be 44.83% which mean there are some statistical significant between these two categories with almost 0% of p-values. Nevertheless, this dataset has very small values in some expected values so the approximation could be wrong so the simulate.p.value=TRUE code is the procedure to solve this problem.

linear regression

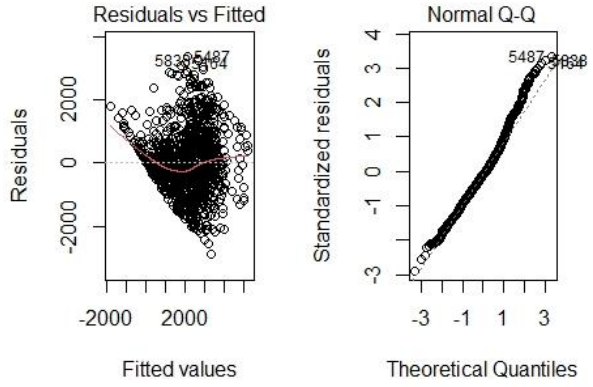
Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)	
(Intercept)	6.738e+03	1.250e+04	0.539	0.590043	
certificate_113	-9.774e+02	1.332e+02	-7.338	4.24e-13	***
certificate_116	-1.230e+03	1.084e+02	-11.350	< 2e-16	***
certificate_118	-1.521e+03	9.017e+01	-16.864	< 2e-16	***
certificate_1others	-1.213e+03	1.145e+02	-10.594	< 2e-16	***
lan_1English	-1.447e+03	3.258e+02	-4.443	9.78e-06	***
lan_1etc	-6.852e+02	3.365e+02	-2.036	0.041992	*
lan_1French	-8.959e+02	3.848e+02	-2.328	0.020093	*
lan_1Japanese	-9.558e+02	3.367e+02	-2.839	0.004611	**
lan_1Korean	-4.996e+02	3.322e+02	-1.504	0.132883	
lan_1Mandarin	-2.962e+02	3.974e+02	-0.745	0.456319	
lan_1Spanish	-8.206e+02	3.500e+02	-2.345	0.019225	*
lan_1Thai	9.720e+02	3.941e+02	2.467	0.013794	*
lan_1Turkish	-1.653e+03	4.285e+02	-3.858	0.000121	***
rating	-3.631e+02	3.546e+01	-10.239	< 2e-16	***

startYear	1.059e+02	1.179e+01	8.980	< 2e-16	***
endYear	-1.060e+02	1.388e+01	-7.631	5.08e-14	***
numVotes	-9.762e-04	4.144e-04	-2.356	0.018656	*

The method that uses in this regression is the forward method and the result comes out to be a certificate, language, rating, start year, end year and the number of votes are the prediction variables to predict the ranking which is the response variable. Since the certificate is a categorical variable so the level of the column will be taken into the calculation; 13, 16, 18 and other certifications have almost zero p-values, then the null hypothesis can be rejected with strong evidence. As well as language is the categorical variable so the level is separate; English and Turkish languages can be rejected the null hypothesis with a 0% significant level. While the Japanese can reject the null hypothesis with a 0.1% level. Thai, Spanish, French and etc can be rejected with a 1% level. Start year and end year together with a rating can be rejected the null hypothesis with almost a 0% level which means strong evidence. A number of votes can be rejected with some evidence; a 1% level.

The response variable would be affected by predicted variables since the response variable is the sum-up of predicted variables and so on; an increase of 1 unit of the number of votes would affect the response variables by -9.762e-04 and the start year would affect by 1.059e+02.



The normal QQ plot describes the trend by using the theoretical distribution; the tail of the graph seems to have more spread out than supposed at those quantiles. In the middle of the graph might be able to conclude that the data have normal behaviour. Even so, the normal QQ plot might provide incomplete information; the residual plot therefore might be added to show the outliers which have approximate three values.

Conclusion:

The model indicates the prediction variables which are the most appropriate terms to previse the response variables. Although the ranking on IMDB may depend on more material, the data provide as much probability as possible and the result was satisfactory. The regression model provides each categories variables in level so the interpretation might be more clear and make the information more useful.

