Final Project

Christina Yu, Damian Kim

Read in the data

```
library(tidyverse)
library(ggfortify)
library(knitr)
library(broom)
library(patchwork)
library(tidymodels)
library(corrplot)
library(nnet)
library(car)
library(mice)
library(naniar)
library(UpSetR)
data <- read_csv("data/movies.csv")
```

Introduction and data

With the rise in the age of information, the processing speeds of data are accelerated to impossible levels. One of the densest forms of media content out there is the transcription of words into pixels. A book that previously took 2 weeks to finish now fits in the span of 2 hours in the insanely digestible form of: movies. The movie industry is worth 95 billion dollars in the US alone, and indeed there is a great demand for quality movies. Though it is not easily apparent to tell what makes a movie great, there are some basic data that might help determine some of the factors of a great movie. We decided to try and tackle this heavy yet essential question. What factors might help make a movie successful? Throughout this project we can consider this question through the lens of a potential movie investor.

This dataset was scraped from IMDb (Internet Movie Database). There are 6820 movies in the dataset (220 movies per year, 1986-2016). Each movie has the following attributes:

– `budget`: the budget of a movie. Some movies don't have this, so it appears as 0

– `company`: the production company

– `director`: the director

– `genre`: main genre of the movie.

– `gross`: revenue of the movie

– `name`: name of the movie

– `rating`: rating of the movie (R, PG, etc.)

– `released`: release date (YYYY-MM-DD)

– `runtime`: duration of the movie

– `score`: IMDb user rating

– `votes`: number of user votes

– `star`: main actor/actress

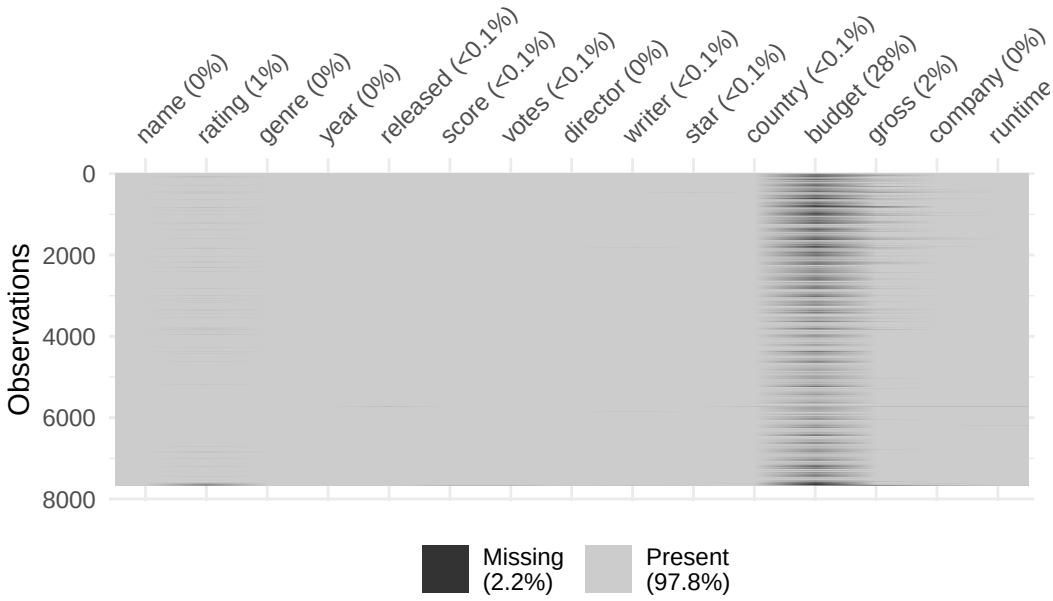
– `writer`: writer of the movie

– `year`: year of release

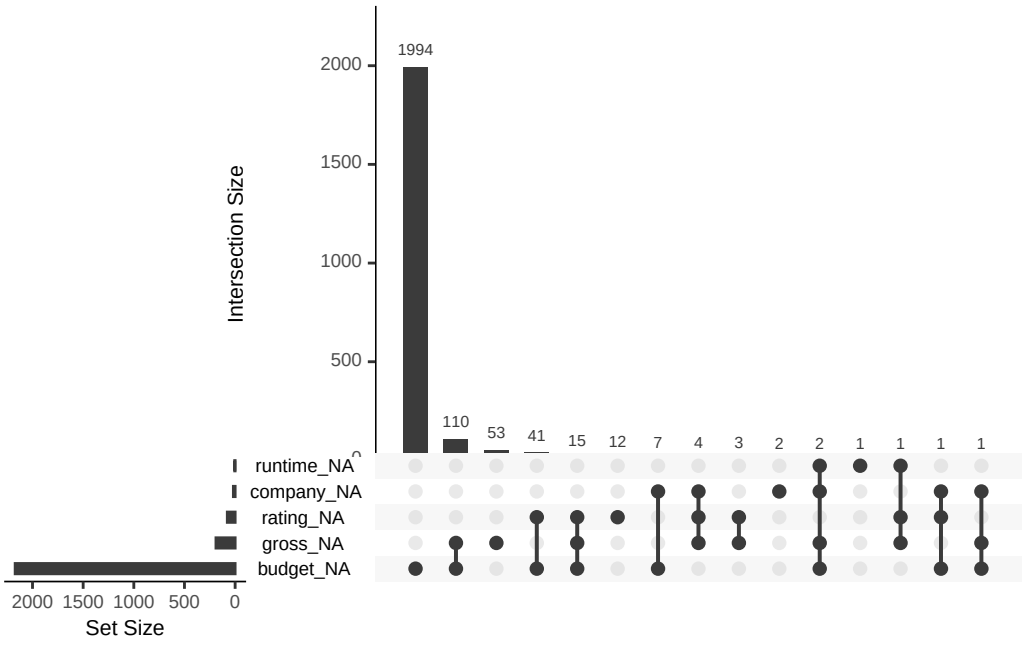
We will explore the factors that make a movie successful through examining the effects of our variaables of interest: `gross`, `budget`, `genre`, `rating`,`score` `votes`, and `runtime` for individual movie.

