R－期末專題報告

透過 IMDb 5000+ 資料分析電影毛利與其自變數之關係

組員

資管四郭泰竹

資管四黃俊翔

資管四莊育新

企管四林秀憶

目錄

一、摘要

本研究主要探討的問題來自於 Kaggle 網站上的 IMDb 5000+的資料，運用該資料做資料分析，探討哪些變數對於電影賣座與否以及毛利多寡是有影響的，進而去預測電影的毛利金額。

首先，本次研究挑選了國家、語言、分級、電影種類等類別型變數作為基本模型的自變數，確認變數以及模型均為顯著後得出模型 fit1，再加入 facebook 讚數與預算的資料，放進模型做一次 overall 的 F 檢定，看出導演讚數與毛利是正相關，不過跟演員讚數相關的變數共線性>10，因此決定刪除部分變數後建立模型 fit2。接著，為了解當電影中的主角一人獨挑大樑時，對毛利影響，本次研究中建立了新變數，增強模型對於此一狀況的解釋性，並建立模型 fit3。加入新的變數之後，再做一次 overall 的 F 檢定，發現其 F 檢定的模型都是顯著的，也沒有共線性的問題，因此本研究之模型抵定是 fit3。

再來，我們利用前面建立之模型 fit3 來預測未來電影毛利，不過在預測之前，先利用 Rstudent 函數去觀察模型配適值與實際值的殘差圖，發現其中有離群值的存在後，於是我們回到資料中找尋並連續刪除幾筆資料，得到較好的模型以及 Rstudent 殘差圖。最後，將原有的 IMDB 之資料放進前面所得到的最終的模型 fit3，進而求得預測值。

二、研究動機與目的

隨著網路的發達以及電影的普及度日趨上升，許多觀眾在看完電影之後都會在網路上分享自己對於該電影的評價與喜好程度，其中一些熱門的電影評論平台，包括 Rotten Tomato，IMDb，以及貫小米看電影，都提供其他未觀賞過電影的民眾參考，作為他們判斷是否值得去戲院觀賞該電影的依據。

雖然如此，在未觀賞過電影之前，為了得知一部電影真實的好壞，民眾必須花時間去瀏覽各式各樣的網站搜集影評們分享的資訊，而這些資訊也並非完全正確。同樣地，在一部新電影正式上映之前，要如何得知它的好壞及預測他的賣座程度？

為了解決上述問題，本研究從 Kaggle 網站上搜集 IMDb 5000+電影的資料，將這些資料的變數做迴歸分析，可以得出哪些變數對於影片評論會有較大的影響，進而去調整每個變數在資料分析中的權重，透過分析資料的結果，我們希望可以得知什麼樣的組合會使電影得到作多收益或是正向的評比，如此一來，當有新電影上映時，可以與此組合做比對，得到一個預測結果，讓觀眾作為參考，因此觀眾不需要花費太多時間去尋找影評以及評分資訊，只需要舒適

地享受影片，也能避免發生進了電影院卻看到不好看的電影的情況。

三、建模過程

(一)建立基本模型

先採用 5000 筆資料內，刪去空值以及 null 值(0)之後，剩餘的類別型資料，建立第一個基本模型如下:

基本模型

$$\hat{Y} = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + b_4x_4 + b_5x_5 + b_6x_6 + b_7x_7$$

其中變數分別為，x_1＝語言，x_2＝國家，x_3＝種類（屬於動作片），x_4＝內容評等（NC-17），x_5＝內容評等（PG），x_6 ＝內容評等（PG-13），x_7 ＝內容評等（R），$\hat{Y}$為電影毛利(gross)。

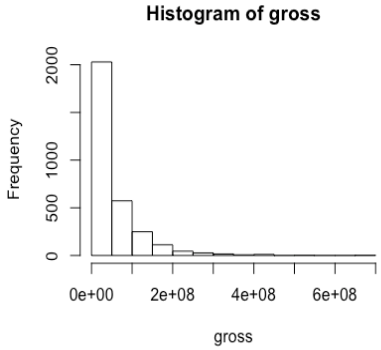
程式碼

```
ml=lm(gross~language+country+genres+as.factor(content_rating))
```

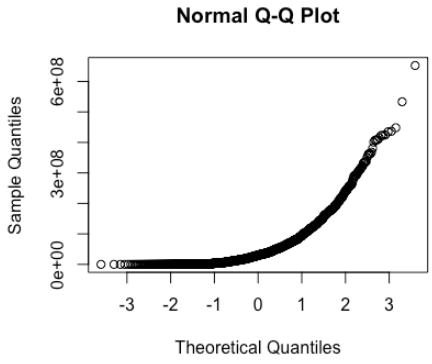
建立模型後，觀察反應變數的直方圖及 Q-Q 圖，可以得知此變數不符合迴歸分析之常態假設，直方圖為右尾，QQ 圖呈指數型態分布，因此以 boxcox 函數去檢視原始模型，建議配適 0.25 次方。(見下圖一、圖二、圖三)

