Table of Content

# Section 1: Introduction

# Section 2: Data Description

# Section 3: Model Selection

# Section 4: Results

# Section 5: Appendices

# Appendix

**Section 1: Introduction**

This project utilizes the statistical foundations of predictive data analytics using R programming language. The goal of the project is to accurately predict the IMDb ratings of the twelve upcoming movies. As a team, our objective is to find a superior model with an optimal selection of predictors, degree relationships, and deliver a clear, concise report that identifies the model selection which can get the most accurate IMDb score prediction of the 12 movies.

The process involves various examinations like predictor significance, non-linearity, heteroskedasticity, outliers, and collinearity. In addition to grouping selected variables, different regression analyses, and excluding variables in order to enhance the model's predictive power. Spoiler Alert: Our finalized ratings in Section 4 might tempt you to renew your Netflix subscription, stay home, and get your own popcorn!

# **Section 2: Data Description**

In the original dataset, there were 42 attributes including the target variable: IMDb scores for each movie. To downsize the models and improve the efficiency, multiple data preprocessing steps were performed to ensure that we have a comprehensive understanding of the data and that it was concise enough for model building.

**2.1 Subset**

Before we started to manipulate the dataset, a subset from the original dataset was created upon our agreement to secure an unmodified reference. In our subset, called “our dataset” in the following paragraphs, the label variables such as *movieID* were eliminated since they were identical to each observation and would not be able to provide valuable information for model formulation. The variable *genres* was removed as well as it is already dummified into numerous columns.

**2.2 Categorical and numerical attribute segregation**

The initial step we took was to segregate categorical and numerical predicators. Columns with alphabetical values are identified as categorical variables. Yet attributes in numbers could still be classified as categorical ones still based on the nature of them. For instance, *aspectRatio* has numerical values such as 1.85, yet it is actually representing different types of films. After identifying the types of all variables, categorical columns were converted into factors to enable us to plug them into the models.

**2.3 Removing insignificant predictors**

Several variables are excluded as we observed that there was no significant relevance to the target value from the scatter plots, including *releaseDay*, *releaseMonth*, and *releaseYear*. Variables that are highly related to others were also removed. For instance, *productionCompany* and *distributor* are distinctly related to *director*. Thus, the first two were dropped.

**2.4 Cardinality**

We continued the preprocessing procedure with null value checks in our dataset on all the columns and found that most rows of the predictor *nbFaces* had 0 value. The homogeneity observed in this column might not attribute much to our prediction model and *nbFaces* was subsequently dropped from our final training dataset. Other categorical columns with either relatively high cardinality or uniform values were dropped. For example, the *language* column of more than 90% of the movies in the original dataset were English thus this variable was dropped as it gave us neither valuable information nor predictive power and was correlated to the attribute *country* as well.

**2.5 Derived variable**

To utilize the information more efficiently, we decided to aggregate or merge specific variables. Based on our general understanding, directors of movies have notable impacts on movie ratings. Instead of having a categorical variable with hundreds of levels, we calculated the average movie ratings for all directors and ranked them accordingly. Furthermore, we grouped the directors by their ranking. For example, directors rank 1 to rank 10 will be in tier 1, directors rank 11 to rank 20 will be in tier 2 etc. Another example of derived variables is that we computed the average of *actor1\_starMeter + actor2\_starMeter + actor3\_starMeter* along with the log of each variable to test for significance, yet the *actor1\_starMeter* resulted in the highest significance and was selected in the model configuration.

**2.6 Final adjustment**

Our entire dataset was subsetted to only color films based on the column *ColourFilm* as all movies released in recent decades are mostly colored. It was also observed that relatively old, black and white films generally had higher ratings, and thus will be potentially skewing the results. This relationship is presented in Plot 1 in the appendix. Once our training dataset was organized after performing all the pre-processing steps mentioned above, multiple models were built based on the filtered predictors and the best fits will be identified.

# **Section 3: Model Selection**

**3.1 Regression Analyses**

Polynomial regression of varying degrees from 1 to 6 and “anova” test was conducted to select the optimal degree relationship (if p < 0.05, the next degree model was selected). This process was repeated for each numerical predictor and the optimal degree was recorded, referring to Table 1a to Table 1e.

In addition to polynomial regression, two models of Spline regression were performed with k = 5 and k = 2, but no significant improvement in the adjusted r2 was observed, and the observations were prone to overfitting. Therefore, the spline regression was neglected.