First we will explore missingness in our dataset.

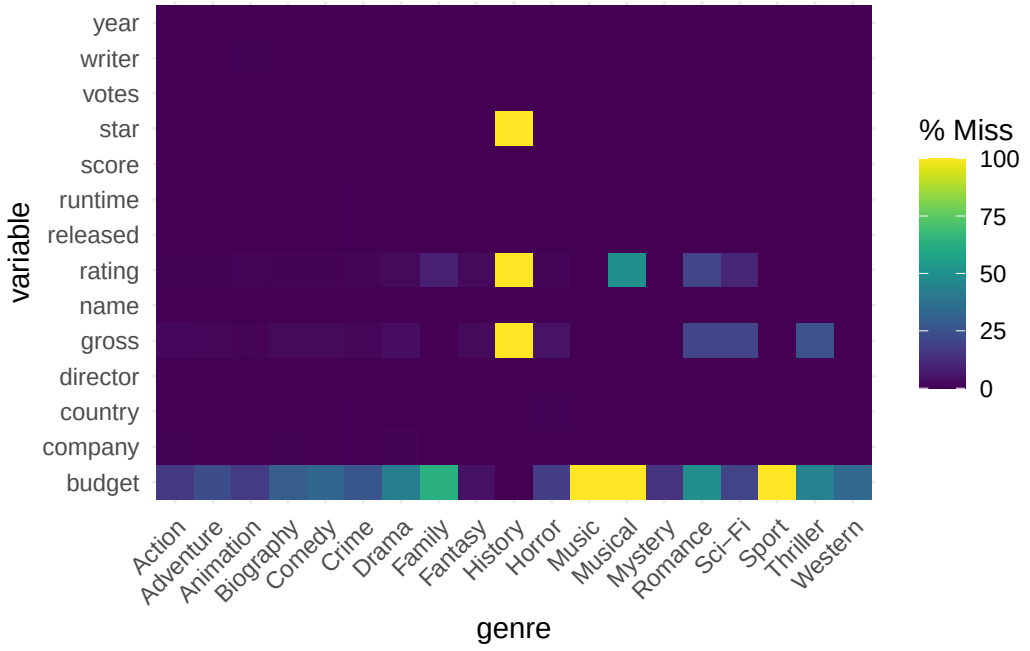
```
vis_miss(data)
```




```
gg_miss_upset(data)
```



```
gg_miss_fct(x = data, fct = genre)
```



The dataset includes movies that haven't been published yet (which causes a missing gross value). In our project we are considering gross revenue of a movie as the defining indicator of success. Considering that, we removed all observations that haven't been published yet or are missing gross values, because we are interested only in movies with a quantitative measure of success.

In fact, we also decided to rid our dataset of missing values for the other predictor variables as well. About 28% of the observations were missing a value for budget, whereas the next most missing variable was gross, with 2%. From elementary missingness analysis there did not seem to be very strong, numerically significant relationships between the missingness of each variable with other variables of interest. Though it is unlikely that the missingness of our budget data in particular was MCAR (generally unlikely in the real world, perhaps lower budgets were less public and thus less likely to have solid budget details), we decided to do a complete case analysis. From our data it was hard to tell if the dataset could be assumed to be MAR. Though there might be some bias introduced, since we were working with such a large number of movies, we decided it was okay to lose some validity of our model. Obviously this is a big limitation of our project, and if such a project were done again we would be more meticulous. Now we have 5423 observations in the dataset.

```
data <- data %>%
  filter(!is.na(gross) & !is.na(budget) & !is.na(genre) & !is.na(rating) & !is.na(score) &
```

The Predictor Variables

We will use the `budget`, `genre`, `rating`,`score votes`, `runtime`, and `country` variables as predictors. Among them, `budget`, `score votes`, and `runtime` are numerical variables, while `genre`, `rating`, and `country` are categorical variables.

The Response Variable

As described earlier, the `gross` numerical variable is our response variable. 1. Summary of the `gross` variable:

```
data %>%
  summarise(mean_gross = mean(gross),
            median_gross = median(gross),
            sd_gross = sd(gross),
            min_gross = min(gross),
            max_gross = max(gross)) %>%
  kable()
```

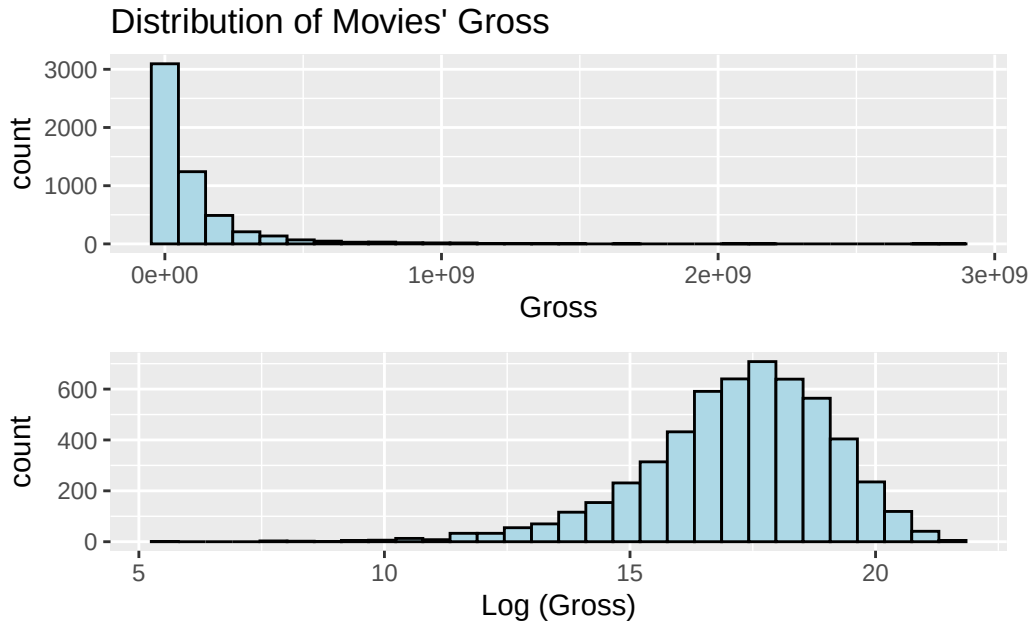
mean_gross	median_gross	sd_gross	min_gross	max_gross
103192280	36850101	187278279	309	2847246203