程式碼

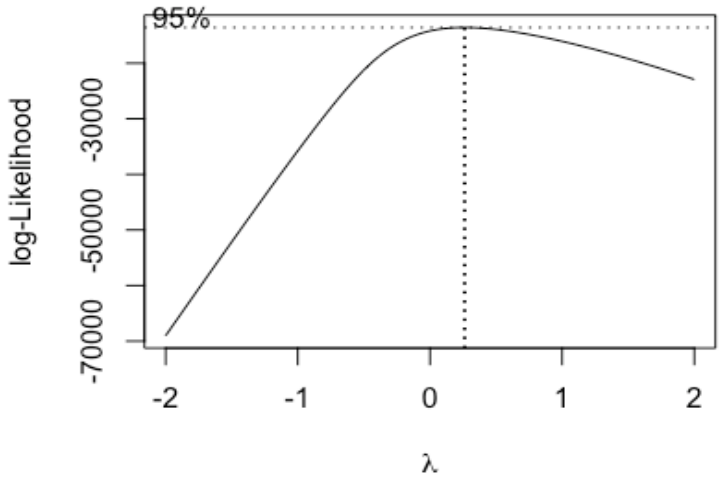
```
qqnorm(gross)
hist(gross)
boxcox(ml)
```



<圖一>



<圖二>



<圖三>

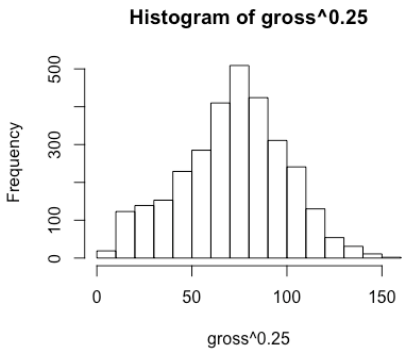
因此配置 0.25 次方給 gross 後，得出模型 fit1，並觀察反應變數的直方圖及 Q-Q 圖，得知此變數符合迴歸分析之常態假設。

模型 fit1

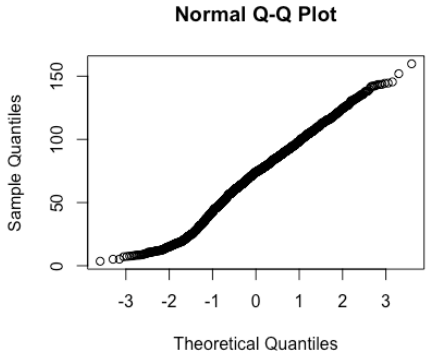
$$\widehat{Y^{0.25}} = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + b_4x_4 + b_5x_5 + b_6x_6 + b_7x_7$$

程式碼

```
fit1=lm(gross^0.25~language+country+genres+as.factor(content_rating))
    qqnorm(gross^0.25)
    hist(gross^0.25)
```



<圖四>



<圖五>

確定模型後，對模型做一次 Overall 的 F 檢定，F 檢定的模型都是顯著的，且無共線性問題。此外，從報表中可看出語言為英語、國家為美國及屬於動作片時對毛利是正相關，而內容評等會因為等級變高可觀賞人數減少，對毛利為負相關。(見下圖六、圖七)

程式碼

```
summary(fit1)
vif(fit1)
```

```
Coefficients:
                            Estimate Std. Error t value Pr(>|t|)
(Intercept)                   57.715      3.577  16.133  < 2e-16 ***
language                      19.874      2.472   8.041 1.26e-15 ***
country                       13.627      1.176  11.583  < 2e-16 ***
genres                        12.117      1.004  12.069  < 2e-16 ***
as.factor(content_rating)NC-17  -48.841     12.448  -3.924 8.92e-05 ***
as.factor(content_rating)PG     -8.941      3.027  -2.954  0.00317 **
as.factor(content_rating)PG-13  -14.416      2.926  -4.927 8.80e-07 ***
as.factor(content_rating)R      -28.026      2.900  -9.663  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 24.25 on 3063 degrees of freedom
Multiple R-squared:  0.2252,    Adjusted R-squared:  0.2234
F-statistic: 127.2 on 7 and 3063 DF,  p-value: < 2.2e-16
```

<圖六>

```
                            GVIF Df GVIF^(1/(2*Df))
language                1.140944  1        1.068150
country                 1.141122  1        1.068233
genres                  1.024013  1        1.011935
as.factor(content_rating) 1.034984  4        1.004308
```

<圖七>

(二)加入 FB 讚數重新建模

完成了基本模型的建置後，我們加入 FB 讚數的自變數，其中包含導演以及三位演員和總演員的讚數，建立一個新的模型 fit1。

模型 fit1

$$\widehat{Y^{0.25}} = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + b_4x_4 + b_5x_5 + b_6x_6 + b_7x_7 + b_8x_8 + b_9x_9 + b_{10}x_{10} + b_{11}x_{11} + b_{12}x_{12} + b_{13}x_{13}$$