**3.2 Model Issues**

We have conducted four tests to prevent model issues.

**3.2.1** **Non-linearity of predictors**

Residual plots of numerical predictors were created to determine the non-linearity check (if P > 0.1, the predictor was considered linear). Only *nbFaces* variable was linear, referring to Table 2 and Plot 2.

**3.2.2 Heteroskedasticity**

The ncv test was conducted on the optimal model selected. The p-value was < 0.5 so heteroskedasticity issue was identified and resolved using coeftest. This allowed the optimal model to be unbiased and that all predictors’ significance is not artificially increased or decreased, referring to Table 3.

**3.2.3 Outliers**

Bonferroni numerical outlier test was conducted separately for each predictor and outliers were removed to enhance predictive power and mitigate the significant influence of certain abnormal observations. Outlier test results are presented in Table 4a to Table 4e in the appendix.

**3.2.4 Collinearity**

Collinearity was identified using the Variance Inflation Factors (VIF) Test, referring to Table 5.

**3. 3 Model Selection**

After narrowing the predictor selection process, the model was tested based on interchangeable degrees, predictors, and regression relationships to identify the highest adjusted r2 value.

**Model 11** scored the highest and was compared against the rest of the models with cross-validation testing to identify the MSE margin.

Our focus was to achieve a minimal number of predictors while maintaining the highest adjusted r2 value and minimizing the MSE. This will allow the selected model to achieve high predictive power with fewer overfitting issues and the least number of predictors. Refer to Table 6 for model comparison.

# **Section 4: Results**

**4.1 Final Model and Prediction**

The final model that we have selected is mathematically represented as,

y = β0 + β1 × *movieBudget*+ β2 × *duration* + β3 × *duration* 2+ β4 × *nbNewsArticles* + β5 × *nbNewsArticles* 2+ β6 × *nbNewsArticles* 3+ β7 × *nbFaces* + β8 × *western* + β9 × *drama* + β10 × *animation* + β11 × *ranking\_bin2* + β12 × *ranking\_bin3* + β13 × *ranking\_bin4* + β14 × *ranking\_bin5* + β15 × *ranking\_bin6* + β16 × *ranking\_bin7* + β17 × *ranking\_bin8* + β18 × *ranking\_bin9* + β19 ×*ranking\_bin10*

where **imdbScore** is our dependant variable represented by y.

**4.2 Test Set Processing and Predictions**

To ensure the test set is compatible with our model, we followed similar data preprocessing as we did with the training set. We got rid of predictors like *colourFilm, movieID, releaseYear* etc. to ensure the number of predictors in train and test set are the same. For derived columns like *ranking\_bin,* we derived the bins that each director should be placed in and then left joined that data with the test set to recreate the *ranking\_bin* column in the test set. We proceeded with the predictions once all the predictors were aligned between the test and train set.

Following are our predictions for the 12 movies using the optimal model selected:

|  |  |
| --- | --- |
| **Movie Name** | **Predicted Score on IMDb** |
| Falling for Christmas | 6.44180 |
| Black Panther: Wakanda Forever | 6.19261 |
| Spirited | 5.81863 |
| Paradise City | 5.47408 |
| Poker Face | 6.55002 |
| Que Viva Mexico! | 6.97907 |
| Slumberland | 5.98879 |
| Blue's Big City Adventure | 6.75056 |
| The Menu | 5.77393 |
| The Fabelmans | 7.67115 |
| Devotion | 6.42526 |
| Strange World | 6.26543 |

**4.3 Performance and Validation**

The R2 of the model comes out to be 0.7481, which implies that 74.81% of the variance in the data can be explained by our model.

The MSE calculated after splitting the dataset by an 8:2 ratio into train and test set respectively is **0.2765225**.

The MSE calculated after performing a LOOCV test is **0.2952443**.

The MSE calculated after performing a K-fold validation test with k=5 is **0.2965276**.

With an average MSE of **0.2894** across the 3 types of tests, our MSE’s are stable and without much deviation for each method.