2. Log-transformation and Distribution of the `gross` variable:

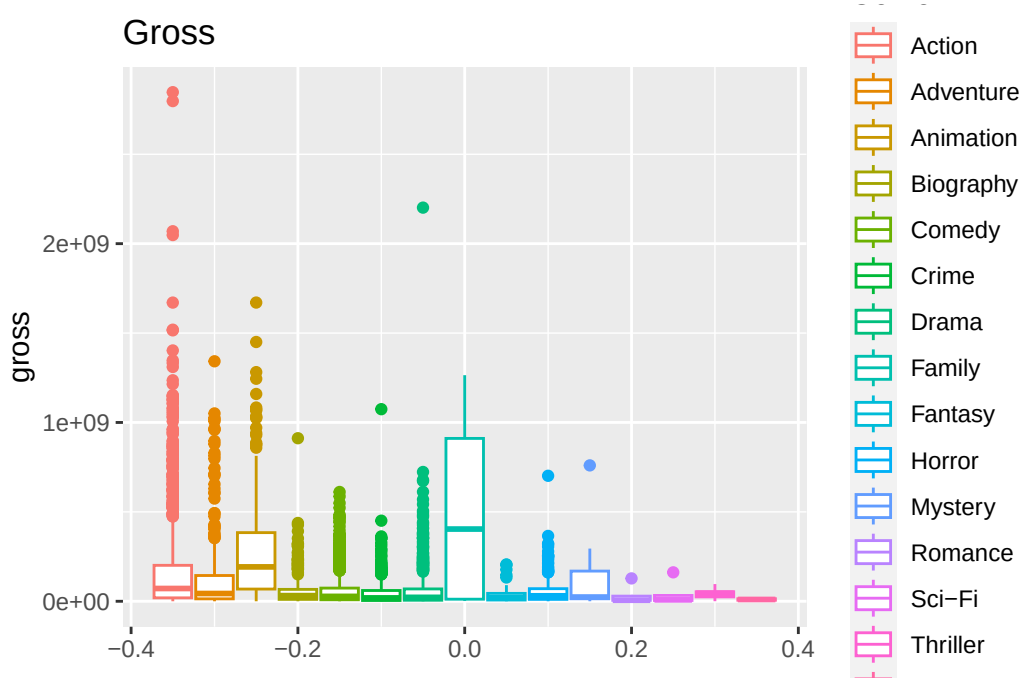
```
data <- data %>%
  mutate(log_gross = log(gross)) %>%
  mutate(mean = mean(log_gross))

p1 <- ggplot(data = data, aes(x = gross))+
  geom_histogram(fill = "light blue", color = "black")+
  labs(title = "Distribution of Movies' Gross",
       x = "Gross")
p2 <- ggplot(data = data, aes(x = log_gross))+
  geom_histogram(fill = "light blue", color = "black")+
  labs(x = "Log (Gross)")
p1/p2
```

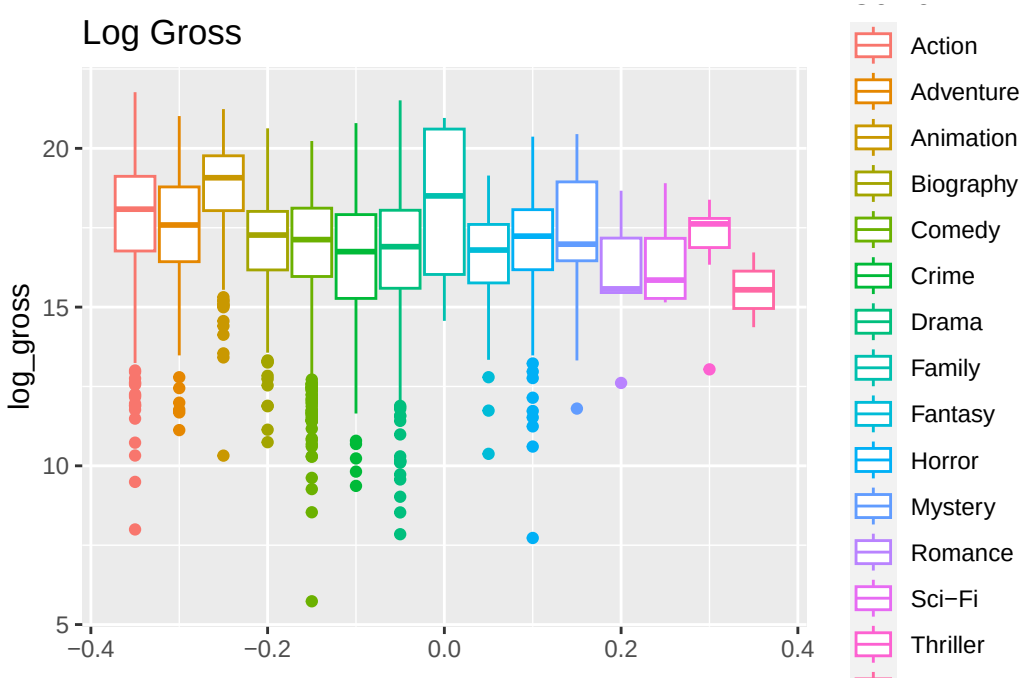
```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
ggplot(data = data) +
  geom_boxplot(aes(y = gross, color = genre)) +
  labs(title = "Gross",
       color = "Genre")
```