程式碼:

```
fit1=lm(gross^0.25~language+country+genres+as.factor(content_rating)+director_likes+actor1_likes+actor2_likes+actor3_likes+allcast_likes+budget)
```

接著對新建立的模型做一次 overall 的 F 檢定，可以從報表中的 F 檢定得知模型是顯著的，變數也都呈顯著，且可看出導演讚數與毛利是正相關，但跟演員讚數相關的變數共線性>10，因此決定刪除部分變數。(見下圖八、圖九)

程式碼

```
summary(fit1)
vif(fit1)
```

```
Coefficients:
                                Estimate Std. Error t value Pr(>|t|)
(Intercept)                    5.711e+01  3.450e+00  16.555  < 2e-16 ***
language                       1.851e+01  2.398e+00   7.721 1.56e-14 ***
country                        1.132e+01  1.136e+00   9.964  < 2e-16 ***
genres                         1.118e+01  9.654e-01  11.576  < 2e-16 ***
as.factor(content_rating)NC-17 -4.743e+01  1.192e+01  -3.980 7.05e-05 ***
as.factor(content_rating)PG    -1.059e+01  2.901e+00  -3.652 0.000265 ***
as.factor(content_rating)PG-13 -1.677e+01  2.807e+00  -5.975 2.56e-09 ***
as.factor(content_rating)R     -2.943e+01  2.783e+00 -10.577  < 2e-16 ***
director_likes                  9.510e-04  1.273e-04   7.469 1.05e-13 ***
actor1_likes                   -2.461e-03  4.095e-04  -6.009 2.09e-09 ***
actor2_likes                   -2.248e-03  4.282e-04  -5.250 1.62e-07 ***
actor3_likes                   -2.401e-03  6.619e-04  -3.628 0.000290 ***
allcast_likes                   2.610e-03  4.057e-04   6.434 1.44e-10 ***
budget                          6.736e-09  1.823e-09   3.695 0.000224 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 23.21 on 3057 degrees of freedom
Multiple R-squared:  0.2916,    Adjusted R-squared:  0.2886
F-statistic: 96.79 on 13 and 3057 DF,  p-value: < 2.2e-16
```

<圖八>

```
                          GVIF Df GVIF^(1/(2*Df))
language              1.171861  1        1.082525
country               1.161057  1        1.077524
genres                1.029528  1        1.014657
as.factor(content_rating) 1.050990  4        1.006236
director_likes        1.040492  1        1.020045
actor1_likes        134.589014  1       11.601251
actor2_likes         21.937591  1        4.683758
actor3_likes          7.769266  1        2.787340
allcast_likes       242.372442  1       15.568315
budget                1.027673  1        1.013742
```

<圖九>

因為在此研究中，變數 allcast_likes 相較於其他兩個變數，考慮到的範圍較廣，因此決定將 actor1_likes 及 actor2_likes 兩變數刪除，並建立模型 fit2。

模型 fit2

$$\widehat{Y^{0.25}} = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + b_4x_4 + b_5x_5 + b_6x_6 + b_7x_7 + b_8x_8 + b_9x_9 + b_{10}x_{10} + b_{11}x_{11}$$

程式碼

```
fit2=lm(gross^0.25~language+country+genres+as.factor(content_rating)+
director_likes+actor3_likes+allcast_likes+budget)
```

同樣地，我們再對 fit2 模型做一次檢定，發現模型以及其變數都是顯著的，且無共線性問題，但僅探討演員三的人氣對模型的解釋效益不大。(見下圖十、圖十一)

程式碼

```
summary(fit2)
vif(fit2)
```

```
Coefficients:
                                 Estimate Std. Error t value Pr(>|t|)
(Intercept)                     5.752e+01  3.468e+00  16.585  < 2e-16 ***
language                        1.917e+01  2.409e+00   7.959 2.41e-15 ***
country                         1.187e+01  1.139e+00  10.421  < 2e-16 ***
genres                          1.142e+01  9.700e-01  11.770  < 2e-16 ***
as.factor(content_rating)NC-17 -4.851e+01  1.198e+01  -4.048 5.30e-05 ***
as.factor(content_rating)PG    -1.091e+01  2.917e+00  -3.739 0.000188 ***
as.factor(content_rating)PG-13 -1.702e+01  2.823e+00  -6.028 1.85e-09 ***
as.factor(content_rating)R     -2.998e+01  2.797e+00 -10.716  < 2e-16 ***
director_likes                  9.434e-04  1.281e-04   7.367 2.24e-13 ***
actor3_likes                    1.263e-03  2.823e-04   4.472 8.02e-06 ***
allcast_likes                   2.182e-04  3.126e-05   6.982 3.56e-12 ***
budget                          6.882e-09  1.834e-09   3.753 0.000178 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 23.34 on 3059 degrees of freedom
Multiple R-squared:  0.2828,    Adjusted R-squared:  0.2803
F-statistic: 109.7 on 11 and 3059 DF,  p-value: < 2.2e-16
```

