DATA621-FinalProject-SmoothOperators

Rob Hodde, Matt Farris, Jeffrey Burmood, Bin Lin

5/26/2017

Introduction

Abstract

Movies: The quintessential form of storytelling that we as humans, have developed thus far. Movies have become the modern past-time for us, a way to escape the humdrum of everyday life into a fantasy world filled with drama, intrigue and delight. Movies have astounded audiences for the best part of a century, and with that, movie making has become a vast and lucrative industry. Studios, actors and actresses, directors, and production companies make up just a small part of world of film, and we hope by looking into some movie data we will be able to find some insights into that world. Finally, as avid fans and lovers of all films, we hope this report will both entertain you and reveal new insights into a fascinating world.

Problem Description

In this project, we explore, analyze and model a dataset containing information on approximately 5,000 movies. The dataset contains movie data extracted from the IMDB website and is available on Kaggle.com.

We develop predictive models for three questions:

1) Will the movie make money or lose money?
2) What is the anticipated gross margin (profit) for the movie?
3) Do any particular movie content keywords influence profitability?

Data Exploration

Data Exploration

The first part of our project consists of exploring our data source. As stated above, it came from Kaggle, a repository/social hub for data analysts like ourselves. Obviously the dataset isn't complete, since thousands of movies are released each year.

First we remove the data columns for the variables that we will not be using in the analysis. We focus on the following variables:

```
##  [1] "duration"                "director_facebook_likes"
##  [3] "actor_3_facebook_likes"  "actor_1_facebook_likes"
##  [5] "gross"                   "movie_title"
##  [7] "num_voted_users"         "cast_total_facebook_likes"
##  [9] "facenumber_in_poster"    "content_rating"
## [11] "budget"                  "title_year"
## [13] "actor_2_facebook_likes"  "imdb_score"
```

After exploring the data, we notice there is a scattering of NAs across the variables. Due to the relatively low number of total NAs, we remove all rows with NAs, leaving 3,828 rows of data.

Furthermore, we notice approximately 800 foreign films. Though we would love for these to be part of our dataset, the production budget and gross receipts variables for these films tend to differ dramatically. The budget is usually in the currency of the home country whereas the gross receipts tend to be in U.S. dollars. Because of the unwieldiness of adjusting for multiple currency exchange rates across many years on a case-by-case basis, we must remove this data. This leaves us with 3042 movies to analyze, which we feel is more than adequate for the project.

Next we explore the nature of the data for the variables to be used in the analysis.

VAR	TYPE
duration	integer
director_facebook_likes	integer
actor_3_facebook_likes	integer
actor_1_facebook_likes	integer
gross	integer
movie_title	character
num_voted_users	integer
cast_total_facebook_likes	integer
facenumber_in_poster	integer
content_rating	character
budget	double
title_year	integer
actor_2_facebook_likes	integer
imdb_score	double

duration	director_facebook_likes	actor_3_facebook_likes	actor_1_facebook_likes	gross
Min. : 37.0	Min. : 0.0	Min. : 0.0	Min. : 0.0	Min. : 703
1st Qu.: 95.0	1st Qu.: 11.0	1st Qu.: 233.0	1st Qu.: 811.2	1st Qu.: 11787482
Median :105.0	Median : 62.0	Median : 467.0	Median : 2000.0	Median : 34264376
Mean :109.5	Mean : 911.3	Mean : 836.2	Mean : 8241.5	Mean : 57651658
3rd Qu.:119.0	3rd Qu.: 235.0	3rd Qu.: 723.0	3rd Qu.: 13000.0	3rd Qu.: 75074326
Max. :330.0	Max. :23000.0	Max. :23000.0	Max. :640000.0	Max. :760505847

movie_title	num_voted_users	cast_total_facebook_likes	facenumber_in_poster	content_rating
Length:3042	Min. : 22	Min. : 0	Min. : 0.000	R :1333
Class :character	1st Qu.: 19117	1st Qu.: 2210	1st Qu.: 0.000	PG-13 :1110
Mode :character	Median : 54462	Median : 4517	Median : 1.000	PG : 472

movie_title	num_voted_users	cast_total_facebook_likes	facenumber_in_poster	content_rating
NA	Mean : 108285	Mean : 12340	Mean : 1.419	G : 70
NA	3rd Qu.: 132124	3rd Qu.: 16904	3rd Qu.: 2.000	Not Rated: 18
NA	Max. :1689764	Max. :656730	Max. :43.000	Unrated : 13
NA	NA	NA	NA	(Other) : 26

budget	title_year	actor_2_facebook_likes	imdb_score
Min. : 218	Min. :1929	Min. : 0.0	Min. :1.600
1st Qu.: 10725000	1st Qu.:1999	1st Qu.: 436.0	1st Qu.:5.800
Median : 25000000	Median :2004	Median : 729.5	Median :6.500
Mean : 40319361	Mean :2003	Mean : 2180.3	Mean :6.383
3rd Qu.: 55000000	3rd Qu.:2010	3rd Qu.: 1000.0	3rd Qu.:7.100
Max. :300000000	Max. :2016	Max. :137000.0	Max. :9.300

We investigate correlations. Below, we can see that none of the variables have significant correlation that we can perceive.

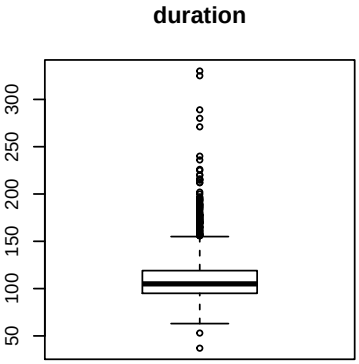
	duration	director_facebook_likes	actor_3_facebook_likes	actor_1_facebook_likes
duration	1.0000000	0.2104197	0.1448777	0.0912903
director_facebook_likes	0.2104197	1.0000000	0.1219467	0.0868426
actor_3_facebook_likes	0.1448777	0.1219467	1.0000000	0.2483043
actor_1_facebook_likes	0.0912903	0.0868426	0.2483043	1.0000000
num_voted_users	0.3705768	0.3190331	0.2818195	0.1741973
cast_total_facebook_likes	0.1349956	0.1172865	0.4830033	0.9459350
facenumber_in_poster	0.0065845	-0.0523321	0.1042739	0.0538466
budget	0.2988689	0.0942904	0.2747815	0.1551897
title_year	-0.1086958	-0.0580504	0.1277213	0.0914452
actor_2_facebook_likes	0.1504159	0.1192872	0.5521997	0.3798140
imdb_score	0.3819342	0.2225461	0.0882029	0.1178984

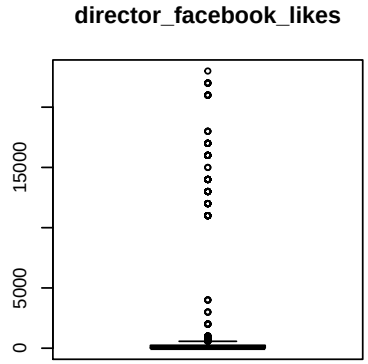
	num_voted_users	cast_total_facebook_likes	facenumber_in_poster	budget
duration	0.3705768	0.1349956	0.0065845	0.2988689
director_facebook_likes	0.3190331	0.1172865	-0.0523321	0.0942904
actor_3_facebook_likes	0.2818195	0.4830033	0.1042739	0.2747815
actor_1_facebook_likes	0.1741973	0.9459350	0.0538466	0.1551897
num_voted_users	1.0000000	0.2486828	-0.0441983	0.4054595

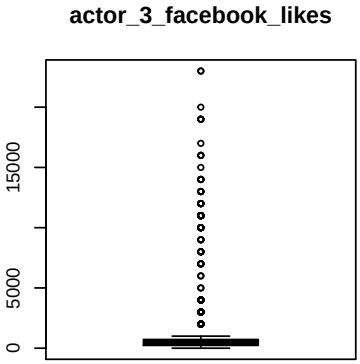
	num_voted_users	cast_total_facebook_likes	facenumber_in_poster	budget
cast_total_facebook_likes	0.2486828	1.0000000	0.0750811	0.2362870
facenumber_in_poster	-0.0441983	0.0750811	1.0000000	-0.0267742
budget	0.4054595	0.2362870	-0.0267742	1.0000000
title_year	0.0241674	0.1256809	0.0873375	0.2412454
actor_2_facebook_likes	0.2524944	0.6319688	0.0625703	0.2526741
imdb_score	0.5089320	0.1377072	-0.0694804	0.0713682