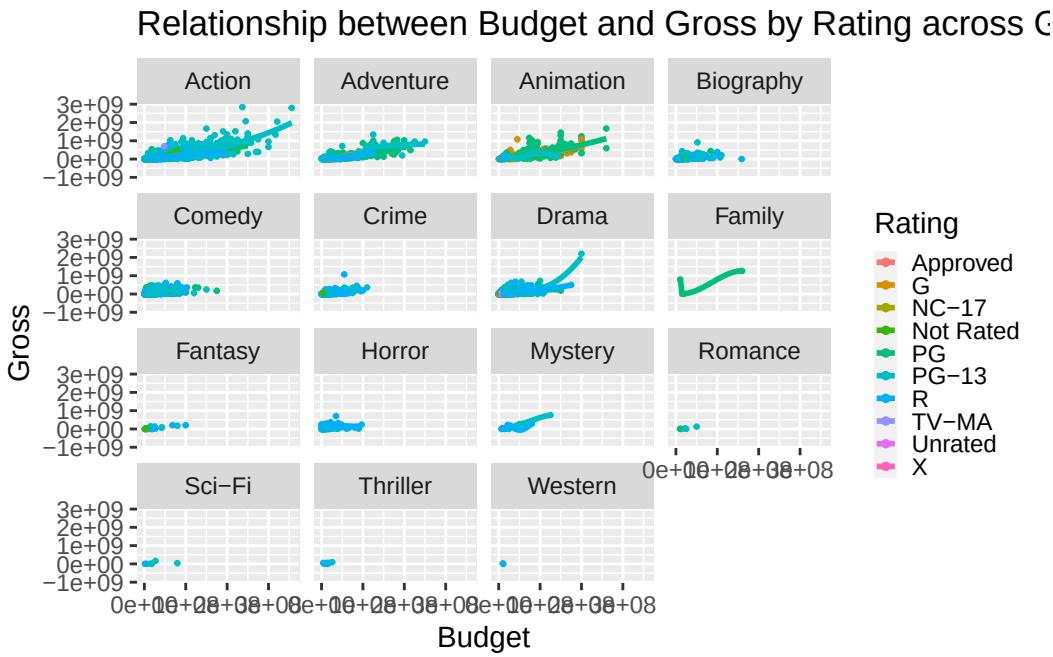

```
ggplot(data = data) +
  geom_boxplot(aes(y = log_gross, color = genre)) +
  labs(title = "Log Gross",
       color = "Genre")
```



Since the response variable is significantly right skewed, we apply a log-transformation to it and will use log(gross) as our new response variable in the future analysis. Now, our response variable is unimodal, following a roughly normal distribution, with a mean at 17.2102, and there exist some outliers on the left end.

3. Relationship between Gross and Budget based on different Genres & Ratings:

```
ggplot(data = data, aes(x=budget, y = gross, color = rating))+
  geom_point(size=0.5, fill=NA) +
  geom_smooth(fill=NA) +
  theme(legend.key.size = unit(0.3, "cm")) +
  facet_wrap(~ genre)+
  ggtitle("Relationship between Budget and Gross by Rating across Genres") +
  xlab("Budget") +
  ylab("Gross")+
  scale_color_discrete(name = "Rating", guide = guide_legend(override.aes = list(size = 1)
  theme(panel.spacing.x = unit(2, "mm"))
```

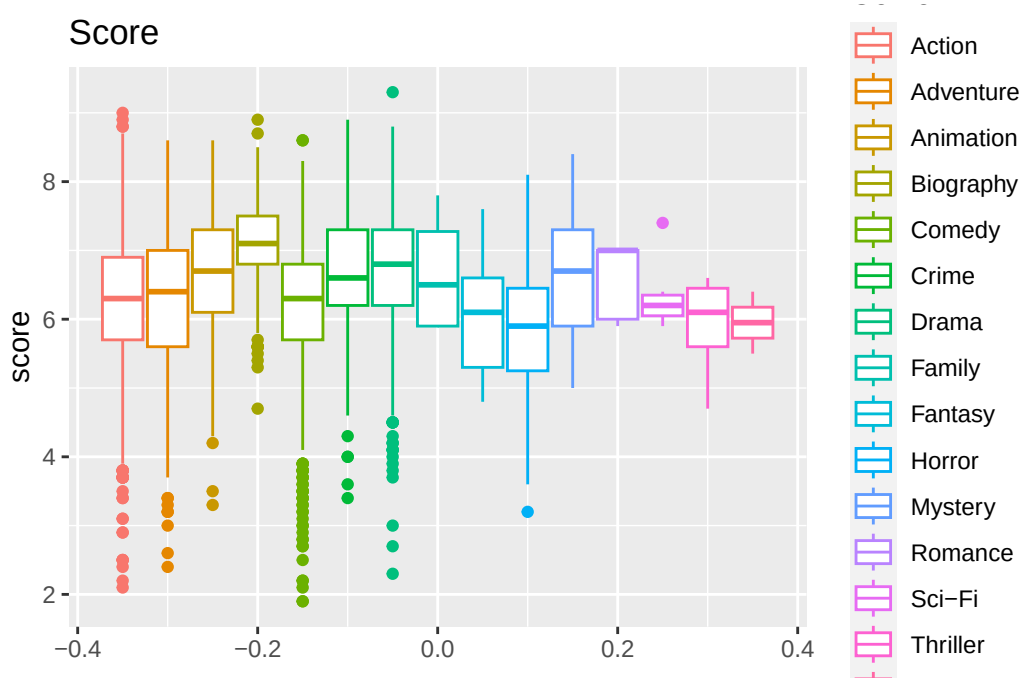


We observe that the relationship between budget and gross is vastly different across genres: action, adventure, animation movies have a steep slope and generally high budget ranges, with outliers which have exceedingly high budget and relatively high gross values. Noticeably, there are also a lot more movies in these three categories, with the most in action. On the other hand, genres such as horror, mystery and romance have a visibly flatter slope, which corresponds to the industry knowledge that certain genres are more conducive to low-budget film making than others, even when provided with additional funding. Thus, we're interested in further exploring the relationship between budget, genre, and success.