<圖十>

```
> vif(fit2)
                              GVIF Df GVIF^(1/(2*Df))
language                  1.168935  1        1.081173
country                   1.153456  1        1.073990
genres                    1.027277  1        1.013547
as.factor(content_rating) 1.045959  4        1.005633
director_likes            1.040391  1        1.019996
actor3_likes              1.397572  1        1.182190
allcast_likes             1.422254  1        1.192583
budget                    1.027489  1        1.013651
```

<圖十一>

為了解當電影中的主角一人獨挑大樑時，對毛利影響，本次研究中建立了新變數 actor1_ratio，actor1_ratio（=actor1_likes/allcast_likes）以演員一讚數佔所有演員讚數的比例為變數，取代演員三讚數的變數，增強模型對於此一狀況的解釋性，並建立模型 fit3。

模型 fit3

$$\widehat{Y^{0.25}} = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + b_4x_4 + b_5x_5 + b_6x_6 + b_7x_7 + b_8x_8 + b_9x_9 + b_{10}x_{10} + b_{11}x_{11}$$

程式碼

```
actor1_ratio=actor1_likes/allcast_likes #建立新變數
fit3=lm(gross^0.25~language+country+genres+as.factor(content_rating)+
director_likes+allcast_likes+actor1_ratio+budget)
```

加入新的變數之後，再做一次 overall 的 F 檢定，發現其 F 檢定的模型都是顯著的，也沒有共線性的問題，因此本研究之模型暫時抵定是 fit3。(見下圖十二、十三)

程式碼

```
summary(fit3)
vif(fit3)
```

```
Coefficients:
                                Estimate Std. Error t value Pr(>|t|)
(Intercept)                    5.832e+01  3.628e+00  16.073  < 2e-16 ***
language                       1.912e+01  2.420e+00   7.903 3.77e-15 ***
country                        1.185e+01  1.145e+00  10.352  < 2e-16 ***
genres                         1.135e+01  9.732e-01  11.662  < 2e-16 ***
as.factor(content_rating)NC-17 -4.861e+01  1.202e+01  -4.044 5.39e-05 ***
as.factor(content_rating)PG    -1.085e+01  2.927e+00  -3.708 0.000212 ***
as.factor(content_rating)PG-13 -1.683e+01  2.832e+00  -5.942 3.14e-09 ***
as.factor(content_rating)R     -3.005e+01  2.808e+00 -10.704  < 2e-16 ***
director_likes                 9.900e-04  1.282e-04   7.719 1.57e-14 ***
allcast_likes                  2.967e-04  2.892e-05  10.259  < 2e-16 ***
actor1_ratio                  -1.391e+00  1.913e+00  -0.727 0.467081
budget                         7.002e-09  1.839e-09   3.807 0.000144 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 23.42 on 3059 degrees of freedom
Multiple R-squared:  0.2783,    Adjusted R-squared:  0.2757
F-statistic: 107.2 on 11 and 3059 DF,  p-value: < 2.2e-16
```

<圖十二>

```
> vif(fit3)
                              GVIF Df GVIF^(1/(2*Df))
language                  1.172203  1        1.082683
country                   1.158029  1        1.076118
genres                    1.027562  1        1.013688
as.factor(content_rating) 1.044049  4        1.005403
director_likes            1.036668  1        1.018169
allcast_likes             1.209771  1        1.099896
actor1_ratio              1.166641  1        1.080112
budget                    1.027269  1        1.013543
```