**4.4 Parameter Analysis**

The following tables show us the coefficients (see appendix 4.1) and significance (see appendix 4.2) of each of the independent variables after performing the heteroskedasticity transformation. All predictors selected are significant with p-value less than 0.05 except for,

* β3 (p-value = 0.16) corresponding to *duration*2 was included since a polynomial function with degree 2 was found out to be the best fit (highest R2) between *imdbScore* and *duration* when modelled individually.
* β6 (p-value = 0.25) corresponding to *nbNewsArticles*3 was included since a polynomial function with degree 3 was found out to be the best fit (highest R2) between *imdbScore* and *nbNewsArticles* when modelled individually.
* β7 (p-value = 0.16) corresponding to *nbFaces*was included since from a business standpoint, the number of faces often has a significant effect on the rating of the movie in terms of having big stars. Furthermore, the p-value is low suggesting there is an 84% chance the predictor is significant

**4.5 Business Insights**

Analysing the coefficients for *ranking\_bin*i (for i = 2,3...10), having directors outside the top 10 percentile (*ranking\_bin1*) has a negative effect on the *imdbScore* and it slowly becomes worse with each bucket towards the bottom. For example, β11=-0.34 means that if a movie has a director from the *ranking\_bin*2 it will result in 0.34 decrease in the *imdbScore*. Same applies for β12 to β19. Movies belonging to the animation genre have the largest effect on *imdbScore* with β10=0.37, implying an animation movie will tend to have a 0.37 increase in their score.

**4.6 Future Recommendations/Improvements**

Based on the review of the dataset and information collected by watching movies all our lives, there are multiple recommendations that we can think of to improve the overall predictive power of our model.

* Our Model is over reliant on a few predictors: Though *budget* and *duration* are good indicators of movie rating, they fall short of capturing a lot of trends observed in the data and in real life and are biased themselves. For example, budgets of movies released a few decades ago were lower than latest movies as the values are not inflation adjusted.

Solution

Supplement the dataset with new predictors that can identify patterns like the one above

* Most ‘intuitively’ useful predictors had to be dropped: predictors like *cinematographer, distributor etc.* had to be dropped from analysis due to high cardinality and collinearity but are intuitively powerful predictors

Solution

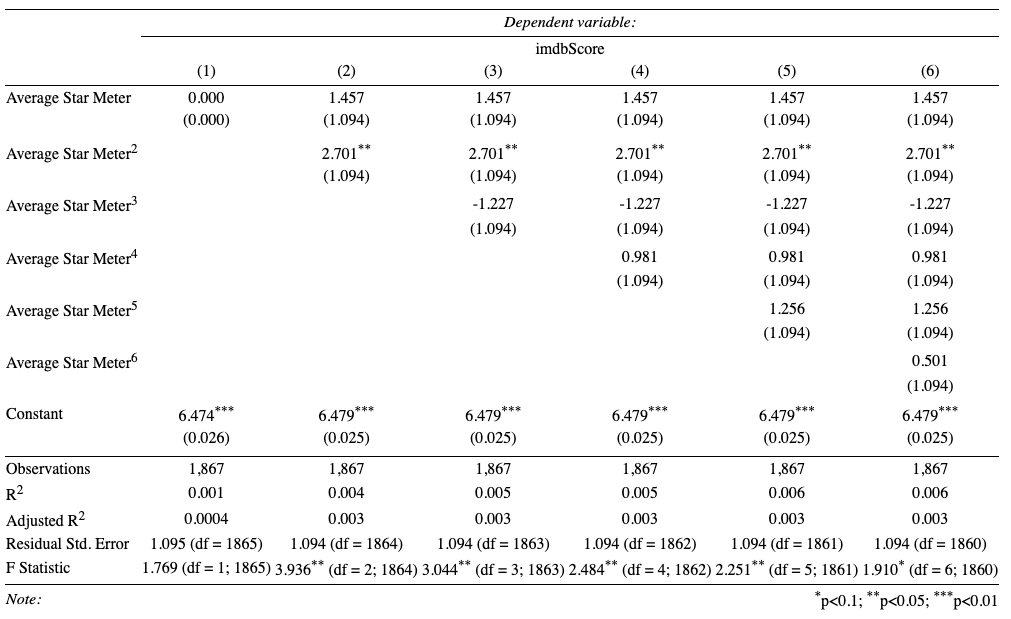
Replace/engineer these predictors with alternatives like cinematographer rank or number of distributors that can be used to enhance the predictive power of the model

