Hold My Popcorn: Predicting a Film's Popularity with Linear, Logistic and Principal Analyses

Leyla Akay and Maryann Zhao

Introduction:

In this project, we were interested in determining whether we can predict how "popular" a film is, given relevant information about it. Although "popularity" is inherently difficult to measure, we relied upon quantitative scores taken from The Movie Database (https://www.kaggle.com/tmdb-movie-metadata/data), based on online votes, clicks, and social media 'likes'. To predict these popularity values, we utilized the variables of budget, revenue, average critic's vote, the amount of times the movie was voted upon, primary spoken language, genre, and whether the movie was made before or after 2007. We chose to investigate the year 2007 as that was when the iPhone was introduced, and subsequent cultural shifts in media consumption have been linked to it (Twenge, 2017). All information used in this study was sourced from The Movie Database. The movies included in this database spanned genres including Comedy, Drama, and Horror, and used a variety of languages, from English to Russian. In total, we used data from over four thousand films to create our model.

Before we began analyzing our data, we modified 5 variables to be binary and filtered for values that were not meaningful. First, we asked whether the primary language spoken in the film was English, coding English to 1. Similarly, we converted movies released 2007 or later and the genres action, adventure and drama to binary variables. To clean our data, we chose to filter certain explanatory variables (budget, revenue, number of votes) to remove zero values when such a value would not be meaningful. For instance, a movie that has a budget of zero in reality has little meaning in the sense that films require at least a small sum of money in order to be able to obtain the necessary actors, production material, props, etc.

Previously, we performed nested F tests to narrow down the significant variables included in our multiple linear regression model described above. This model will start as our baseline model that we will use for comparisons in our current study.

```
#Converting if the primary language was English into binary (En = 1)
lang <- c()
for (i in 1:4803){
 lang[i]<- ifelse(movies$original_language[i] == "en", 1, 0)
}

#Converting if release date was after 2007 into binary (2007 or later = 1)
dates <- as.Date(movies$release_date, "%m/%d/%y")
dates <- as.numeric(substring(dates, 1, 4))
movies <- movies %>%
  mutate(years = dates)
oh7 <- c()
for (i in 1:4803){
 oh7[i]<- ifelse(movies$years[i] >= 2007, 1, 0)
}

#Convert if first 2 genres are action, adventure or drama based on most popular genres (website) (genre
action <- c()
adv <- c()
drama <- c()
for (i in 1:4803){
 action[i] <- ifelse(movies$genres1[i] == "Action" | movies$genres2[i] == "Action", 1, 0)
```

```
 action[i] <- ifelse(is.nan(action[i]) | is.na(action[i]), 0, action[i])
 adv[i] <- ifelse(movies$genres1[i] == "Adventure" | movies$genres2[i] == "Adventure", 1, 0)
 adv[i] <- ifelse(is.nan(adv[i]) | is.na(adv[i]), 0, adv[i])
 drama[i] <- ifelse(movies$genres1[i] == "Drama" | movies$genres2[i] == "Drama", 1, 0)
 drama[i] <- ifelse(is.nan(drama[i]) | is.na(drama[i]), 0, drama[i])
}

#Remove values where budget, revenue, or vote count is 0 because not meaningful
movies.zero <- movies %>%
  select(popularity, budget, revenue, vote_average, vote_count, oh7, adv, action, drama, lang) %>%
  filter(budget > 0, revenue > 0, vote_count > 0)

#Full MLR with log transformations
full.lm <- lm(log(popularity) ~ log(revenue) + vote_average + log(budget) + log(vote_count) + oh7 + adv
anova(full.lm)
```