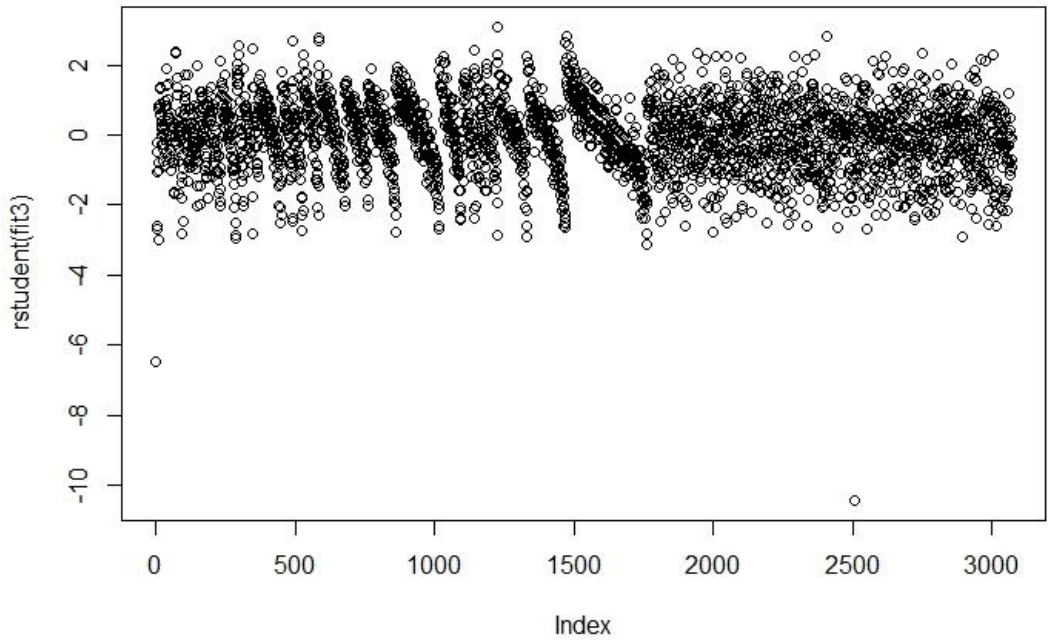
<圖十三>

(三)預測

完成以上模型的建置後，本部分將利用前面建立之模型 fit3 來預測未來電影毛利。在預測之前，我們利用 Rstudent 函數去觀察模型配適值與實際值的殘差圖，發現其中有離群值的存在。(如下圖十四)