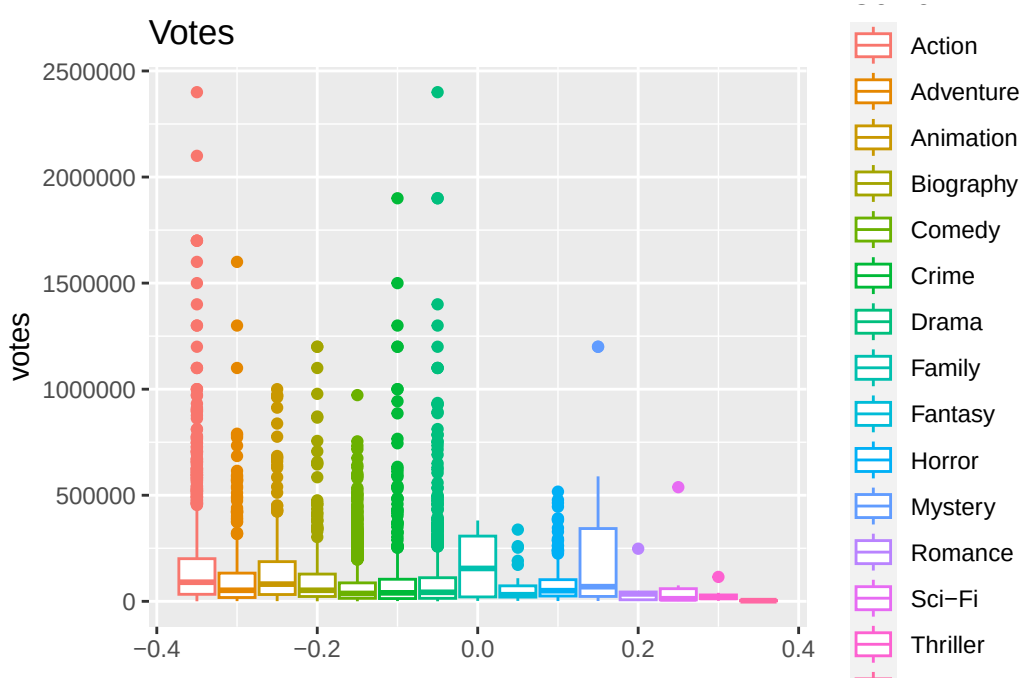
EDA: Visualizatioins and Summary Statistics

The dataset contains many interesting variables that we want to explore fully. We will first do some elementary exploratory data analysis on our datasets to show potential insights that we can explore further.

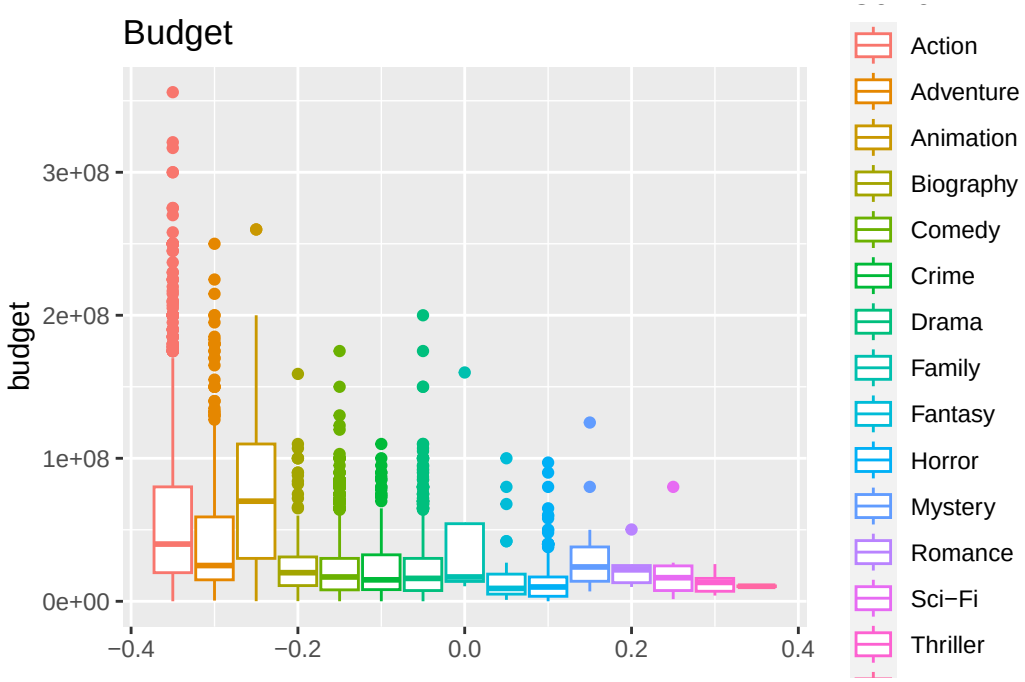
```
ggplot(data = data) +
  geom_boxplot(aes(y = score, color = genre)) +
  labs(title = "Score",
       color = "Genre")
```