```
## Analysis of Variance Table
##
## Response: log(popularity)
##                            Df Sum Sq Mean Sq F value Pr(>F)
## log(revenue)                1   1530    1530 8183.44 <2e-16 ***
## vote_average                1    221     221 1181.85 <2e-16 ***
## log(budget)                 1     26      26  140.43 <2e-16 ***
## log(vote_count)             1   1523    1523 8142.86 <2e-16 ***
## oh7                         1      1       1    3.78  0.052 .
## adv                         1      0       0    0.11  0.736
## action                      1      0       0    0.53  0.469
## drama                       1      0       0    2.37  0.124
## lang                        1      0       0    0.01  0.907
## log(revenue):log(budget)    1      5       5   28.93  8e-08 ***
## Residuals                3216    601       0
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Shrinking The Variables

Our first goal was to reduce the variability associated with our coefficients. We did this by using two different methods: "Ridge Regression", which minimizes the coefficients, and "Lasso", which shrinks the coefficients to zero. Unlike Ordinary Least Squares, both Ridge Regression and Lasso add a little bit of bias into the model, such that coefficients that explain the data well are kept, and unimportant coefficients are shrunk. This helps avoid overfitting the model to the random quirks of the data. To determine the best tuning parameter to use for these methods, we performed 10-fold cross-validation. The values of the tuning parameter that minimized the mean square error for Lasso and RR were 0.01 and 0.0118, respectively. We see that as we increase the tuning parameter, lambda, the variable of how many votes a movie received is the slowest to be shrunk to zero. This indicates that this variable may be important for our model's predictions.

The original coefficients produced by our multiple linear regression model were (0.0077) for language; (0.0005) for Action; (0.016) for Adventure; and (0.29) for Drama. The coefficients of Adventure and Drama were shrunk by Ridge Regression to values of (0.013) and (0.031), respectively, and were reduced to zero by Lasso. In contrast, the coefficients of language and Action increased to (0.018) and (0.013) under Ridge Regression. We conclude that the variables shrunk by Ridge Regression and Lasso are less efficient in reducing variability in our model, whereas those kept or increased by Ridge Regression are more efficient.

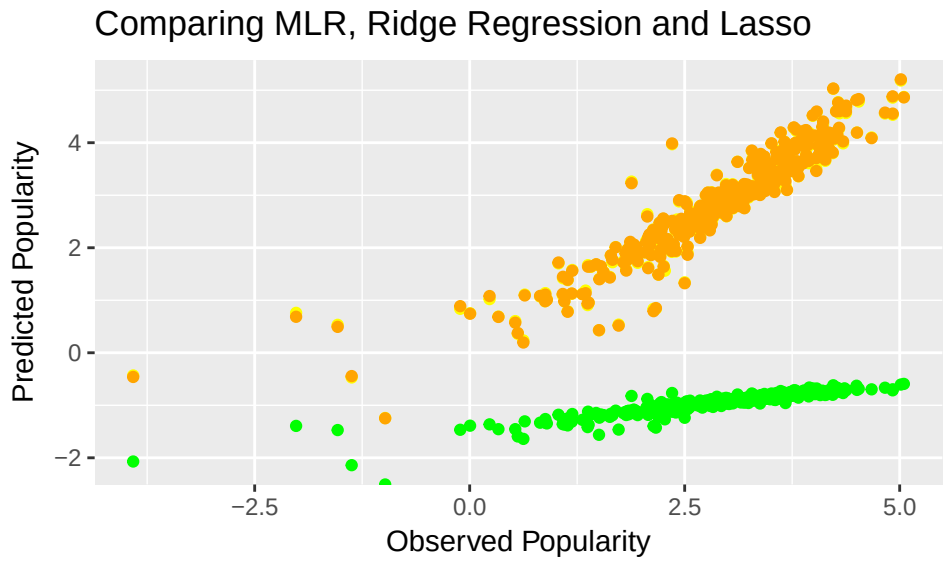

Figure 1. Comparison of MLR, Ridge Regression and Lasso models. Ridge Regression (yellow) and Lasso (orange) display a comparable comparable relationship between observed popularirty and predicted popularity where both predict similar values as the observed. MLR (green) on the otherhand appears to underpredict the popularity compared to the observed popularity.

From the plots, it is clear that using Ridge Regression or Lasso predicts our data in a more representative manner. The data points represent how well the predicted popularity correlates with the observed popularity in the test set. If the predicted is accurate in predicting the observed, we would expect a slope of around 1 and that is what we observe for Ridge Regression and Lasso but not MLR.

Smoothing

We were next interested in smoothing the variable of vote counts, as it had the most significant p-value for the nested F test and the largest estimated coefficient in the model. We used both kernel and spline smoothing methods, each time changing the degrees of freedom and span, respectively. With larger degrees of freedom and smaller span, the model became more flexible–but approached the dangerous territory of overfitting.