程式碼:

```
plot(rstudent(fit3))
```

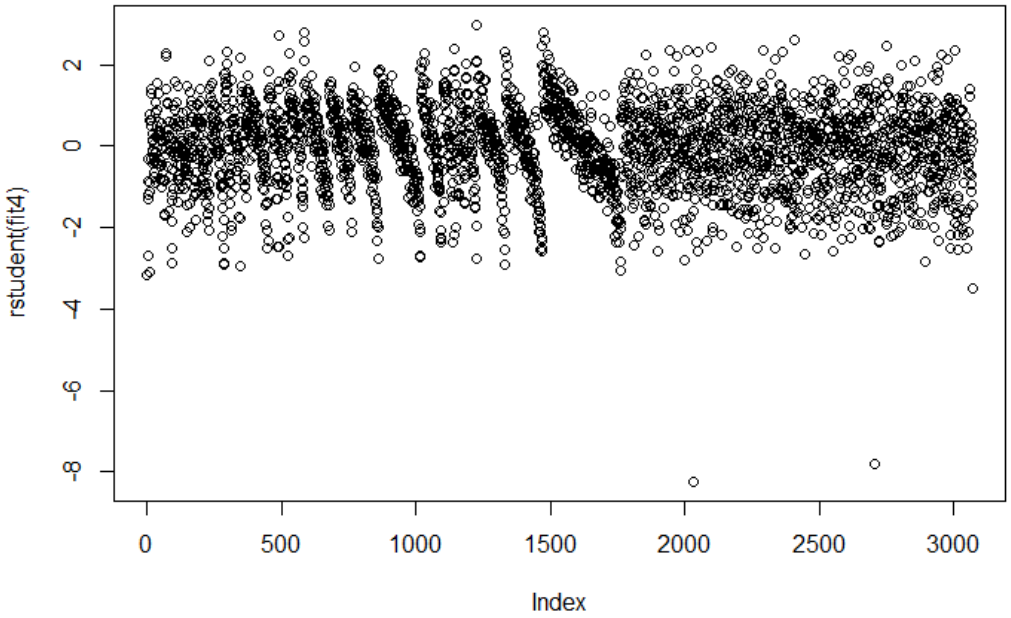


<圖十四>

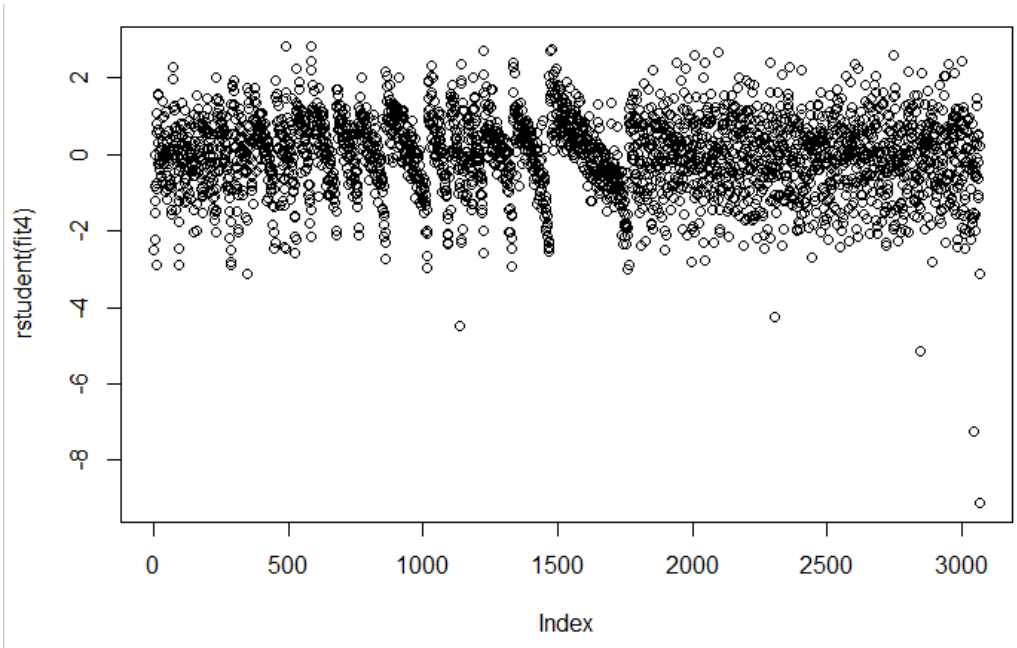
藉由觀察 Rstudent 圖，可以發現有一明顯的離群值，回到資料中找尋此筆資料，找到的資料如下圖所示，分別是第 1 筆資料以及第 2508 筆資料(見圖十五)，刪除資料後，再跑一次檢定，其中 Rstudent 圖又發現了離群值，連續刪除幾筆資料後，可以得到較好的模型以及 Rstudent 殘差圖。(見圖十六、十七、十八)

	director_likes	actor3_likes	actor1_likes	gross	genres	allcast_likes	language	country	content_rating	budget	actor2_likes
1	74	891	260000	96734	1	263584	1	1	PG-13	1.00e+06	984
2508	584	74	629	2201412	0	1173	0	0	R	12215500000	398

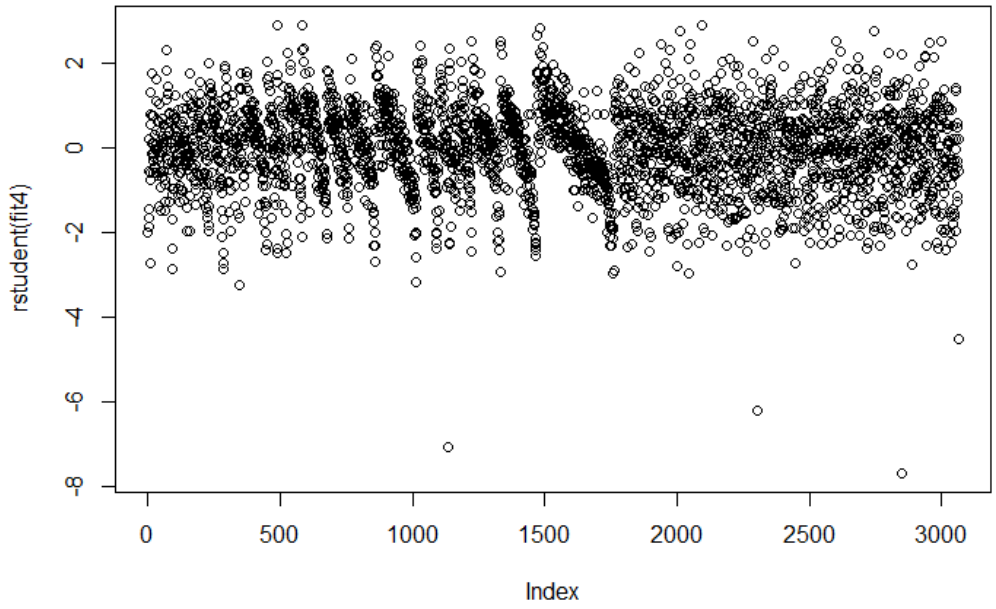
<圖十五>


rstudent(fit4)
2
0
-2
-4
-6
-8
0
500
1000
1500
2000
2500
3000
Index

〈圖十六〉


rstudent(fit4)
2
0
-2
-4
-6
-8
0
500
1000
1500
2000
2500
3000
Index

〈圖十七〉



<圖十八>

刪去以上幾筆資料在做一次 overall 的 F 檢定，可以得到以下的報表，可以發現 R 平方明顯的升高了(0.2783->0.4322)，也就是 fit3 模型中的自變數x_i對於模型的解釋力變高了（見下圖十九、二十），於是我們針對此模型在做一次 Rstudent 的殘差檢定，可以確定 Rstudent 的分布為常態。(見下圖二十一)