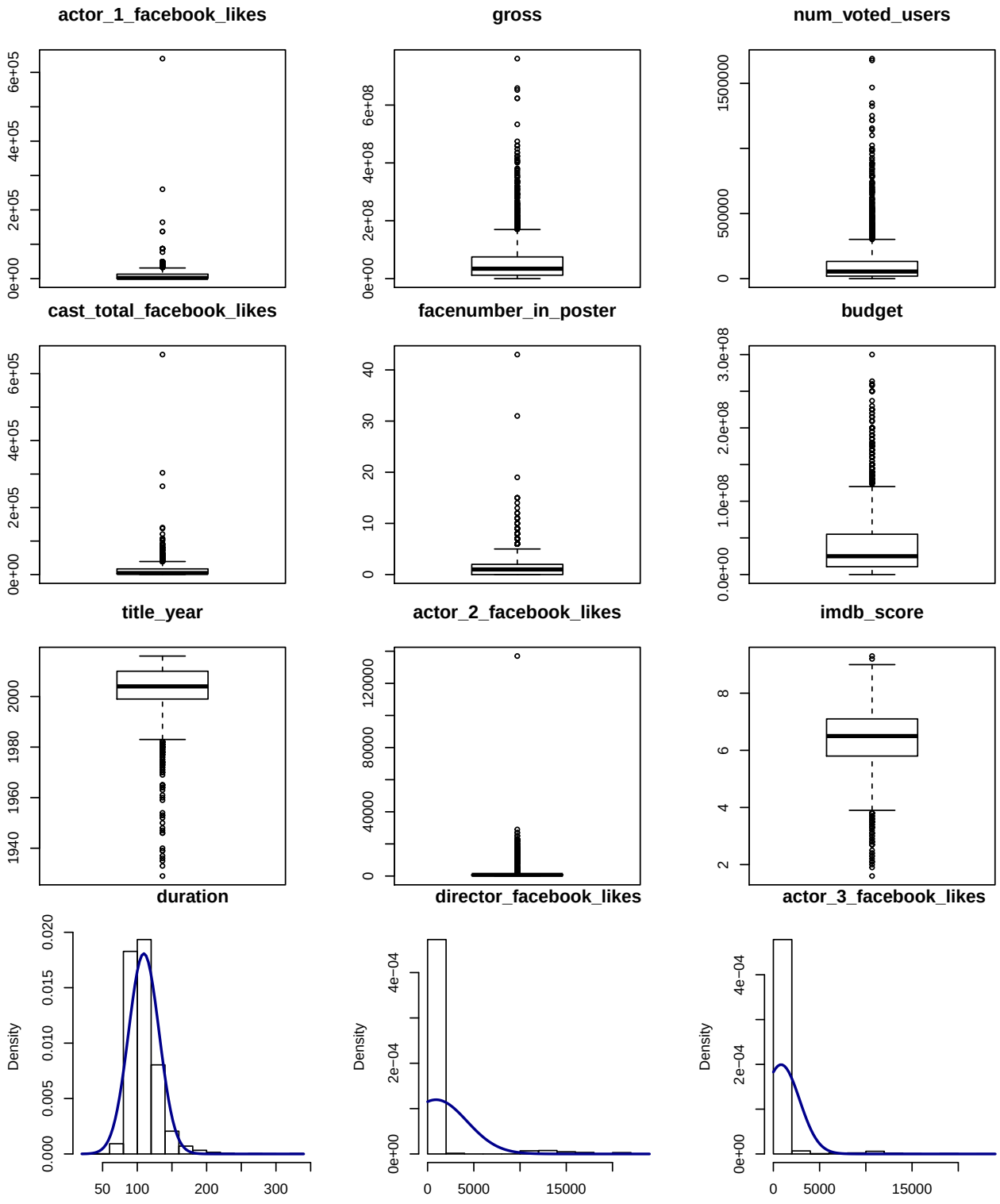
	title_year	actor_2_facebook_likes
duration	-0.1086958	0.1504159
director_facebook_likes	-0.0580504	0.1192872
actor_3_facebook_likes	0.1277213	0.5521997
actor_1_facebook_likes	0.0914452	0.3798140
num_voted_users	0.0241674	0.2524944
cast_total_facebook_likes	0.1256809	0.6319688
facenumber_in_poster	0.0873375	0.0625703
budget	0.2412454	0.2526741
title_year	1.0000000	0.1253783
actor_2_facebook_likes	0.1253783	1.0000000
imdb_score	-0.1504498	0.1274387

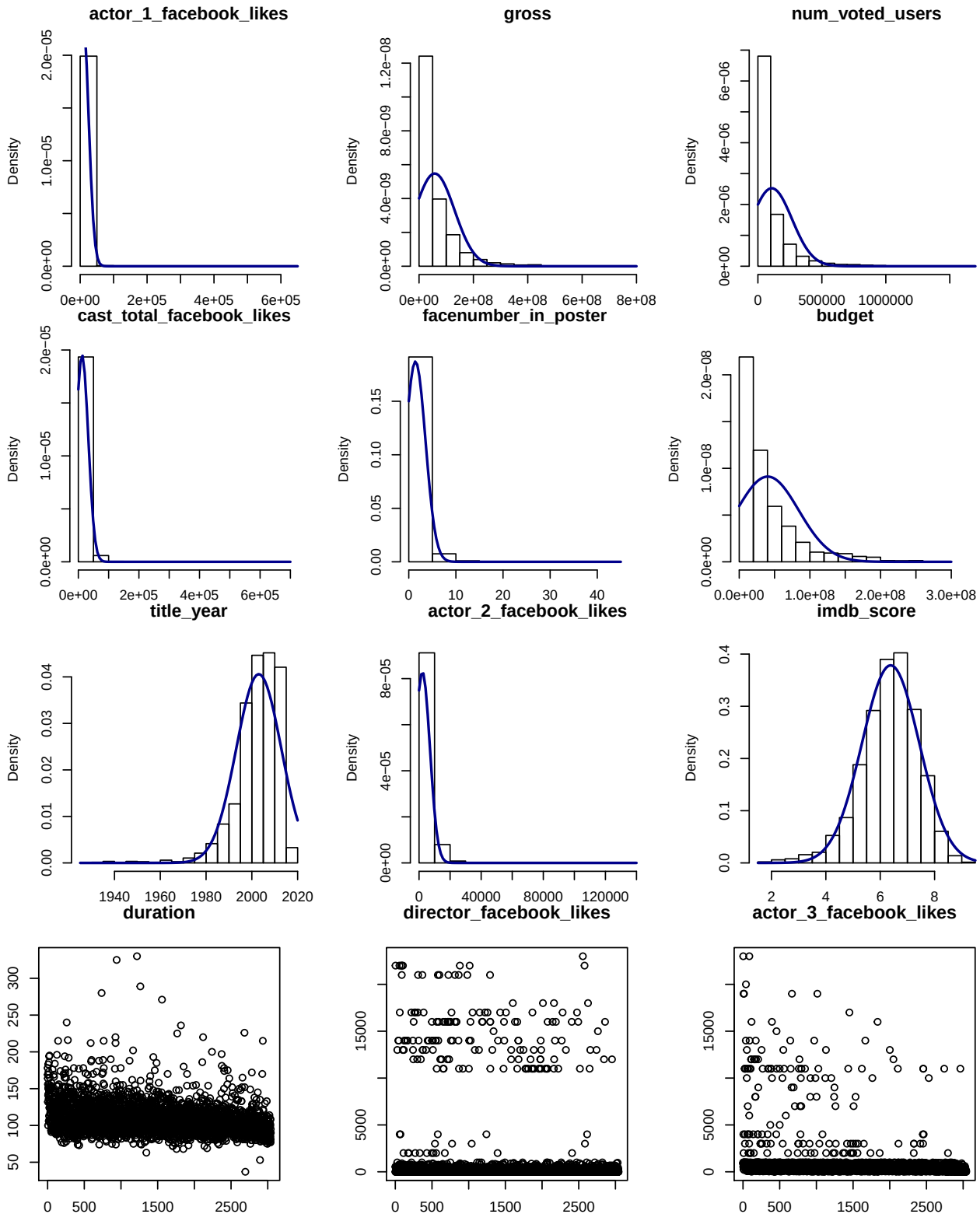
Lastly for exploration, we examine all variables using boxplots, histograms, and scatter plots.