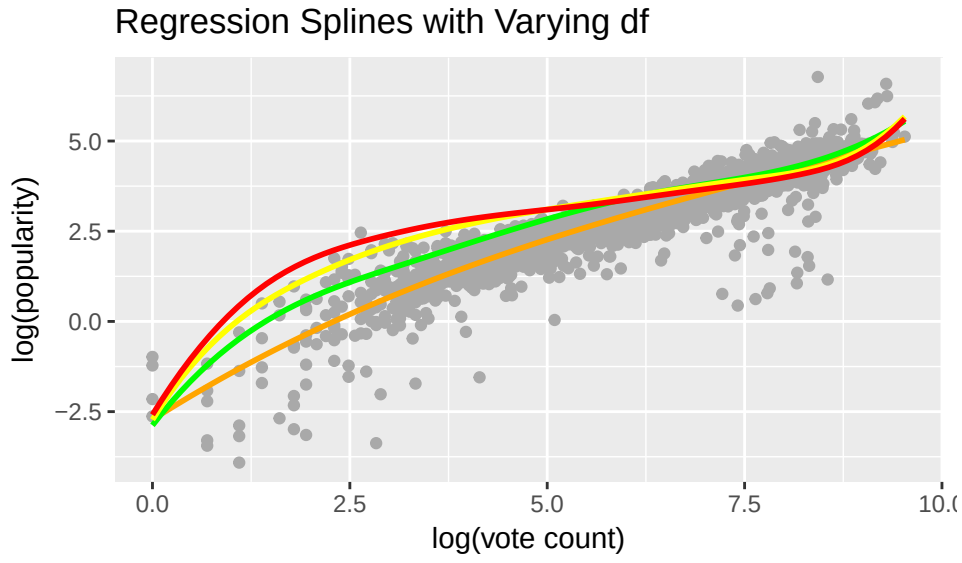

Figure 2. Regression spline model based on vote count. The explanatory variable vote count was used to create a regression spline model using four different degrees of freedom: 3 (orange), 5 (green), 7 (yellow), 9 (red).

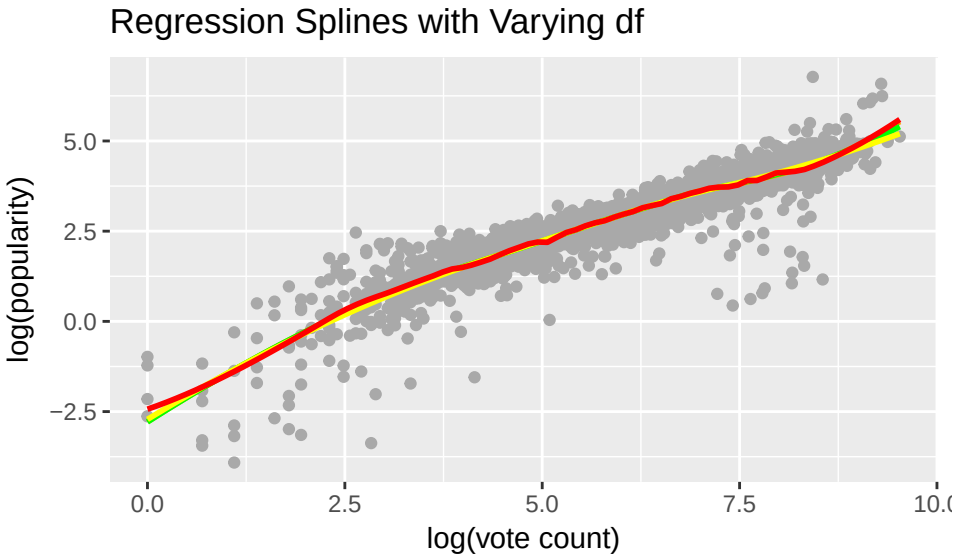

Figure 3. LOESS model based on vote count. Based on the explanatory variable vote count, LOESS models were built using four different span values: 0.1 (red), 0.3 (orange), 0.5 (green), 0.7 (yellow).

We conclude that the best model is one that accurately describes the trends in the data, without overfitting it. Such a model of regression splines has three degrees of freedom (the orange line) in this case because the relationship between the log vote count and log popularity appears to be linear based off the distribution of the data. This line captures the overall trend without being swayed by the increased variability on both ends. All four parameters assessed in the LOESS models appear to be relatively similar, but the line with the smallest span (in red) seems to overfit the data. It follows the individual variations in the data, obscuring the overall linear relationship between the two variables and making the fit less smooth.