**Section 5: Appendices**

**5.1 Tables**

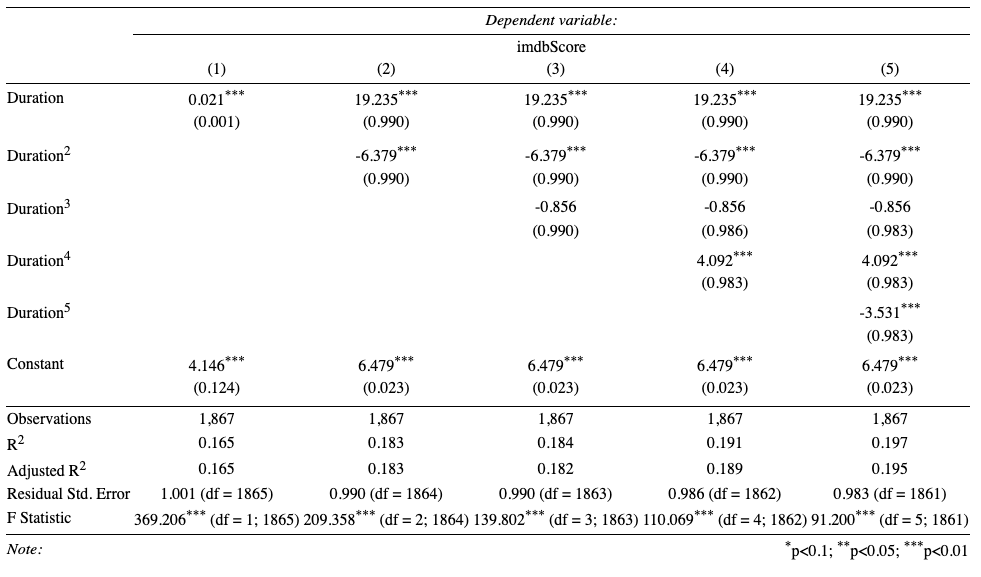
**Table 1a.** Polynomial Regression: *imdbScore* vs. *avgStarMeter*

Adjusted R2 = 0.004 of degree 1 was reported to be the highest and selected.



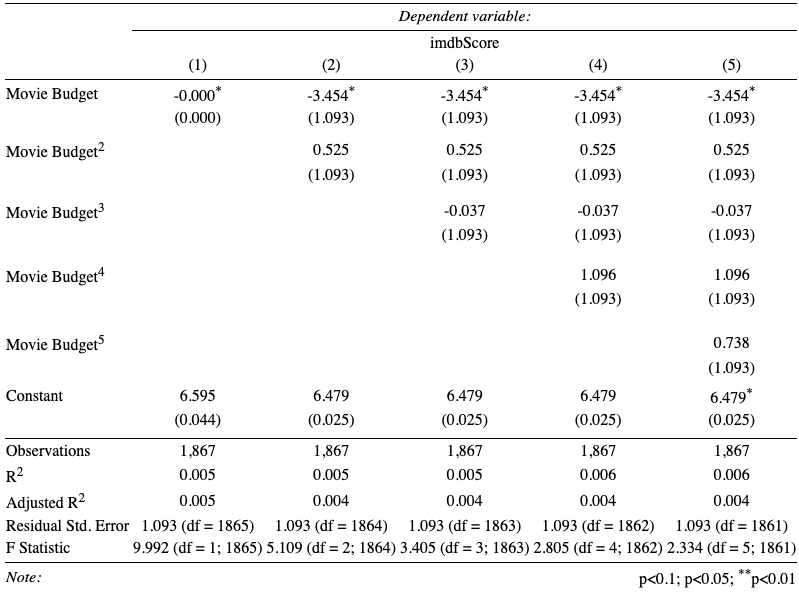
**Table 1b.** Polynomial Regression: *imdbScore* vs. *duration*

Adjusted R2  = 0.183 of degree 2 was reported to be the highest and therefore selected