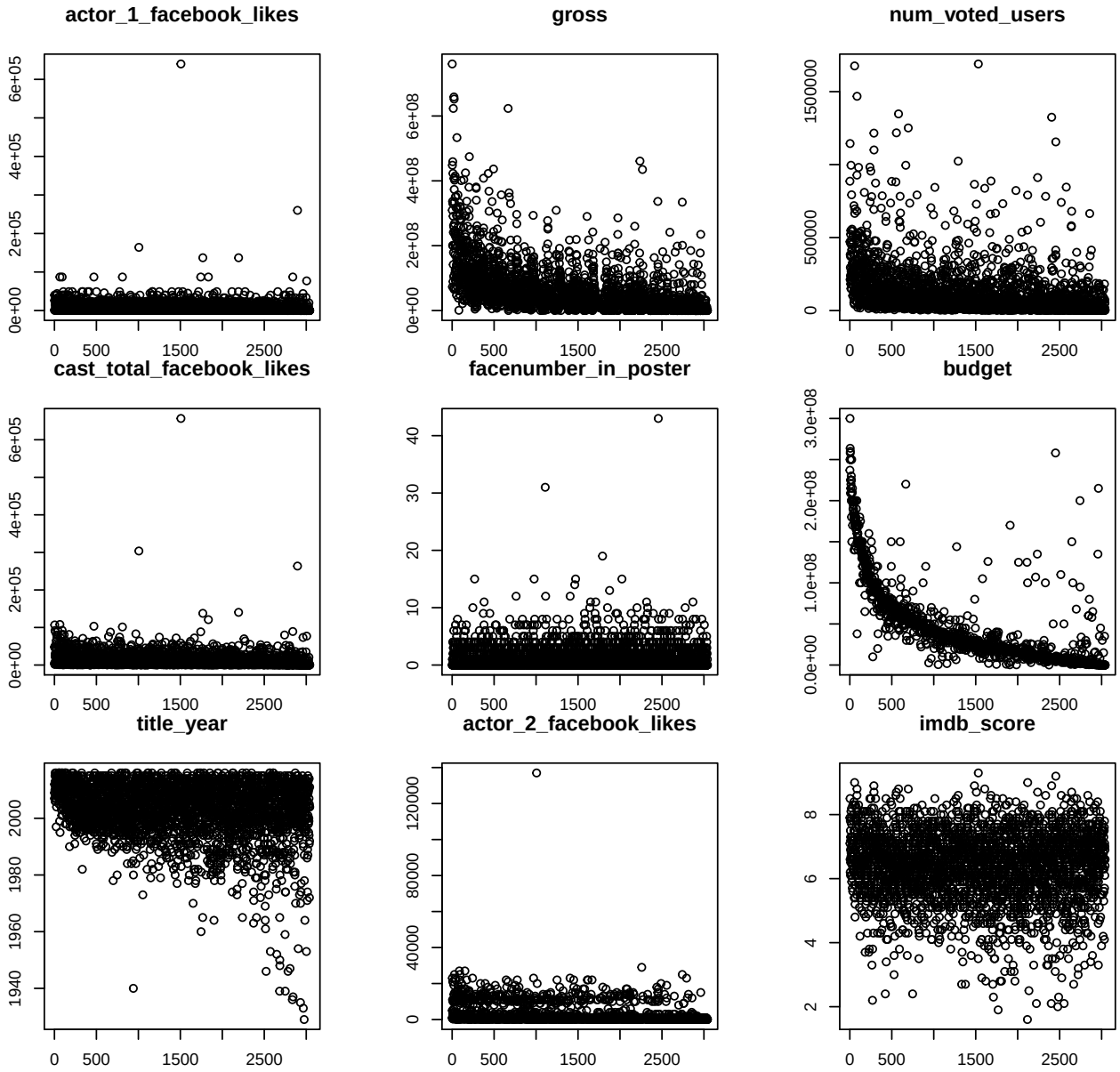







actor_1_facebook_likes
gross
num_voted_users
cast_total_facebook_likes
facenumber_in_poster
budget
title_year
actor_2_facebook_likes
imdb_score
duration
director_facebook_likes
actor_3_facebook_likes
Density
Density
Density


actor_1_facebook_likes
gross
num_voted_users
cast_total_facebook_likes
facenumber_in_poster
budget
title_year
actor_2_facebook_likes
imdb_score
duration
director_facebook_likes
actor_3_facebook_likes
Density



As we can see from the plots and statistical summaries above, most of the variables are nearly normally distributed except those variables associated with Facebook Likes. There are five variables related to Facebook Likes that are highly skewed due to a large number of zeros. An examination of the dataset source reveals that the "zero" values from the Facebook Likes were caused by simple errors in Web page scraping. At this point we assume these zeros represent NAs in the Facebook data, and we use the MICE package to impute the Facebook Likes data for the zeros/NAs.

```
##      actor_1_facebook_likes cast_total_facebook_likes
## 2502                      1                         1
##  520                      1                         1
##   10                      1                         1
##    1                      1                         1
##    6                      1                         1
##    2                      1                         1
##    1                      0                         0
##                           1                         1
```

```
##      actor_2_facebook_likes actor_3_facebook_likes director_facebook_likes
## 2502                      1                      1                       1
##  520                      1                      1                       0
##   10                      1                      0                       1
##    1                      1                      0                       0
##    6                      0                      0                       1
##    2                      0                      0                       0
##    1                      0                      0                       1
##                           9                     20                     523
##
## 2502   0
##  520   1
##   10   1
##    1   2
##    6   2
##    2   3
##    1   4
##      554
```

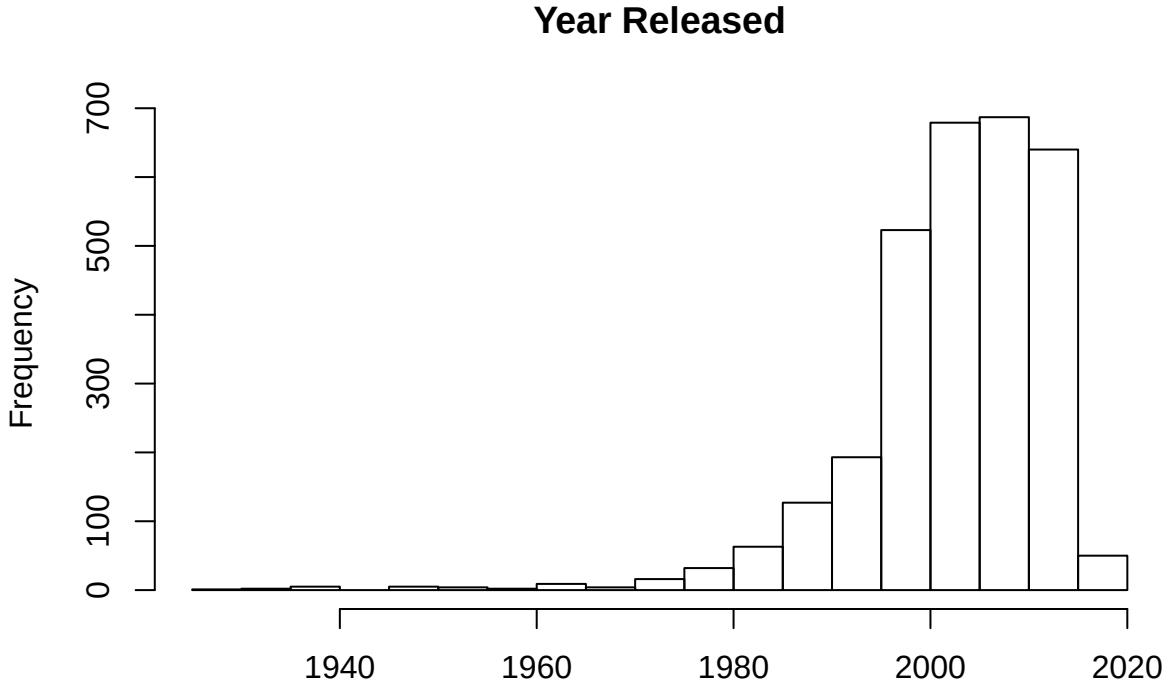
```
##     duration     director_facebook_likes actor_3_facebook_likes
##  Min.   : 37.0   Min.   :    2           Min.   :    2.0
##  1st Qu.: 95.0   1st Qu.:   32           1st Qu.:  233.0
##  Median :105.0   Median :   99           Median :  467.0
##  Mean   :109.5   Mean   : 1134           Mean   :  836.4
##  3rd Qu.:119.0   3rd Qu.:  309           3rd Qu.:  723.0
##  Max.   :330.0   Max.   :23000           Max.   :23000.0
##
##  actor_1_facebook_likes     gross           movie_title
##  Min.   :     2.0       Min.   :      703   Length:3042
##  1st Qu.:   811.2       1st Qu.: 11787482   Class :character
##  Median :  2000.0       Median : 34264376   Mode  :character
##  Mean   :  8241.6       Mean   : 57651658
##  3rd Qu.: 13000.0       3rd Qu.: 75074326
##  Max.   :640000.0       Max.   :760505847
##
##  num_voted_users   cast_total_facebook_likes facenumber_in_poster
##  Min.   :     22   Min.   :     2            Min.   : 0.000
##  1st Qu.:  19117   1st Qu.:  2210            1st Qu.: 0.000
##  Median :  54462   Median :  4517            Median : 1.000
##  Mean   : 108285   Mean   : 12340            Mean   : 1.419
##  3rd Qu.: 132124   3rd Qu.: 16904            3rd Qu.: 2.000
##  Max.   :1689764   Max.   :656730            Max.   :43.000
##
##    content_rating     budget            title_year
##  R         :1333   Min.   :      218   Min.   :1929
##  PG-13     :1110   1st Qu.: 10725000   1st Qu.:1999
##  PG        : 472   Median : 25000000   Median :2004
##  G         :  70   Mean   : 40319361   Mean   :2003
##  Not Rated:  18   3rd Qu.: 55000000   3rd Qu.:2010
##  Unrated   :  13   Max.   :300000000   Max.   :2016
##  (Other)   :  26
##  actor_2_facebook_likes   imdb_score
##  Min.   :     2.0       Min.   :1.600
##  1st Qu.:   436.0       1st Qu.:5.800
```

```
##  Median :   729.5       Median :6.500
##  Mean   :  2180.4       Mean   :6.383
##  3rd Qu.:  1000.0       3rd Qu.:7.100
##  Max.   :137000.0       Max.   :9.300
##
```