Principal Component Analysis

We used many parameters to describe our data, some of which could conceivably be correlated. For example, budget is likely related to genre; movies requiring computer-generated graphics probably cost more to produce than those without, like comedies. Therefore, we were interested in using Principal Component Analysis to reduce the dimensionality of our data. If many parameters are correlated, then theoretically we could only use one to explain the variation in the data. The assumptions for PCA rest on the fact that there's a linear relationship between variables, which we had previously established in our last study by producing the variables' correlation matrix. To perform PCA, we therefore first calculated a correlation matrix of the continuous variables (revenue, budget, vote count, vote average), and used the eigenvectors as the principal components of our data's variation. We chose to only look at continous variables because their co-variance would be more meaningful than binary variables. We used the first two principal components, which together accounted for 84% of the variation within our data, as axes on which to transform each data point.

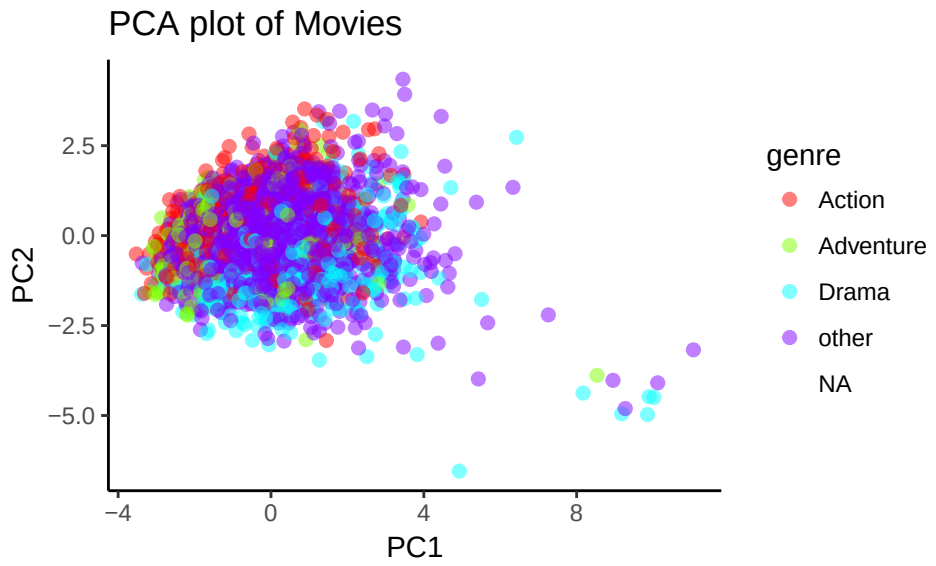

Figure 4. PCA analysis of continuous variables. The PCA analysis is visualized using the two principle components PC1 and PC2. The data is categorzied based on whether the genre is action (red), adventure (green), drama (blue), or other (purple).

We were curious if plotting the data according to these principal components would reveal differences in variation between genres. We therefore color-coded individual movies according to whether they were an Action, Adventure, Drama, or "other" kind of film. There appeared to be no difference in variation relevant to genre; the colors are evenly dispersed amongst the axes. We conclude that the two principal components do not account for the variation between genre.

Logistic regression

Finally, we were interested in exploring the concept of Action movies. There has been an increase in recent years of high-profile films like the Marvel superhero movies, which are commonly considered "Action" movies. As the concept of genre is itself rather nebulous, we were curious if there is something inherent in the particular combination of a movie's popularity, budget, etc., that leads it to being described as an Action film.

To test the idea that we can predict whether a film is labeled as an Action movie or not from its popularity, budget, vote scores and average, spoken language, and whether it was produced before or after 2007 (the same variables previously used), we made use of a logistic regression model. Logistic regression models are similar to linear regression models in that they utilize different explanatory parameters to predict a response, but differ in that the response is binomial. Logistic regressions assume that the observations are independent, not multicollinear, and the sample size is large, as in our model. In this case, the response was whether a movie belonged to the genre of 'Action' or not. We included all the variables of the previous model, and performed a nested Chi-square test evaluating the sequential significance of each parameter. With a chi-square value of 296, 7 degrees of freedom, and an associated p-value less than 5e-60, the parameters fit the data significantly better than the null model, so we kept all of them.