程式碼

```
summary(fit3)
vif(fit3)
```

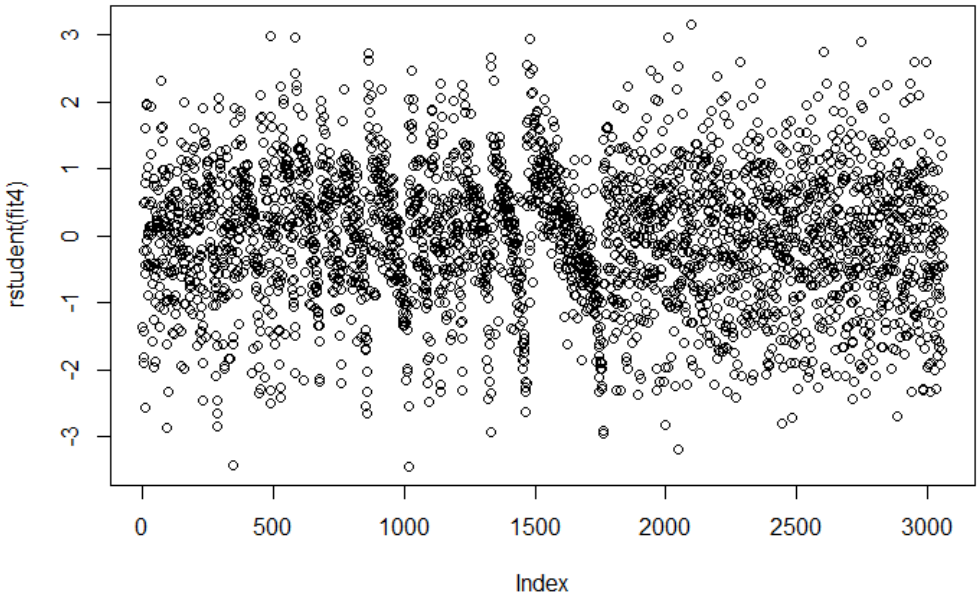
```
Coefficients:
                                Estimate Std. Error t value Pr(>|t|)
(Intercept)                    4.867e+01  3.274e+00  14.868  < 2e-16 ***
language                       1.624e+01  2.197e+00   7.390 1.88e-13 ***
country                        9.866e+00  1.016e+00   9.711  < 2e-16 ***
genres                         1.729e+00  9.319e-01   1.855 0.063642 .
as.factor(content_rating)NC-17 -3.621e+01  1.065e+01  -3.399 0.000685 ***
as.factor(content_rating)PG    -7.244e+00  2.596e+00  -2.791 0.005290 **
as.factor(content_rating)PG-13 -1.079e+01  2.518e+00  -4.287 1.87e-05 ***
as.factor(content_rating)R     -1.792e+01  2.524e+00  -7.097 1.58e-12 ***
director_likes                  6.495e-04  1.143e-04   5.684 1.44e-08 ***
allcast_likes                   1.614e-04  2.768e-05   5.832 6.07e-09 ***
actor1_ratio                   -4.029e+00  1.703e+00  -2.366 0.018059 *
budget                          3.156e-07  1.107e-08  28.518  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 20.73 on 3049 degrees of freedom
Multiple R-squared:  0.4322,    Adjusted R-squared:  0.4301
F-statistic:   211 on 11 and 3049 DF,  p-value: < 2.2e-16
```

<圖十九>

	GVIF	Df	GVIF^(1/(2*Df))
language	1.138921	1	1.067202
country	1.149340	1	1.072073
genres	1.194514	1	1.092938
as.factor(content_rating)	1.172277	4	1.020067
director_likes	1.048454	1	1.023940
allcast_likes	1.298439	1	1.139491
actor1_ratio	1.174956	1	1.083954
budget	1.441478	1	1.200616

<圖二十>



<圖二十一>

得到新的資料後，本研究將進入預測的階段，將原有的 IMDB 之資料放進前面所得到的最終的模型 fit3，進而求得預測值，將預測的結果與實際毛利相減，可以看出此預測的成效大部分集中在 0 附近，雖然也有非常多散布得非常離譜，但有可能是因為資料的來源不齊全導致無法準確預測毛利值，因此預測結果是可以接受的。(見圖二十二、圖二十三)

模型 fit3

$$\widehat{Y^{0.25}} = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + b_4x_4 + b_5x_5 + b_6x_6 + b_7x_7 + b_8x_8 + b_9x_9 + b_{10}x_{10} + b_{11}x_{11}$$