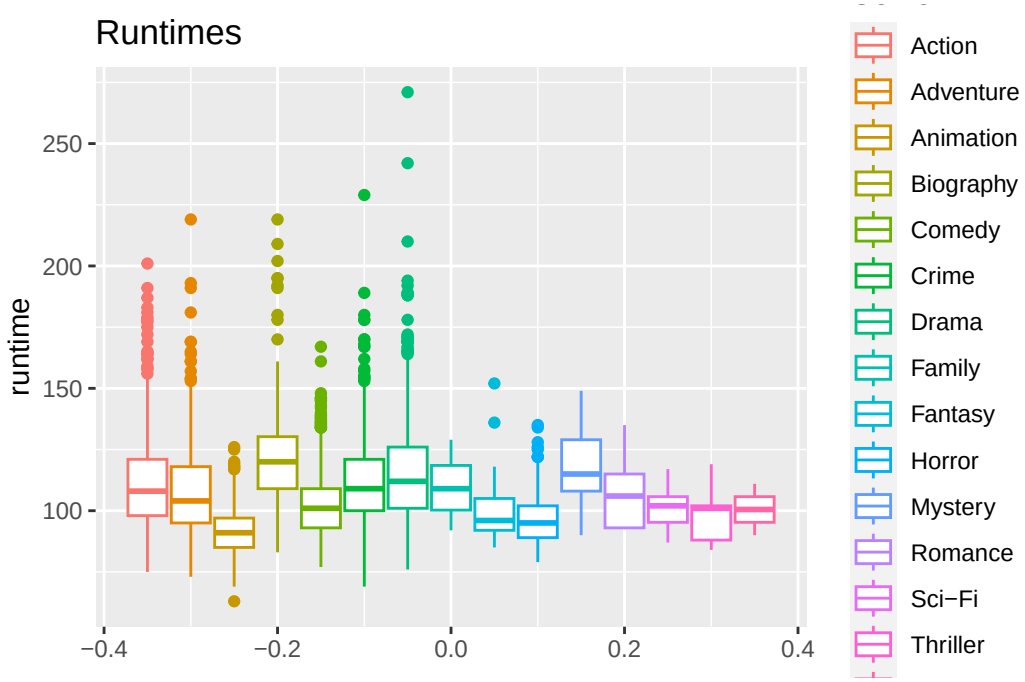

```
ggplot(data = data) +
  geom_boxplot(aes(y = votes, color = genre)) +
  labs(title = "Votes",
       color = "Genre")
```




```
ggplot(data = data) +
  geom_boxplot(aes(y = budget, color = genre)) +
  labs(title = "Budget",
       color = "Genre")
```



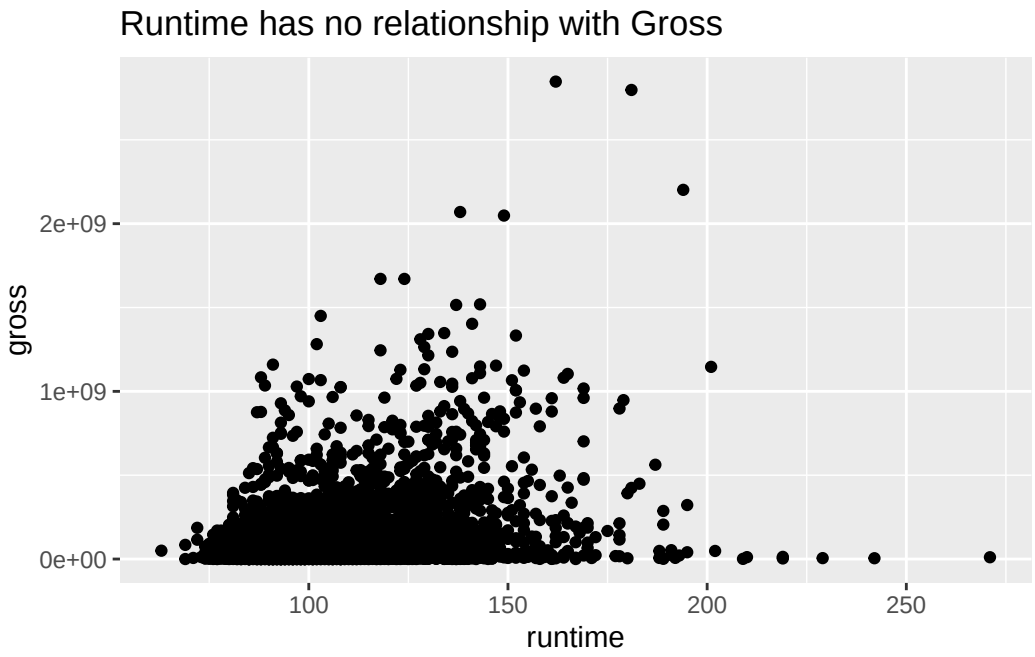

```
ggplot(data = data) +
  geom_boxplot(aes(y = runtime, color = genre)) +
  labs(title = "Runtimes",
       color = "Genre")
```



As can be seen, the score variable is relatively unskewed, whereas the votes, budget, and runtime variables are heavily positively-skewed.

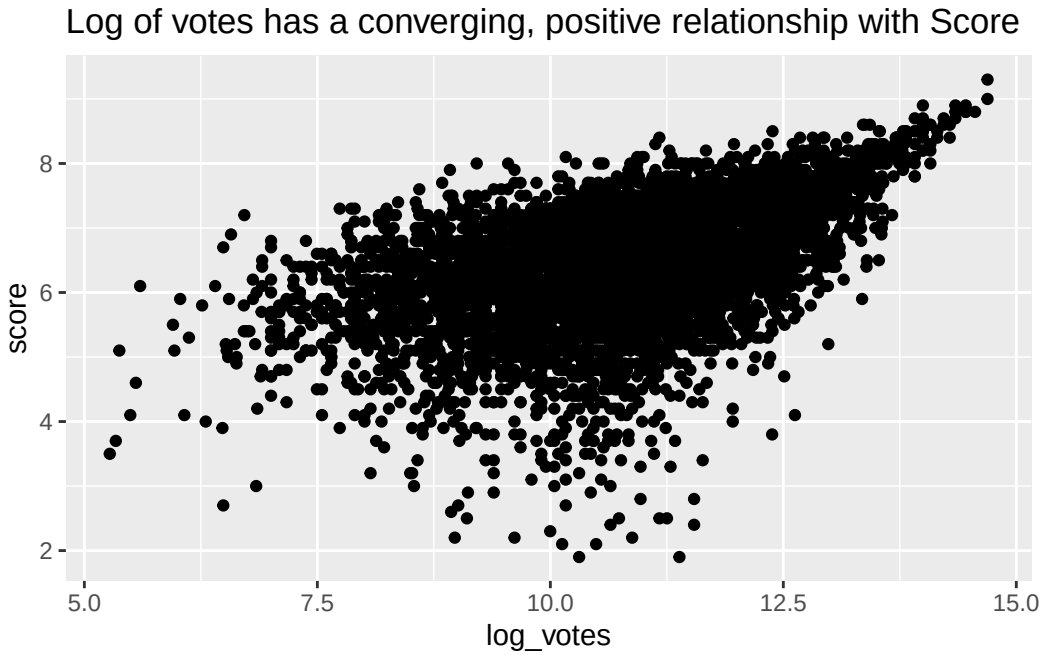
```
data2 <- data %>%
  mutate(log_votes = log(votes))

ggplot(data = data, aes(x = runtime, y = gross)) +
  geom_point() +
  labs(title = "Runtime has no relationship with Gross")
```