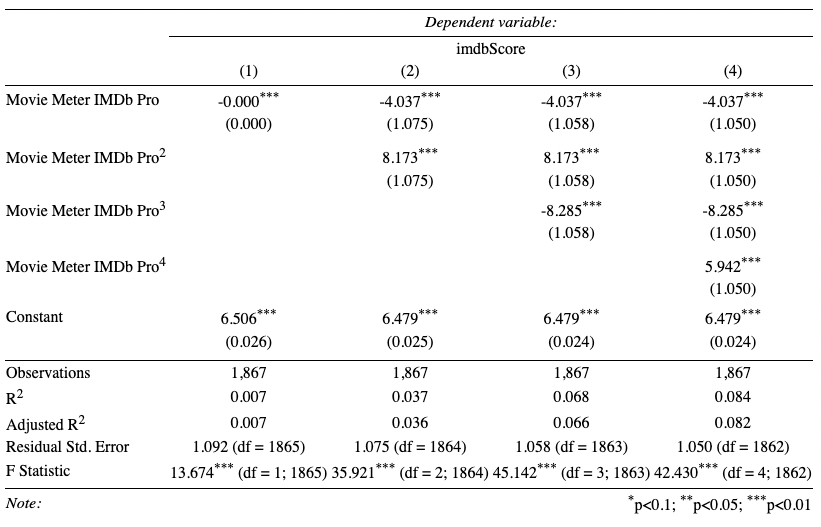
**Table 1c.** Polynomial Regression: *imdbScore* vs. *movieBudget*

Adjusted R2  = 0.005 of degree 1 was reported to be the highest and therefore selected



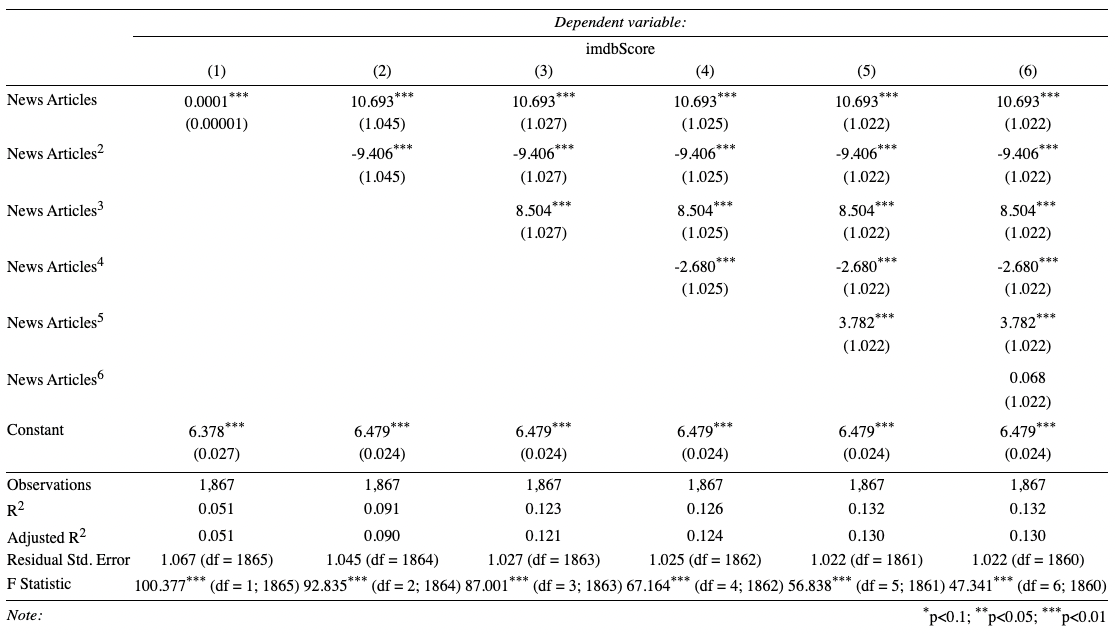
**Table 1d.** Polynomial Regression: *imdbScore* vs. *movieMeter\_IMDBpro*

Adjusted R2  = 0.066 of degree 4 was reported to be the highest but the model was trained based on lower degrees to prevent overfitting



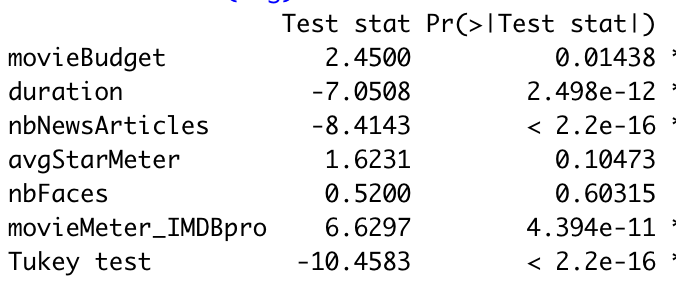
**Table 1e.** Polynomial Regression: *imdbScore* vs. *nbNewsArticles*

Adjusted R2  = 0.005 of degree 3 was reported to be of highest improvement compared to degree 2 and therefore selected. Note that models were also trained for higher models.