Data Preparation

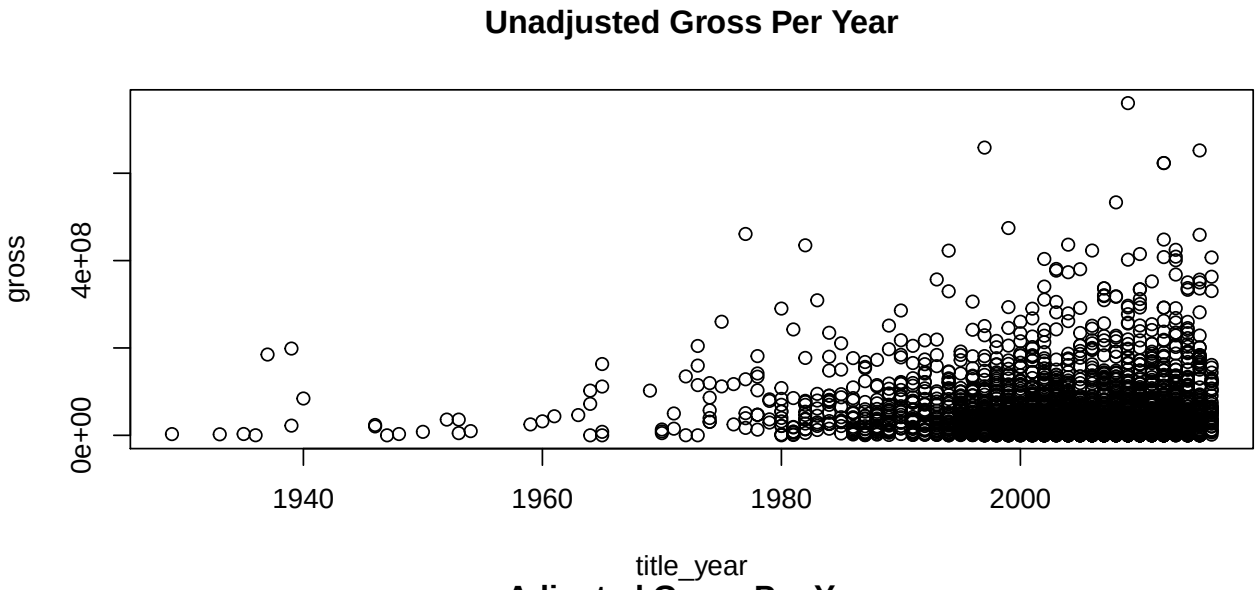
Data Preparation

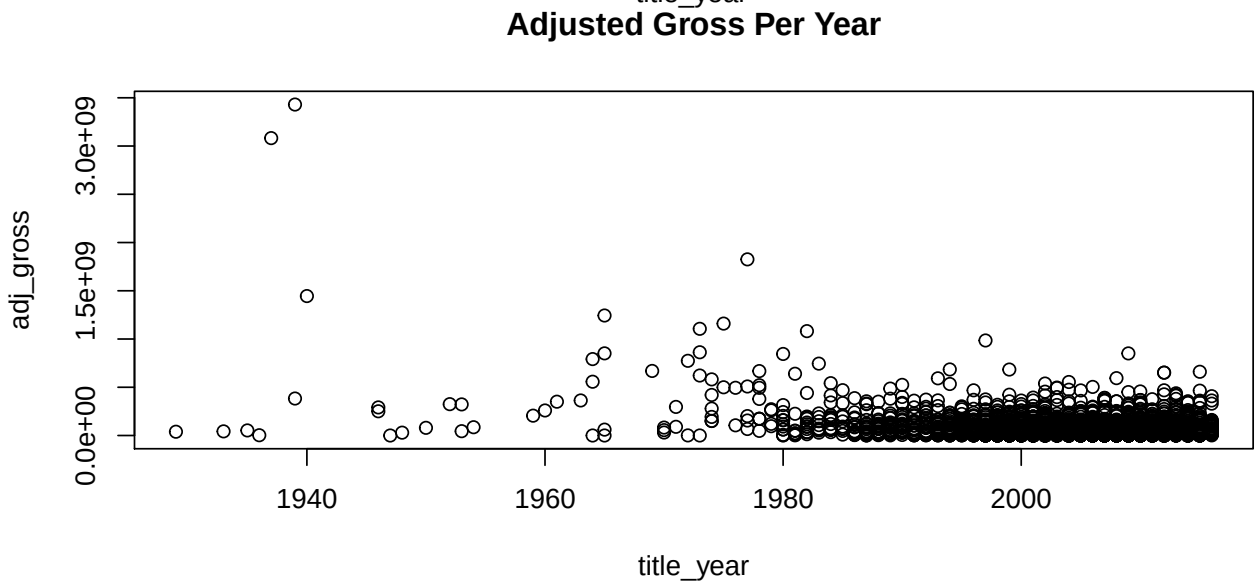
One of the big issues with using this dataset is the time-frame. These movies were released over the past 80+ years. The following histogram shows the distribution of movies released by year.



As you can see, the vast majority come from the 1990's and later, but we don't want to ignore the movies from previous years. In order to accurately portray data values from the more distant past, we institute a rate of inflation calculation. Using the Consumer Price Index released by the U.S. Bureau of Labor Statistics, we calculate the adjusted production budget and gross receipts values by year. As a basis of comparison, we use the CPI from 2016, since the last movie was released in 2016.

```
## The following object is masked _by_ .GlobalEnv:
##
##     cpi
```





From the above scatter plots, we can see that the adjustment for inflation does indeed create a more realistic picture of overall movie business revenues. (U.S. movie ticket buys peaked around 1940 and have been shrinking ever since.)

As a point of interest, the movies that made over one billion dollars (U.S. receipts only, adjusted) are shown below:

```
##                                movie_title     gross  adj_gross
## 5     Snow White and the Seven Dwarfs 184925485 3082091417
## 7                  Gone with the Wind 198655278 3430019188
## 8                           Pinocchio  84300000 1445142857
## 26                The Sound of Music 163214286 1243537417
## 39                      The Exorcist 204565000 1105756757
## 48                              Jaws 260000000 1159851301
## 53 Star Wars: Episode IV - A New Hope 460935665 1825487782
## 90         E.T. the Extra-Terrestrial 434949459 1081739587
```

A quick Google search indicates that the above movies are consistently listed as the top grossing movies

of all time. Furthermore, our "estimated adjusted gross" mimics the findings that we see with adjusted gross (for the most part, there are two schools of thought on how to adjust gross, using ticket prices or our method adjusting based on CPI). Though our dollar amounts vary slightly from other sources, any variance is consistent across our dataset, and would not negatively impact the overall results.

Build Models

Build Models

Binomial Regression

In our first model we investigate whether or not we can predict if a film will make a profit, given the cast and direction. To do this, we create a binary regression model, transforming our adjusted margin into a simple binary choice: 0 equals a loss of money, 1 equals a profit. This could be thought of as a "Go / No-Go" model.

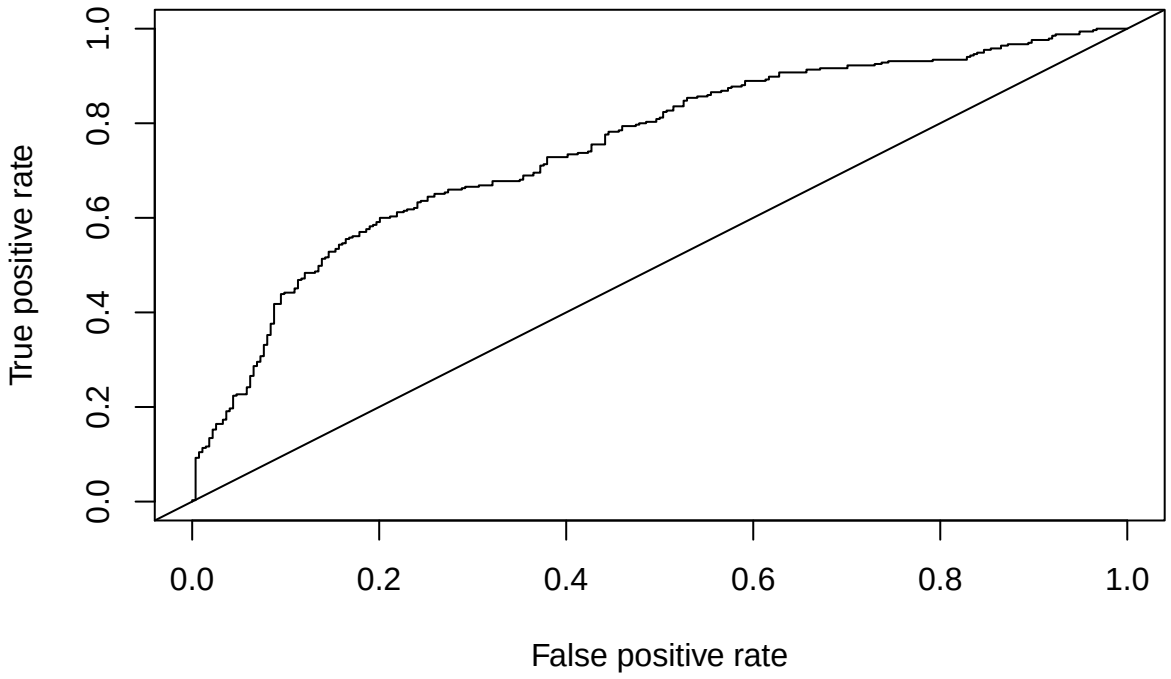
Below we utilize the binomial logistic regression function in R to create the "Go / No-Go" model.

```
##
## Call:
## glm(formula = money ~ ., family = binomial(link = "logit"), data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -4.5054  -1.1121   0.5104   1.0640   1.8663
##
## Coefficients:
##                              Estimate Std. Error z value Pr(>|z|)
## (Intercept)                 8.022e+01  1.067e+01   7.520 5.48e-14 ***
## title_year                 -3.958e-02  5.308e-03  -7.456 8.89e-14 ***
## duration                   -1.333e-02  2.465e-03  -5.409 6.34e-08 ***
## director_facebook_likes    -3.527e-05  1.389e-05  -2.539   0.0111 *
## actor_3_facebook_likes     -1.269e-04  7.471e-05  -1.699   0.0893 .
## actor_1_facebook_likes     -1.206e-04  5.029e-05  -2.399   0.0165 *
## num_voted_users             8.664e-06  6.483e-07  13.364  < 2e-16 ***
## cast_total_facebook_likes   1.165e-04  5.024e-05   2.319   0.0204 *
## facenumber_in_poster        4.109e-02  2.223e-02   1.849   0.0645 .
## actor_2_facebook_likes     -1.219e-04  5.253e-05  -2.321   0.0203 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3307.1  on 2432  degrees of freedom
## Residual deviance: 2956.5  on 2423  degrees of freedom
## AIC: 2976.5
##
## Number of Fisher Scoring iterations: 5

## [1] 0.7489051
```

Using all the prediction variables at hand, the "Go / No-Go" model accurately predicts profitability 74% of the time.