Appendix:

```
install.packages("RColorBrewer")

library(RColorBrewer)


netflix = read.csv("netflix_list.csv", header = TRUE)


netflix$popular_rank = as.numeric(gsub(",", "", netflix$popular_rank))


netflix = na.omit(netflix)



netflix_con = netflix[c(3,5,6,14,15)]

summary(netflix_con)


netflix_cat = netflix[c(4,9,11)]

summary(netflix_cat)


#certificate

netflix_cat$certificate_1 = ifelse(netflix_cat$certificate == "12" , "others",
netflix_cat$certificate)

netflix_cat$certificate_1 = ifelse(netflix_cat$certificate == "12+" , "others",
netflix_cat$certificate_1)
```

```
netflix_cat$certificate_1 = ifelse(netflix_cat$certificate == "13+" , "others",
netflix_cat$certificate_1)

netflix_cat$certificate_1 = ifelse(netflix_cat$certificate == "15" , "others",
netflix_cat$certificate_1)

netflix_cat$certificate_1 = ifelse(netflix_cat$certificate == "15+" , "others",
netflix_cat$certificate_1)

netflix_cat$certificate_1 = ifelse(netflix_cat$certificate == "18+" , "others",
netflix_cat$certificate_1)

netflix_cat$certificate_1 = ifelse(netflix_cat$certificate == "7+" , "others",
netflix_cat$certificate_1)

netflix_cat$certificate_1 = ifelse(netflix_cat$certificate == "A" , "others",
netflix_cat$certificate_1)

netflix_cat$certificate_1 = ifelse(netflix_cat$certificate == "All" , "others",
netflix_cat$certificate_1)

netflix_cat$certificate_1 = ifelse(netflix_cat$certificate == "Not Rated" , "others",
netflix_cat$certificate_1)

netflix_cat$certificate_1 = ifelse(netflix_cat$certificate == "PG" , "others",
netflix_cat$certificate_1)

netflix_cat$certificate_1 = ifelse(netflix_cat$certificate == "U" , "others",
netflix_cat$certificate_1)

netflix_cat$certificate_1 = ifelse(netflix_cat$certificate == "UA" , "others",
netflix_cat$certificate_1)

netflix_cat$certificate_1 = ifelse(netflix_cat$certificate == "16+" , "others",
netflix_cat$certificate_1)

netflix_cat$certificate_1 = ifelse(netflix_cat$certificate == "7" , "others",
netflix_cat$certificate_1)


#lan
```

```
netflix_cat$lan_1 = ifelse(netflix_cat$language == "-" , "etc", netflix_cat$language)
netflix_cat$lan_1 = ifelse(netflix_cat$language == "Arabic" , "etc", netflix_cat$lan_1)
netflix_cat$lan_1 = ifelse(netflix_cat$language == "Cantonese" , "etc", netflix_cat$lan_1)
netflix_cat$lan_1 = ifelse(netflix_cat$language == "Catalan" , "etc", netflix_cat$lan_1)
netflix_cat$lan_1 = ifelse(netflix_cat$language == "Danish" , "etc", netflix_cat$lan_1)
netflix_cat$lan_1 = ifelse(netflix_cat$language == "Dutch" , "etc", netflix_cat$lan_1)
netflix_cat$lan_1 = ifelse(netflix_cat$language == "Filipino" , "etc", netflix_cat$lan_1)
netflix_cat$lan_1 = ifelse(netflix_cat$language == "Finnish" , "etc", netflix_cat$lan_1)
netflix_cat$lan_1 = ifelse(netflix_cat$language == "Galician" , "etc", netflix_cat$lan_1)
netflix_cat$lan_1 = ifelse(netflix_cat$language == "German" , "etc", netflix_cat$lan_1)
netflix_cat$lan_1 = ifelse(netflix_cat$language == "Hebrew" , "etc", netflix_cat$lan_1)
netflix_cat$lan_1 = ifelse(netflix_cat$language == "Hindi" , "etc", netflix_cat$lan_1)
netflix_cat$lan_1 = ifelse(netflix_cat$language == "Icelandic" , "etc", netflix_cat$lan_1)
netflix_cat$lan_1 = ifelse(netflix_cat$language == "Indonesian" , "etc", netflix_cat$lan_1)
netflix_cat$lan_1 = ifelse(netflix_cat$language == "Italian" , "etc", netflix_cat$lan_1)
netflix_cat$lan_1 = ifelse(netflix_cat$language == "Latin" , "etc", netflix_cat$lan_1)
netflix_cat$lan_1 = ifelse(netflix_cat$language == "Luxembourgish" , "etc", netflix_cat$lan_1)