```
#Logistic Regression Model
lr2 <- glm (action ~ pop.log+ rev.log + vote_average + bud.log + count.log + oh7 + lang, data = movies.l
```

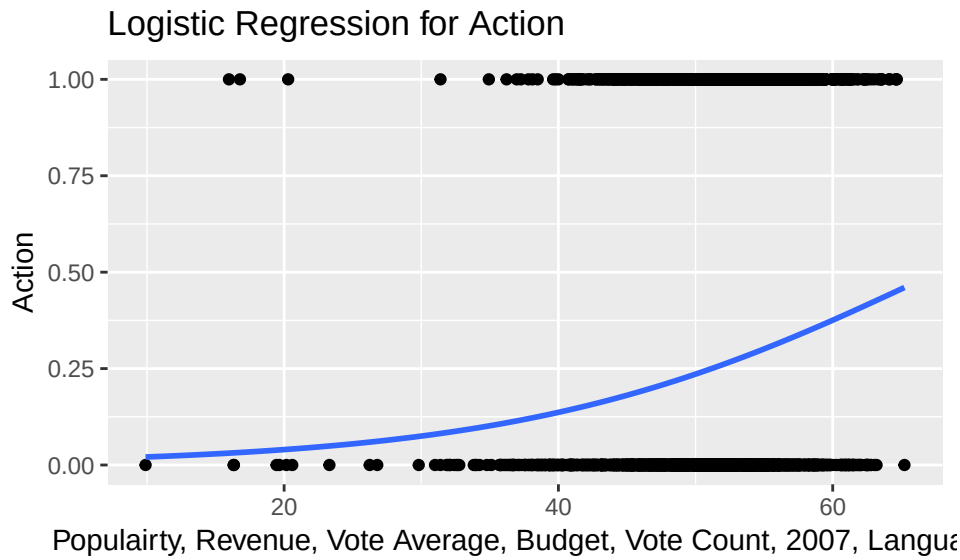

Figure 5. Logistic Regression for Action. Using the seven variables that were determined to be significant through nested Chi-square tests, a logstic regression model was fit to model Action.

Our logistic regression then used these parameters to calculate the odds associated with predicting a movie's genre as 'Action' or not. This revealed some interesting findings. A doubling of budget, holding all other variables constant, was associated with a 0.42 increase in odds that a movie would be classified as Action. This may not be surprising; action movies tend to demand expensive equipment and high-profile actors. Similarly, a doubling in the amount of votes cast on a movie was associated with a 0.50 increase in odds of classifying the movie as Action. Interestingly, a doubling of the actual vote score was associated with a 0.57 decrease in the odds of our model classifying the movie as Action. One interpretation of these findings is that people may be more likely to vote on action movies, but do not necessarily award them higher scores than other types of movies.

Conclusion

Our aim with this project was to explore the limits of predicting a movie's popularity, given financial, social, and temporal data about its production. We found that the greatest predictor variable, with a coefficient estimate of 0.67, was the "vote count", or sheer number of times the movie had been voted upon. Interestingly, the average vote score parameter was shrunk by Ridge Regression and Lasso, suggesting that it is not as efficient in explaining the variation in our data. One possible explanation of these seemingly confounding results is that the average vote is taken from critics, while the popularity score is derived from anybody interacting with the movie online. Perhaps movie critics simply do not agree with popular taste.

Another very interesting result our analysis revealed was that the parameter of whether a movie was produced before or after the year 2007 was significant in predicting its popularity. Since the introduction of the iPhone in 2007, teenager's anxiety levels have spiked, and their rates of socializing outside the house have dropped. As popularity scores are primarily determined from online activity, we wondered whether movies produced in the post-iPhone era might be affected by this decline in physical, and rise in digital, social activity. Indeed, the parameter's coefficient actually grew following Ridge Regression, meaning that a movie's being produced after 2007 is associated with a 0.03 increase in the movie's median popularity.

References

Twenge, J. M. (2017, September). Have Smartphones Destroyed a Generation? The Atlantic. Retrieved from https://www.theatlantic.com/magazine/archive/2017/09/has-the-smartphone-destroyed-a-generation/

534198/