Next, using backward stepwise regression, we attempt to remove some variables that may not have significance in our model.

```
## Start:  AIC=2976.51
## money ~ title_year + duration + director_facebook_likes + actor_3_facebook_likes +
##     actor_1_facebook_likes + num_voted_users + cast_total_facebook_likes +
##     facenumber_in_poster + actor_2_facebook_likes
##
##                              Df Deviance    AIC
## <none>                            2956.5 2976.5
## - actor_3_facebook_likes      1   2959.4 2977.4
## - facenumber_in_poster        1   2960.0 2978.0
## - actor_2_facebook_likes      1   2962.2 2980.2
## - cast_total_facebook_likes   1   2962.2 2980.2
## - actor_1_facebook_likes      1   2962.6 2980.6
## - director_facebook_likes     1   2962.9 2980.9
## - duration                    1   2986.8 3004.8
## - title_year                  1   3019.1 3037.1
## - num_voted_users             1   3243.1 3261.1

##
## Call:
## glm(formula = money ~ title_year + duration + director_facebook_likes +
##     actor_3_facebook_likes + actor_1_facebook_likes + num_voted_users +
##     cast_total_facebook_likes + facenumber_in_poster + actor_2_facebook_likes,
##     family = binomial(link = "logit"), data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -4.5054  -1.1121   0.5104   1.0640   1.8663
##
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)                  8.022e+01  1.067e+01   7.520 5.48e-14 ***
## title_year                  -3.958e-02  5.308e-03  -7.456 8.89e-14 ***
## duration                    -1.333e-02  2.465e-03  -5.409 6.34e-08 ***
## director_facebook_likes     -3.527e-05  1.389e-05  -2.539   0.0111 *
## actor_3_facebook_likes      -1.269e-04  7.471e-05  -1.699   0.0893 .
## actor_1_facebook_likes      -1.206e-04  5.029e-05  -2.399   0.0165 *
## num_voted_users              8.664e-06  6.483e-07  13.364  < 2e-16 ***
## cast_total_facebook_likes    1.165e-04  5.024e-05   2.319   0.0204 *
## facenumber_in_poster         4.109e-02  2.223e-02   1.849   0.0645 .
## actor_2_facebook_likes      -1.219e-04  5.253e-05  -2.321   0.0203 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3307.1  on 2432  degrees of freedom
## Residual deviance: 2956.5  on 2423  degrees of freedom
## AIC: 2976.5
##
## Number of Fisher Scoring iterations: 5
```




```
## [1] 0.7489051
```

As you can see, using backward stepwise regression produces slightly better AIC scores. However, the AUC decreases slightly, which is undesirable. Another revelation is that the Director Facebook Likes score is not a significant variable in our model, and is thus removed by the backward stepwise regression process. It appears that for our purposes here, the actors' Facebook Likes are better indicators of profitability than Directors', which goes to show how the industry has unfolded. A few directors have become prominent in our culture, but the recognizability of actors and actresses determines more strongly whether or not a movie will make money.

As a final step, we use a confusion matrix to show the relative strength of our model.

```
##            Reference
## Prediction   0   1
##          0 157  84
##          1 117 251
```

As you can see we tend to have more false negatives than false positives, and the break down of accuracy, specificity, precision and F1-score can be seen below:

Parameters	Model1
Accuracy	0.6699507
Classification Error Rate	0.3300493
Precision	0.6514523
Sensitivity	0.5729927
Specificity	0.7492537
F1 Score	0.6097087

Gross Margin Model

We now have a prediction model for whether a particular movie will make money or not. However, from a movie investor's perspective, that information is not sufficient to make a decision about which movie is the most attractive to fund when faced with multiple options. Therefore, another focus of our final project is to build a model that attempts to predict how much money a movie will make (Gross Margin).

Below we build a multivariable linear regression model for Gross Margin, without any data transformation.

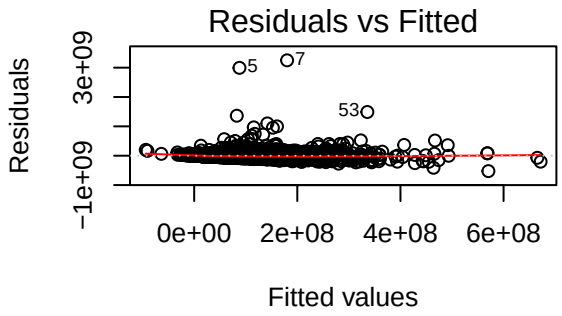
```
##               title_year                  duration
##                        0                         0
##   director_facebook_likes    actor_3_facebook_likes
##                        0                         0
##    actor_1_facebook_likes                     gross
##                        0                         0
##          num_voted_users cast_total_facebook_likes
##                        0                         0
##      facenumber_in_poster                    budget
##                        0                         0
##    actor_2_facebook_likes                imdb_score
##                        0                         0
##                      cpi                 adj_gross
##                        0                         0
##               adj_budget                adj_margin
##                        0                         0
```

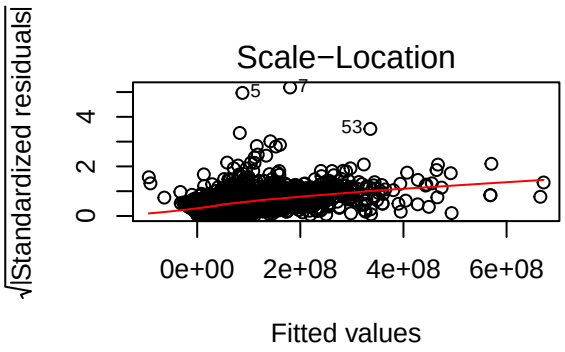
```
##               title_year                  duration
##                        0                         0
##   director_facebook_likes    actor_3_facebook_likes
##                        0                         0
##    actor_1_facebook_likes                     gross
##                        0                         0
##          num_voted_users cast_total_facebook_likes
##                        0                         0
##      facenumber_in_poster                    budget
##                     1259                         0
##    actor_2_facebook_likes                imdb_score
##                        0                         0
##                      cpi                 adj_gross
##                        0                         0
##               adj_budget                adj_margin
##                        0                         0
```

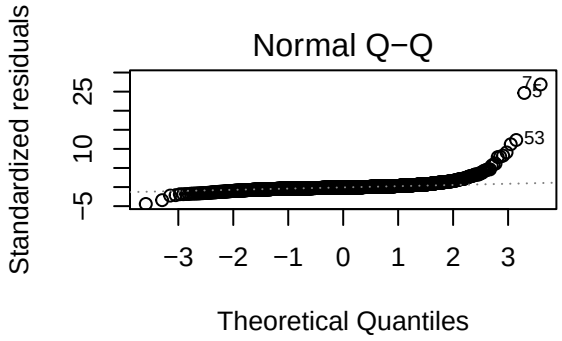
Our first attempt at a Gross Margin model does not show good performance. The AIC is 121904.1, BIC is 121970.4 and logLik is -60941.1. After we take a look at the histogram of the residuals plot, we realize it is highly skewed to the right. The skewness is as high as 13.7.

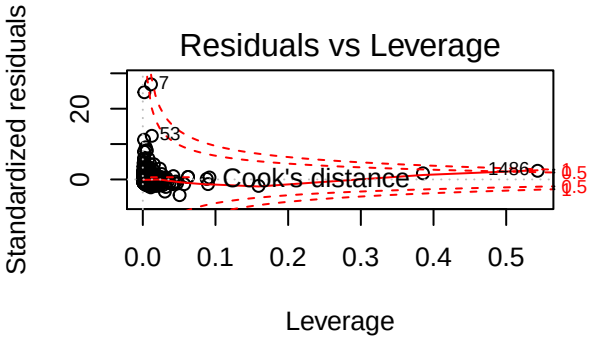
```
##
## Call:
## lm(formula = adj_gross ~ (duration + director_facebook_likes +
##     actor_3_facebook_likes + actor_1_facebook_likes + num_voted_users +
```

```
##     cast_total_facebook_likes + actor_2_facebook_likes + imdb_score +
##     adj_budget + adj_margin) - adj_margin, data = movies1)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -524681457  -36734530  -14575833   15197005 3249856255
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)               -8.797e+07  1.710e+07  -5.146 2.84e-07 ***
## duration                   2.437e+05  1.164e+05   2.093 0.036395 *
## director_facebook_likes   -1.715e+03  6.322e+02  -2.713 0.006713 **
## actor_3_facebook_likes    -1.008e+04  3.158e+03  -3.194 0.001419 **
## actor_1_facebook_likes    -7.542e+03  1.924e+03  -3.920 9.05e-05 ***
## num_voted_users            2.827e+02  1.880e+01  15.039  < 2e-16 ***
## cast_total_facebook_likes  7.224e+03  1.920e+03   3.762 0.000172 ***
## actor_2_facebook_likes    -7.942e+03  2.033e+03  -3.906 9.59e-05 ***
## imdb_score                 1.248e+07  2.588e+06   4.823 1.48e-06 ***
## adj_budget                 6.696e-01  5.130e-02  13.054  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 121600000 on 3032 degrees of freedom
## Multiple R-squared:  0.2687, Adjusted R-squared:  0.2665
## F-statistic: 123.8 on 9 and 3032 DF,  p-value: < 2.2e-16

##   r.squared adj.r.squared     sigma statistic       p.value df    logLik
## 1 0.2686814     0.2665106 121572655  123.7707 1.147397e-198 10 -60941.34
##        AIC      BIC     deviance df.residual
## 1 121904.7 121970.9 4.481269e+19        3032
```