netflix_cat$lan_1 = ifelse(netflix_cat$language == "Min Nan" , "etc", netflix_cat$lan_1)
netflix_cat$lan_1 = ifelse(netflix_cat$language == "None" , "etc", netflix_cat$lan_1)
netflix_cat$lan_1 = ifelse(netflix_cat$language == "Norwegian" , "etc", netflix_cat$lan_1)
netflix_cat$lan_1 = ifelse(netflix_cat$language == "Polish" , "etc", netflix_cat$lan_1)
netflix_cat$lan_1 = ifelse(netflix_cat$language == "Portuguese" , "etc", netflix_cat$lan_1)
```

```
netflix_cat$lan_1 = ifelse(netflix_cat$language == "Russian" , "etc", netflix_cat$lan_1)

netflix_cat$lan_1 = ifelse(netflix_cat$language == "Swedish" , "etc", netflix_cat$lan_1)

netflix_cat$lan_1 = ifelse(netflix_cat$language == "Tagalog" , "etc", netflix_cat$lan_1)


cer_table = table(netflix_cat$certificate_1)

color <- brewer.pal(length(cer_table), "Paired")

pie(cer_table, labels = c("607", "13 = 74", "16 = 124", "18 = 177", "etc = 120"), col = color,
main = "Certificate", cex = 0.7)

legend("right",c("NA", "13", "16", "18", "etc"), cex = 0.5, fill = color)


types_table = table(netflix_cat$type)

color4 <- c("#E9896A", "#387C6D")

pie(types_table, labels = types_table, col = color4, main = "Type", cex = 0.7)

legend("right",c("tvMiniSeries", "tvSeries"), cex = 0.5, fill = color4)


lan_table = table(netflix_cat$lan_1)

color5 <- brewer.pal(length(lan_table), "Paired")

pie(lan_table, labels = lan_table, col = color5, main = "languages", cex = 0.7)

legend("right",c("Chinese", "English", "etc", "French", "Japanese", "Korean", "Mandarin",
"Spanish", "Thai",

          "Turkish"), cex = 0.5, fill = color5)


str(netflix_con)

summary(netflix_con)
```

```
par(mfrow=c(1,3))

boxplot(netflix_con$popular_rank, main="Popular rank")

boxplot(netflix_con$rating, main="Average rating")

boxplot(netflix_con$numVotes, main="Number of vote")

par(mfrow=c(1,2))

boxplot(netflix_con$startYear, main="Start Year")

boxplot(netflix_con$endYear, main="End Year")


#-------------------------------------------------

str(netflix$language)

str((netflix_cat$lan_1))


lang = tapply(netflix_cat$lan_1, netflix_cat$lan_1, table)

lang


plot(netflix$rating,netflix$numVotes,

    xlab = "score in IMDB",

    ylab = "number of votes in Netflix",

    main = "comparison between Netflix and IMDB",

    col = color5[factor(netflix_cat$lan_1, levels = c("Chinese", "English", "etc", "French",
"Japanese", "Korean", "Mandarin", "Spanish", "Thai", "Turkish"))],
```

```
    pch = 20)

legend("left",c("Chinese", "English", "etc", "French", "Japanese", "Korean", "Mandarin",
"Spanish", "Thai",

            "Turkish"), cex = 0.5, fill = color5)



#-------------------------------------------------



netflix$popular_rank_01 = ifelse(netflix$popular_rank >= mean(netflix$popular_rank) , "low",
"high")



dt = table(netflix$popular_rank_01, netflix_cat$lan_1)

addmargins(dt)

round(prop.table(dt,2)*100,digits=2)



chisq.test(netflix$popular_rank_01, netflix_cat$lan_1, simulate.p.value=TRUE)

fisher.test(netflix$popular_rank_01, netflix_cat$lan_1, simulate.p.value=TRUE)



#-------------------------------------------------



attach(netflix_use)



str(A)
```

```
netflix_use = data.frame(netflix_cat,netflix_con)


A = netflix_use[c(-1,-3,-6)]

model1 = lm(netflix$popular_rank~1, data=A)

step1<-step(model1, scope=~ type + certificate_1 + startYear + endYear + lan_1 + rating +
numVotes,

        method="forward")

summary(step1)


par(mfrow = c(1, 2))

plot(step1)
```