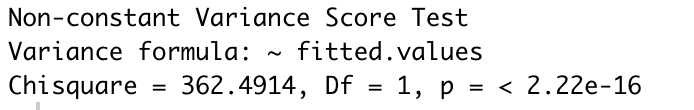
**Table 2.** Non-linearity Check

Since p-values of *nbFaces* and *avgStarMeter* were > 0.1, they were the only ones considered to be linear.

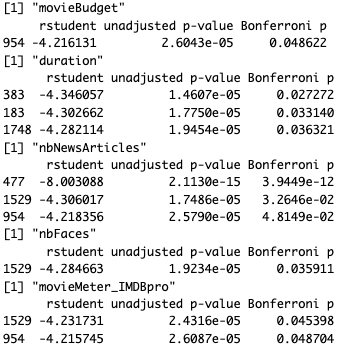


**Table 3.** Heteroskedasticity Check

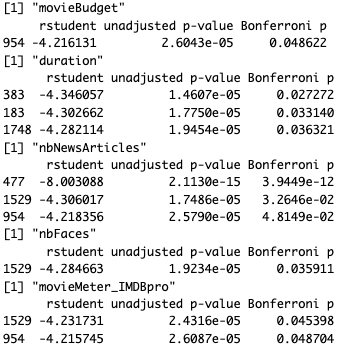
Since p < 0.05, the heteroskedasticity issue exists. Thus, coeftest was used to eliminate it.



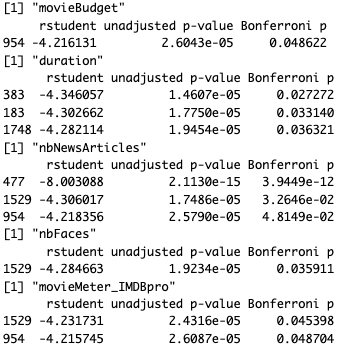
**Table 4a.** Outliers Check: *movieBudget*



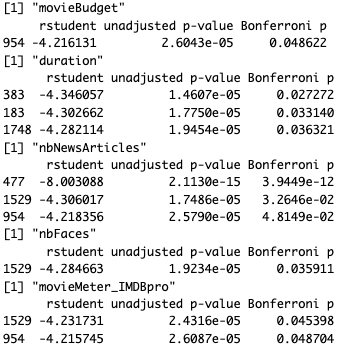
**Table 4b.** Outliers Check: *duration*



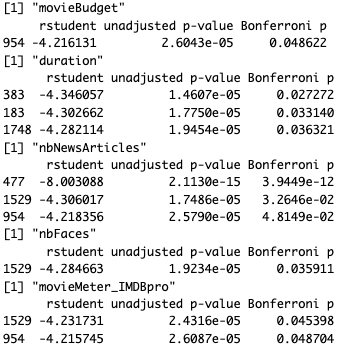
**Table 4c.** Outliers Check: *nbNewsArticles*