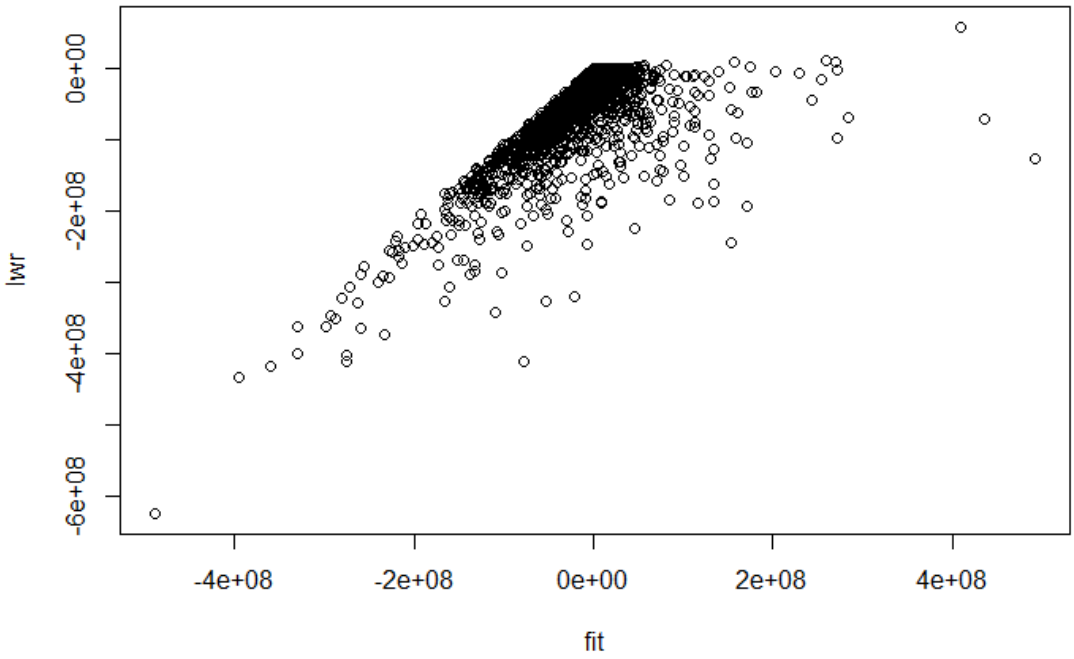
程式碼

```
pred=predict(lm(fit3),movie_metadata1,interval = "prediction")
console=pred^4
plot(console-gross)
```

預測結果(部分資料):

	fit	lwr	upr
1	183862169	2.810764e+07	656650059
2	39750363	2.124400e+06	211733252
3	73469643	6.948740e+06	320705707
4	353801525	8.540470e+07	1007574691
5	39883427	2.170516e+06	211222662
6	264324067	5.630701e+07	804009492
7	73789472	7.172516e+06	318723415
8	60815421	5.024284e+06	279274361
9	21359967	5.393634e+05	140464664
10	143779365	2.230942e+07	510025592
11	61292098	5.074729e+06	281238354
12	137378882	2.069968e+07	493871933
13	207836408	3.949779e+07	669588620
14	21067099	5.300123e+05	138665760
15	63620306	5.535120e+06	286622460
16	18183423	3.643473e+05	126405986
17	126348414	1.802994e+07	465421489
18	161358257	2.677166e+07	554903970
19	36284101	1.841872e+06	196417021
20	88217252	9.946988e+06	359210770
21	134662459	2.000797e+07	487178568
22	80980539	8.585615e+06	338041768
23	48625525	3.336587e+06	238504611
24	40530366	2.325843e+06	211023955
25	31989467	1.410584e+06	180745797
26	43873949	2.722788e+06	222635729
27	18257785	3.681128e+05	126740520
28	35583523	1.761147e+06	194141983
29	15211829	2.223208e+05	113381263
30	53405868	3.977239e+06	254579579
31	40927560	2.369185e+06	212490514
32	19851876	4.585045e+05	133437704
33	22503469	6.278970e+05	144295401
34	25676662	8.527983e+05	157239994
35	12449592	1.203348e+05	100704139

〈圖二十二〉



<圖二十三>

四、結論

本研究之最終模型為

$$\widehat{Y^{0.25}} = 48.67 + 16.24x_1 + 9.866x_2 + 1.729x_3 - 36.21x_4 - 7.244x_5 - 10.79b_6x_6 - 17.92x_7 + 0.0006495x_8 + 0.0001614x_9 - 4.029x_{10} + 0.0000003156x_{11}$$

解釋模型變數

x_1＝語言，x_2＝國家，x_3＝種類（屬於動作片），x_4＝內容評等（NC-17），x_5＝內容評等（PG），x_6 ＝內容評等（PG-13），x_7 ＝內容評等（R），x_8 ＝導演臉書讚數，x_9=全部卡司讚數 ，x_{10} ＝一號演員臉書讚數佔全部卡司讚數之比例，x_{11} ＝電影預算，$\widehat{Y}$為電影毛利(gross)。

藉由此模型，我們可以由 IMDb 得到的資料，在每一部電影上映前，去預測此部電影是否會賣座，並且將此資料提供給電影業者以及觀眾做參考，使電影業者在電影上映前能夠做出相應的對策，觀眾也能挑選自己想看的電影，避免發生進了電影院卻看到不好看的電影的情況。雖然此模型預測出來的毛利額跟實際上的毛利額有一定的誤差，但隨著資料來源齊全、模型修正，預測的成效可望更加成功。