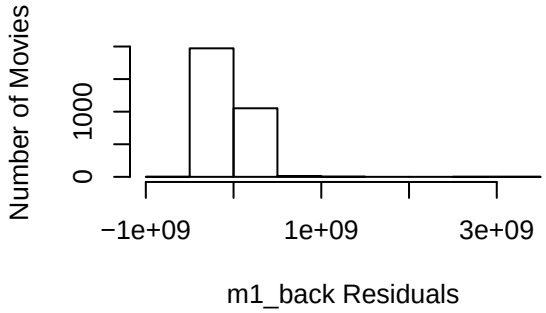








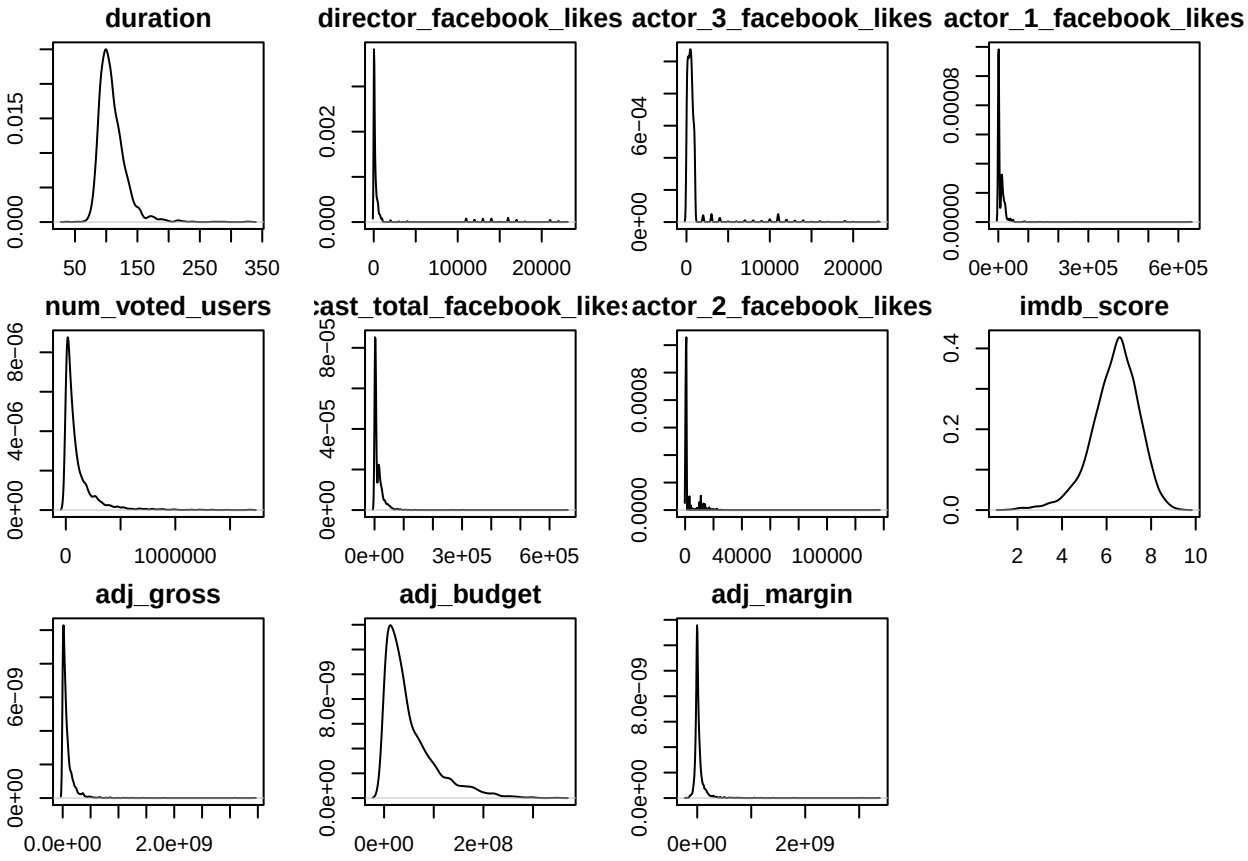

```
## [1] 13.6585
## attr(,"method")
## [1] "moment"
```



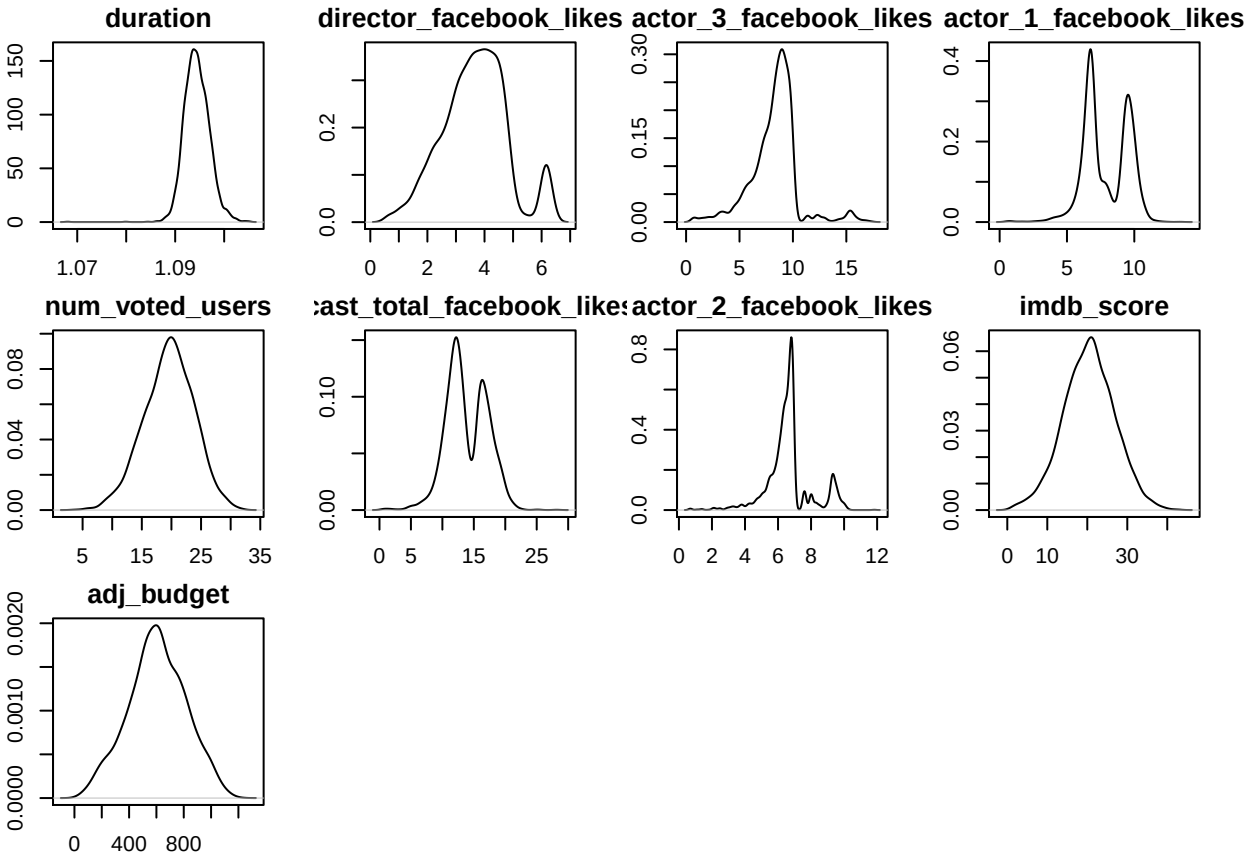
The weak performance is most likely due to the abnormal skewness and kurtosis from the original data. Therefore, we attempt to correct the problem by employing Box-Cox transformation.

The following compares distributions of the two datasets. The first is the dataset before transformation; the second, after.

Before Transformation:



After transformation:

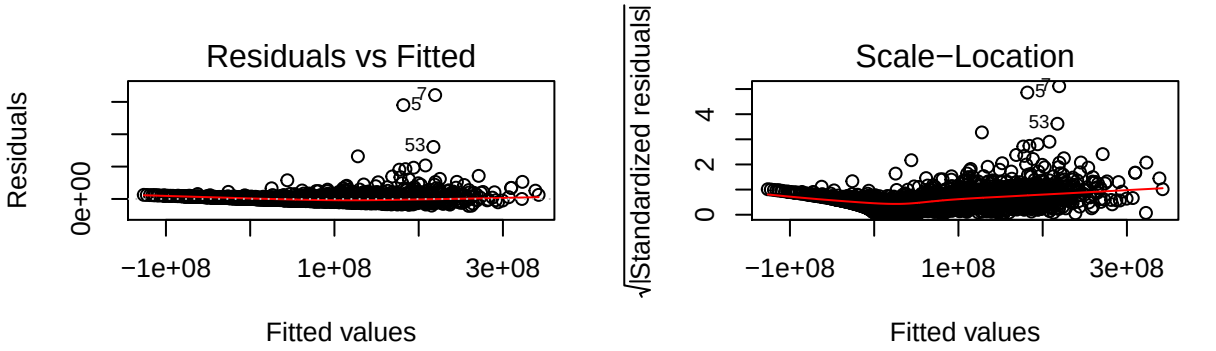


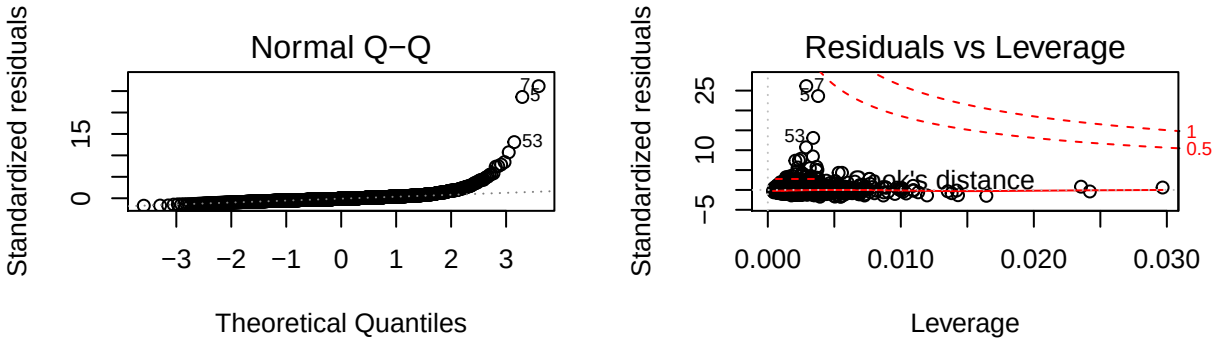
From this comparison we can observe that the Box-Cox transformation approximately normalizes the data, so we proceed.

Our second attempt at the Gross Margin model has very similar AIC, BIC, and loglikelihood values as the first. However, the skewness of the residual histogram is reduced. Therefore, the second attempt is a superior model.

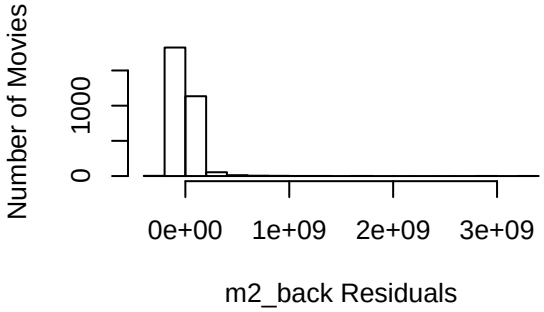
```
##
## Call:
## lm(formula = adj_gross ~ actor_1_facebook_likes + num_voted_users +
##     cast_total_facebook_likes + actor_2_facebook_likes + imdb_score +
##     adj_budget, data = movies2)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -213301475  -48954766  -14875368   26028702 3210453166
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)               -156278007   16691895  -9.363  < 2e-16 ***
## actor_1_facebook_likes     -38065352    6545721  -5.815 6.68e-09 ***
## num_voted_users              8848153     795705  11.120  < 2e-16 ***
## cast_total_facebook_likes   15896608    3989105   3.985 6.91e-05 ***
## actor_2_facebook_likes      -5206564    3315062  -1.571    0.116
## imdb_score                   3757599     429346   8.752  < 2e-16 ***
## adj_budget                    170517      13381  12.744  < 2e-16 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.23e+08 on 3035 degrees of freedom
## Multiple R-squared:  0.2508, Adjusted R-squared:  0.2493
## F-statistic: 169.3 on 6 and 3035 DF,  p-value: < 2.2e-16

##   r.squared adj.r.squared     sigma statistic     p.value df    logLik
## 1 0.2508153     0.2493342 122987877  169.3451 3.5576e-186  7 -60978.06
##        AIC      BIC     deviance df.residual
## 1 121972.1 122020.3 4.590746e+19        3035
```






```
## [1] 12.63841
## attr(,"method")
## [1] "moment"
```



We apply the corrected values to our model. Finally, we create a master dataframe containing our predicted results and actual data.

Final output of Gross Margin model:

```
##                       Movie Title Actual Adjusted Gross Predicted Gross
## 1             The Broadway Melody              39181395         4338440
## 2                     42nd Street              42790698        38354799
## 3                         Top Hat              52554745        49890641
## 4                     Modern Times               2818619       174655298
## 5 Snow White and the Seven Dwarfs            3082091417       181798508
## 6                 The Wizard of Oz             383354452       210518781
##   Actual Profit Margin Predicted Profit Margin
## 1            0.8650285              -0.2189570
## 2            0.8091304               0.7870555
## 3            0.7970000               0.7861600
## 4           -8.1886428               0.8517120
## 5            0.9891848               0.8166468
## 6            0.8738887               0.7703515
```

The predicted profit margin variable can serve as a reference for investors to decide if they want to contribute to the production of a movie and share the profit that is generated.

Since we have the actual profit margin variable available to us, we can also investigate if the quality of the movie will have any impact on the profitability of the movie. We use the IMDB rating as an analog for movie quality.

```
##
##  Pearson's product-moment correlation
##
## data:  movies$imdb_score and movies$profit_margin
## t = 2.6579, df = 3040, p-value = 0.007905
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.01263244 0.08354498
## sample estimates:
##        cor
## 0.04814938
```

The result above suggests only a weak positive relationship between the quality and profitability of movies. The p-value is 0.007905, which is less than the significance level of 0.05. In addition, the 95% confidence interval is (0.01263244, 0.08354498), which does not cross zero. It also shows the result is statistically significant. However, the correlation coefficient is only 0.048 (about 5% higher profit), which is a weak association between the two variables.

Therefore, if investors care more about profitability, it is recommended not to care too much about the quality of the movie. Spending huge amounts of money on improving movie quality may lead to minuscule returns.

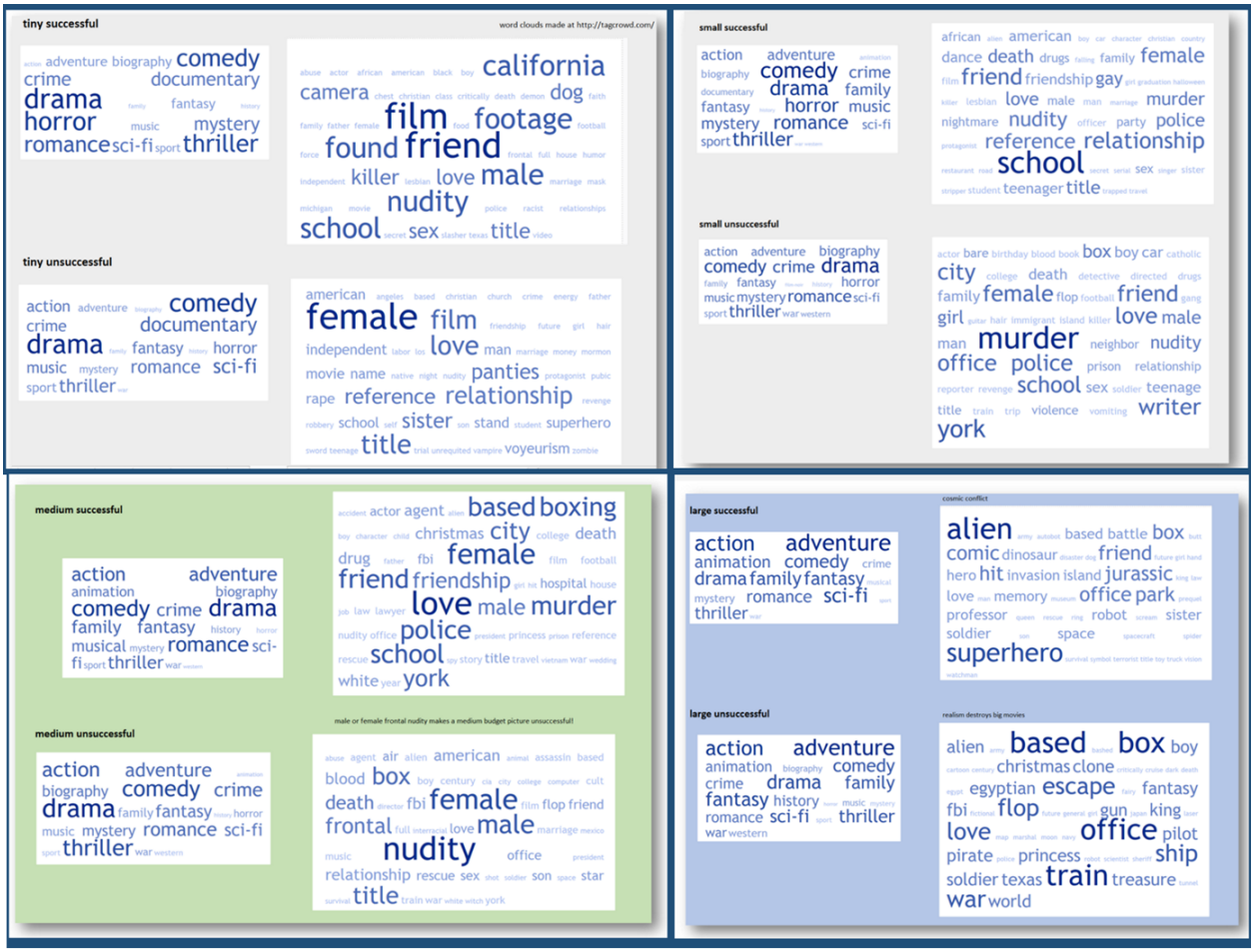
Plot Keywords

The Kaggle dataset includes a nefarious-looking collection of plot keywords for each movie. We could not resist exploring them.