**Table 4d.** Outliers Check: *nbFaces*

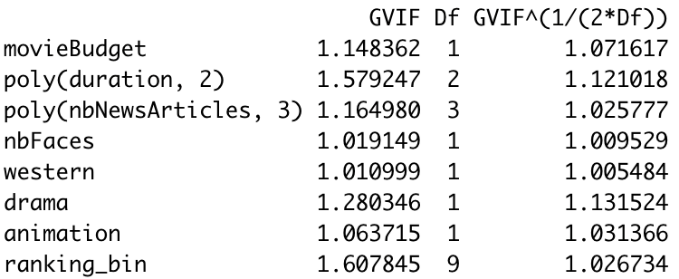


**Table 4e.** Outliers Check: *movieMeter\_IMDBpro*



**Table 5.** Collinearity Check

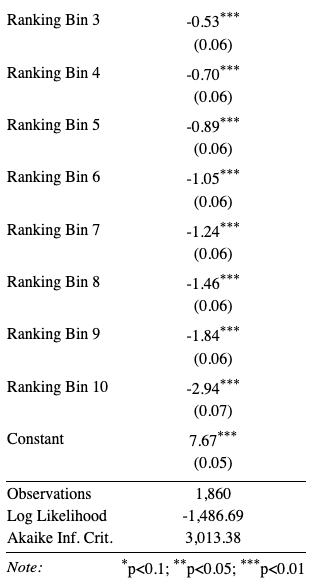
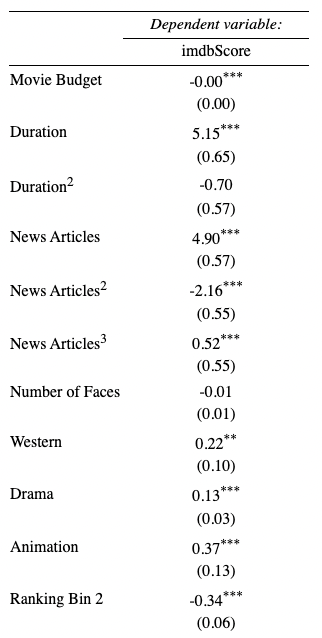
Since all values < 4, no collinearity issue was detected between predictors.