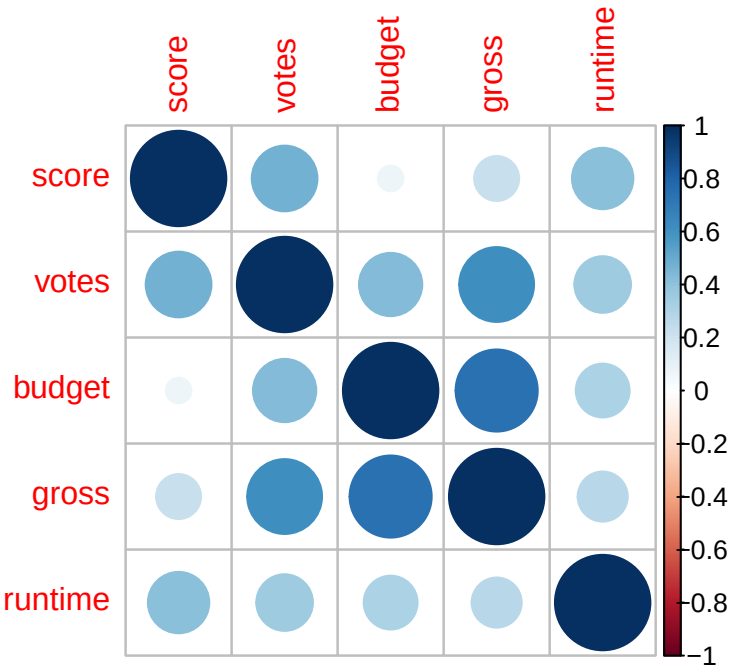



```
data2 <- data %>%
  mutate(log_votes = log(votes))

ggplot(data = data2, aes(x = log_votes, y = score)) +
  geom_point() +
  labs(title = "Log of votes has a converging, positive relationship with Score")
```




```
numeric_data <- data |>
  select("score", "votes", "budget", "gross", "runtime")
corrplot(cor(numeric_data))
```



As can be seen, we have some correlated variables between our predictor variables, but none are particularly strong (0.8+) so we don't have to remove any of these predictor variables on the basis of correlation for our later models. Though there seem to be no particularly strong correlations between predictor variables, the strongest correlation is between votes and score. We may consider adding this into our model because it also fits with our domain knowledge that a movie that is rated by more people might generally have a higher score because, in our experience, people are more motivated to rate a movie they enjoy than one they dislike.

Methodology

As a potential movie investor, we're obviously curious about the prediction of movies' success, and the factors that lead to success. Therefore, we want to explore the following topics in detail in order for us to be better investors: 1) Prediction of movies' gross value 2) the factors that affect the success of the movie. Some of the methods by which we are going to tackle this is through linear regression, residual plots, linear mixed models, LASSO, and repeated k-fold validation.

LASSO (Variable Selection):

As mentioned above we are interested in the relationships between the variables and success, but we still wish to filter variables that are more likely to have a significant effect on our model.

Therefore, we will apply a LASSO variable selection process in order to determine significant variables to explore.

```
set.seed(919)
library(glmnet)
y <- data$log_gross
x <- model.matrix(log_gross ~ budget + genre + rating + score + votes +
                    runtime + country, data = data)

m_lasso_cv <- cv.glmnet(x, y, alpha = 1)
best_lambda <- m_lasso_cv$lambda.min
best_lambda
```

```
[1] 0.01709603
```

```
m_best <- glmnet(x, y, alpha = 1, lambda = best_lambda)
m_best$beta
```

```
77 x 1 sparse Matrix of class "dgCMatrix"
                                                 s0
(Intercept)                                   .
budget                                        1.710419e-08
genreAdventure                               -2.563361e-02
genreAnimation                                3.569258e-01
genreBiography                               -1.090329e-01
genreComedy                                   .
genreCrime                                   -2.735146e-01
genreDrama                                   -3.253671e-01
genreFamily                                   .
genreFantasy                                  .
genreHorror                                   3.553472e-01
genreMystery                                 -6.331902e-02
genreRomance                                 -7.115569e-01
genreSci-Fi                                  -1.842418e-01
genreThriller                                 .
genreWestern                                  .
ratingG                                       .
ratingNC-17                                  -6.514778e-01
ratingNot Rated                              -2.373021e+00
ratingPG                                      7.755363e-03
ratingPG-13                                   .
```

```
ratingR                                -5.430934e-01
ratingTV-MA                            -6.434521e-01
ratingUnrated                          -2.268548e+00
ratingX                                 .
score                                   1.057433e-01
votes                                   2.700184e-06
runtime                                 5.266885e-03
countryAruba                            .
countryAustralia                        .
countryAustria                          .
countryBelgium                         -5.838031e-01
countryBrazil                           .
countryCanada                           .
countryChile                            .
countryChina                            4.377750e-01
countryColombia                         .
countryCzech Republic                   .
countryDenmark                          .
countryFederal Republic of Yugoslavia -1.731288e+00
countryFinland                          1.963027e-01
countryFrance                          -7.354614e-02
countryGermany                          8.225104e-03
countryHong Kong                        1.741440e-01
countryHungary                          .
countryIceland                         -1.548560e+00
countryIndia                            9.056585e-01
countryIndonesia                        .
countryIran                            -3.242545e-01
countryIreland                          .
countryIsrael                           .
countryItaly                           -1.229005e+00
countryJamaica                          .
countryJapan                            .
countryKenya                           -6.597400e-01
countryLebanon                          4.922046e-01
countryMalta                            5.845391e-01
countryMexico                           .
countryNetherlands                      .
countryNew Zealand                     -7.151114e-01
countryNorway                          -2.014447e-01
countryPanama                           .
countryPortugal                         .
countryRepublic of Macedonia            .
```

```
countryRussia                          .
countrySouth Africa                    1.361201e-01
countrySouth Korea                    -1.535603e-01
countrySpain                           .
countrySweden                          .
countrySwitzerland                    -1.013101e+00
countryTaiwan                          1.978536e-01
countryThailand                       -2.026262e+00
countryUnited Arab Emirates            .
countryUnited Kingdom                 -1.502888e-01
countryUnited States                   1.323596e-01
countryWest Germany                    .
countryYugoslavia                      .
```