To do so required some manual cleanup of typo's, duplicates, and the addition of a small amount of information from Ian Cavalier's Filmometer, as well as IMDB's Parent's Guide. We also have an additional variable called Movie Scale, which categorizes each movie into one of four groups:

```
Movie Scale -|- CPI-Adjusted Budget
-------------|------------------------------
Tiny         |  Less than 5 million
Small        |  Between 5 and 25 million
Medium       |  Above 25, Below 100 million
Large        |  100 million and above
-------------|------------------------------
```

To get some idea of where to start looking, we use a free online tool called TagCrowd to paint word clouds of movie genre and keywords by movie scale:



Tiny movie genres and keywords are shown top left, followed by small films to the right. Medium and large films are in the bottom two blocks. Within each block, successfilm film genres and keywords are shown on top, and unsuccessful ones below.

There are many questions one could ask about this information. Some examples:

1. For tiny-budget films, do male-oriented crime films with graphic sex and violence perform better at the box office than films about female relationships?

2. For large films, does escapism trump realism?
3. Is there a limited U.S. audience for films with nudity? This seems to be the opposite of "sex sells."

Since nudity is a kind of fun and perhaps taboo topic that weaker souls than us would carefully avoid writing about in a graduate student paper on regression analysis, let's tackle question 3.

We split each budget category into two groups - movies with nudity and movies without. (Note: Some external sources were used to scrub the quality of the keywords for this variable).

Our variable "Gross Over Budget" is the adjusted gross receipts of the movie (U.S. Box Office only) divided by the adjusted production budget. While this is not an indicator of the true profitability of the movie because it doesn't include foreign screenings, DVD sales, etc, it does help provide a rough proportional gross margin percentage.

Movies without nudity: (V1 = Gross Over Budget)

```
##    movie_scale        V1
## 1:       large  1.167507
## 2:      medium  2.057582
## 3:       small  2.734082
## 4:        tiny 38.424806
```

Counts:

```
##    movie_scale   V1
## 1:       large  337
## 2:      medium 1001
## 3:       small  632
## 4:        tiny  206
```

Movies with nudity: (V1 = Gross Over Budget)

```
##    movie_scale       V1
## 1:       large 1.006667
## 2:      medium 1.680380
## 3:       small 2.784118
## 4:        tiny 6.635000
```

Counts:

```
##    movie_scale V1
## 1:       large  9
## 2:      medium 79
## 3:       small 68
## 4:        tiny 28
```

We can see a kind of auto-filtering going on here in the industry. Only nine large movies include nudity, while 337 do not! 98% of large films avoid nudity at all costs. Among medium budget films, only about 7% include nudity, and about 11% of small and tiny films do.

Furthermore, we can see that the average gross margin percents are lower for all categories of movies with nudity, except small budget. Among large and medium films, the difference is about 16%! Tantalizing...

We create some linear regression models to determine if these results are significant:

```
##
## Call:
## lm(formula = gross_over_budget ~ content_rating + imdb_score +
##     num_critic_for_reviews + num_user_for_reviews + nudity, data = tiny.US)
##
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -577.97  -22.01    5.68   27.04 2303.98
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 94.33157  213.17332   0.443  0.65855
## content_ratingNC-17        -33.59518  215.73785  -0.156  0.87639
## content_ratingNot Rated      8.38265  203.46971   0.041  0.96717
## content_ratingPG           -19.00483  198.73121  -0.096  0.92390
## content_ratingPG-13        -22.77117  195.40845  -0.117  0.90734
## content_ratingR            -29.24621  193.61503  -0.151  0.88007
## content_ratingUnrated      382.97853  206.09186   1.858  0.06445 .
## content_ratingX           -215.73187  274.18059  -0.787  0.43222
## imdb_score                 -12.83142   13.79428  -0.930  0.35328
## num_critic_for_reviews      -0.45565    0.15627  -2.916  0.00391 **
## num_user_for_reviews         0.52248    0.05263   9.928  < 2e-16 ***
## nudityy                    -13.31237   40.35422  -0.330  0.74180
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 192.7 on 222 degrees of freedom
## Multiple R-squared:  0.3694, Adjusted R-squared:  0.3381
## F-statistic: 11.82 on 11 and 222 DF,  p-value: < 2.2e-16


##
## Call:
## lm(formula = gross_over_budget ~ content_rating + imdb_score +
##     num_critic_for_reviews + num_user_for_reviews + nudity, data = small.US)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -12.237  -1.759  -0.952   0.520  43.884
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                1.4710209  4.6132193   0.319   0.7499
## content_ratingG           -1.7435141  4.6075352  -0.378   0.7052
## content_ratingNot Rated   -1.3935283  5.4385295  -0.256   0.7978
## content_ratingPassed       3.1628084  6.2824949   0.503   0.6148
## content_ratingPG          -0.7888556  4.4799758  -0.176   0.8603
## content_ratingPG-13       -2.2194300  4.4644538  -0.497   0.6193
## content_ratingR           -2.2825759  4.4574104  -0.512   0.6088
## content_ratingUnrated      0.3528819  6.2780783   0.056   0.9552
## content_ratingX           -0.1745978  4.9729803  -0.035   0.9720
## imdb_score                 0.3925761  0.1659640   2.365   0.0183 *
## num_critic_for_reviews    -0.0013034  0.0019989  -0.652   0.5146
## num_user_for_reviews       0.0049277  0.0007593   6.490 1.65e-10 ***
## nudityy                   -0.8200637  0.6021214  -1.362   0.1737
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.439 on 687 degrees of freedom
## Multiple R-squared:  0.1001, Adjusted R-squared:  0.08439
```

```
## F-statistic: 6.369 on 12 and 687 DF,  p-value: 8.817e-11

##
## Call:
## lm(formula = gross_over_budget ~ content_rating + imdb_score +
##     num_critic_for_reviews + num_user_for_reviews + nudity, data = medium.US)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -30.945  -1.109  -0.357   0.521  60.587
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)               2.470e+01  2.077e+00  11.892  < 2e-16 ***
## content_ratingG          -2.611e+01  2.013e+00 -12.970  < 2e-16 ***
## content_ratingM          -1.539e+01  4.222e+00  -3.646 0.000279 ***
## content_ratingNot Rated -2.152e+01  3.269e+00  -6.582 7.26e-11 ***
## content_ratingPassed     -2.694e+01  3.270e+00  -8.239 5.06e-16 ***
## content_ratingPG         -2.770e+01  1.913e+00 -14.478  < 2e-16 ***
## content_ratingPG-13      -2.918e+01  1.907e+00 -15.305  < 2e-16 ***
## content_ratingR          -2.947e+01  1.906e+00 -15.464  < 2e-16 ***
## content_ratingUnrated    -2.568e+01  2.883e+00  -8.907  < 2e-16 ***
## imdb_score                9.127e-01  1.232e-01   7.410 2.57e-13 ***
## num_critic_for_reviews   -1.435e-03  1.313e-03  -1.093 0.274745
## num_user_for_reviews      1.744e-03  4.207e-04   4.147 3.64e-05 ***
## nudityy                  -2.815e-01  4.738e-01  -0.594 0.552479
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.775 on 1067 degrees of freedom
## Multiple R-squared:  0.2816, Adjusted R-squared:  0.2735
## F-statistic: 34.85 on 12 and 1067 DF,  p-value: < 2.2e-16

##
## Call:
## lm(formula = gross_over_budget ~ content_rating + imdb_score +
##     num_critic_for_reviews + num_user_for_reviews + nudity, data = large.US)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.2834 -0.4418 -0.0620  0.3113  3.9768
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -8.352e-01  2.948e-01  -2.833  0.00488 **
## content_ratingPG         -5.706e-03  1.525e-01  -0.037  0.97018
## content_ratingPG-13      -1.314e-01  1.473e-01  -0.892  0.37283
## content_ratingR          -3.902e-01  1.611e-01  -2.422  0.01597 *
## imdb_score                2.984e-01  4.042e-02   7.383 1.22e-12 ***
## num_critic_for_reviews   -7.855e-04  2.815e-04  -2.790  0.00557 **
## num_user_for_reviews      5.305e-04  6.086e-05   8.717  < 2e-16 ***
## nudityy                   7.729e-04  2.281e-01   0.003  0.99730
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.6207 on 338 degrees of freedom
## Multiple R-squared:  0.3762, Adjusted R-squared:  0.3632
## F-statistic: 29.12 on 7 and 338 DF,  p-value: < 2.2e-16
```

According to the models, there is no statistical significance to the variable Nudity at any of the budget sizes!

Of the factors considered, for tiny and small budget movies, the number of user reviews is the most significant and helpful variable for profitability. For medium and large budget films, the number of user reviews remains important but the IMDB score enters the picture as far more impacting.

It would be interesting to find out why the IMDB score is not significant for tiny and small movies, but it is very significant for medium and large movies.

The lesson learned is simple - just because the average value of one group is different from another does not mean there is a significant difference between them. In this case, the sample size may be too low. If time permitted, more sophisticated techniques, such as zero-inflated models, could be deployed.

Concluding Thoughts:

There are many companies doing very sophisticated analysis to assess the viability of potential movie projects. One is called The Numbers. It uses a data source called Opus Data. The models are very intricate.

It is impossible to create competitive models with a sample dataset like ours, but it is also kind of amazing that we were able to create a "Go No/Go" model with 74% accuracy, and a robust framework for predicting gross margin percentage for a given movie. We also had a little fun illustrating the dangers of using simple averages to prove a point, rather than robust modeling. We hope you learned some new things about the movie business and had fun doing so!

Smooth Operators - All Done!