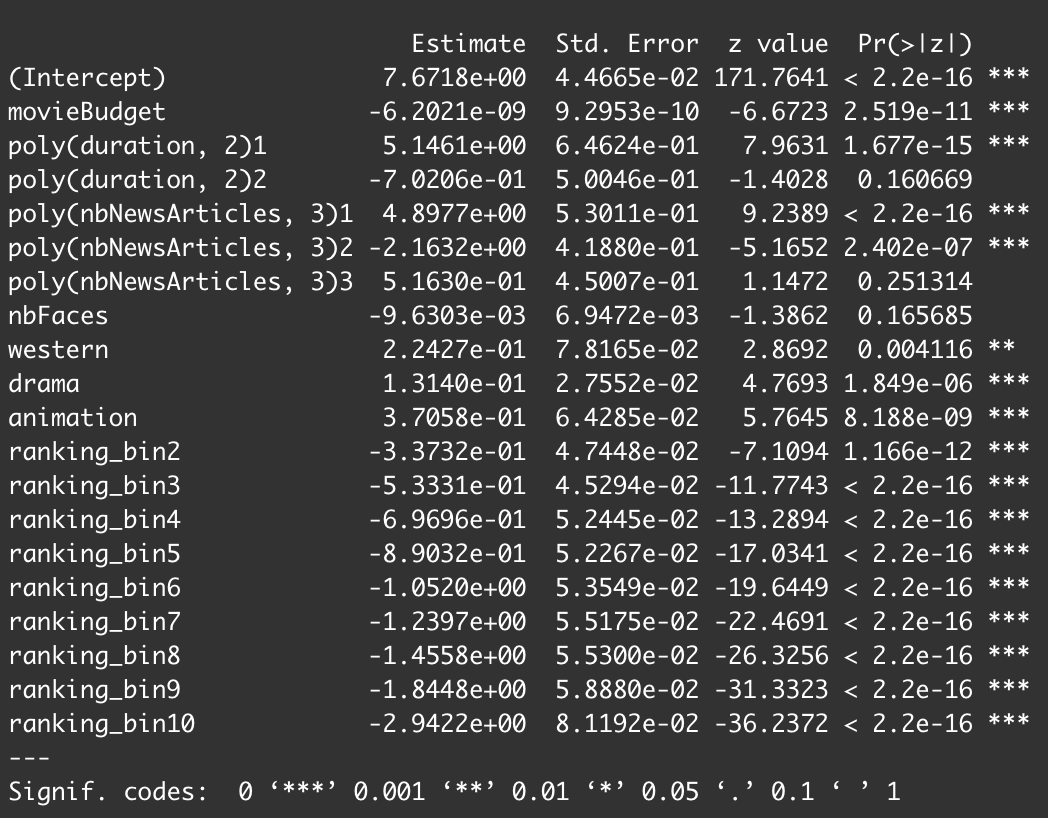


**Table 6.** Model Comparison

|  |  |  |
| --- | --- | --- |
| Model | Multiple R2 | Adjusted R2 |
| **model1** = lm(imdbScore~ movieBudget +poly(duration, 2) + maturityRating +poly(nbNewsArticles,2) +nbFaces + action + adventure+scifi +thriller + musical + romance + western +sport +horror + drama + war + animation + crime +ranking\_bin + poly(movieMeter\_IMDBpro,2) + avgStarMeter) | 0.7536 | 0.7491 |
| **model4** = lm(imdbScore~ movieBudget +poly(duration, 2) + maturityRating +poly(nbNewsArticles,4) +nbFaces + action + adventure+scifi +thriller + musical + romance + western +sport +horror + drama + war + animation + crime +ranking\_bin + poly(avgStarMeter,3)) | 0.7517 | 0.7468 |
| **model3** = lm(imdbScore~ movieBudget +poly(duration, 2) + maturityRating +poly(nbNewsArticles,4) +nbFaces + action + adventure+scifi +thriller + musical + romance + western +sport +horror + drama + war + animation + crime +ranking\_bin + poly(movieMeter\_IMDBpro) + poly(avgStarMeter,3)) | 0.7517 | 0.7466 |
| **model2** = lm(imdbScore~ movieBudget +poly(duration, 3) + maturityRating +poly(nbNewsArticles,3) +nbFaces + action + adventure+scifi +thriller + musical + romance + western +sport +horror + drama + war + animation + crime +ranking\_bin + movieMeter\_IMDBpro+poly(avgStarMeter,2)) | 0.7513 | 0.7464 |
| **model8** = lm(imdbScore~ movieBudget +poly(duration, 2) +poly(nbNewsArticles,3) +nbFaces + action +scifi +thriller + musical + romance + western +sport +horror + drama + war + animation +  crime +ranking\_bin) | 0.7494 | 0.7456 |
| **model9** = lm(imdbScore~ movieBudget +poly(duration, 2) + poly(nbNewsArticles,3) +nbFaces + action +scifi +thriller + romance + western +sport +horror + drama + war + animation + crime + ranking\_bin) | 0.7493 | 0.7456 |
| **model11** = lm(imdbScore~ movieBudget +poly(duration, 2) +poly(nbNewsArticles,3) +nbFaces + western + drama + animation + ranking\_bin) | 0.7481 | **0.7455** |
| **model7** = lm(imdbScore~ movieBudget +poly(duration, 2) +poly(nbNewsArticles,3) +nbFaces + action + adventure+scifi +thriller + musical + romance + western +sport +horror + drama + war + animation + crime +ranking\_bin) | 0.7494 | 0.7454 |
| **model5** = lm(imdbScore~ movieBudget +poly(duration, 3) +poly(nbNewsArticles,3) +nbFaces + action + adventure+scifi +thriller + musical + romance + western +sport +horror + drama + war + animation + crime +ranking\_bin + poly(avgStarMeter,2)) | 0.7497 | 0.7453 |
| **model10** = lm(imdbScore~ movieBudget +poly(duration, 2) +poly(nbNewsArticles,3) +nbFaces + action +scifi +thriller + romance + western +sport +horror + drama + war + animation + crime) | 0.3757 | 0.3696 |

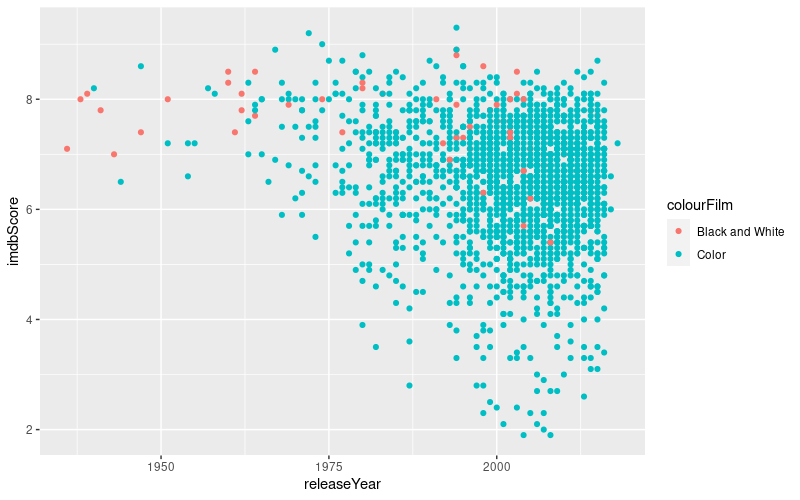
**Table 7a.** Final Model Coefficients



**Table 7b.** Final Model Predictor Significance

**5.2 Plots**

**Plot 1.** The relationship between black and white and colour films.



**Plot 2.** Non-linearity Check