Some coefficients of the categories of the variables of genre, rating, and country have coefficients were shrunk to zero. Our data is processed in a way where an observation can take only one "level" of a category, for example, a movie can only have one genre. Since even one of the factors of the genre, rating, or country variables has been excluded, we will remove those factor variables as a whole. This is because LASSO performs feature selection, and for a factor variable such as genre, which is set with a baseline of "action", a singular comparison of, say, adventure vs action is not a feature, rather, the comparison of the entire variable versus action is a feature. Similar logic applies to rating, with a baseline of "approved", and country, with a baseline of "Argentina". Our model therefore contains, after LASSO selection, the variables of budget, score, votes, and runtime.

Linear Regression of all variables

Gross revenue is a continuous response variable, so we will first examine a linear regression model for our data. We've gained the equation of XX.

```r
m1 <- lm(log_gross ~ log(budget) + score + log(votes) + log(runtime),
         data = data)
m1_aug <- augment(m1)
summary(m1)
```

```
Call:
lm(formula = log_gross ~ log(budget) + score + log(votes) + log(runtime),
    data = data)

Residuals:
```

```
    Min      1Q  Median      3Q     Max
-9.9698 -0.4906  0.1014  0.6526  5.0475

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   1.70328    0.46455   3.667 0.000248 ***
log(budget)   0.61679    0.01489  41.425  < 2e-16 ***
score        -0.14262    0.02070  -6.890 6.22e-12 ***
log(votes)    0.73265    0.01396  52.467  < 2e-16 ***
log(runtime) -0.38888    0.11556  -3.365 0.000770 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.12 on 5418 degrees of freedom
Multiple R-squared:  0.6482,    Adjusted R-squared:  0.648
F-statistic:  2496 on 4 and 5418 DF,  p-value: < 2.2e-16
```

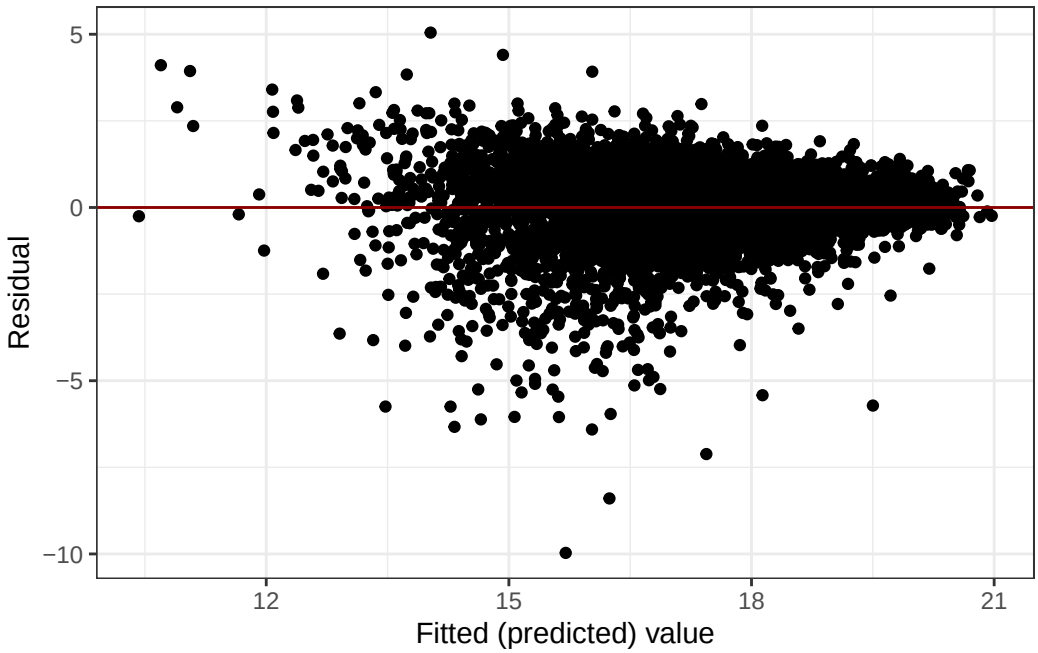
```
rmse(m1_aug, truth = `log_gross`, estimate = .fitted)
```

```
# A tibble: 1 x 3
  .metric .estimator .estimate
  <chr>   <chr>          <dbl>
1 rmse    standard        1.12
```

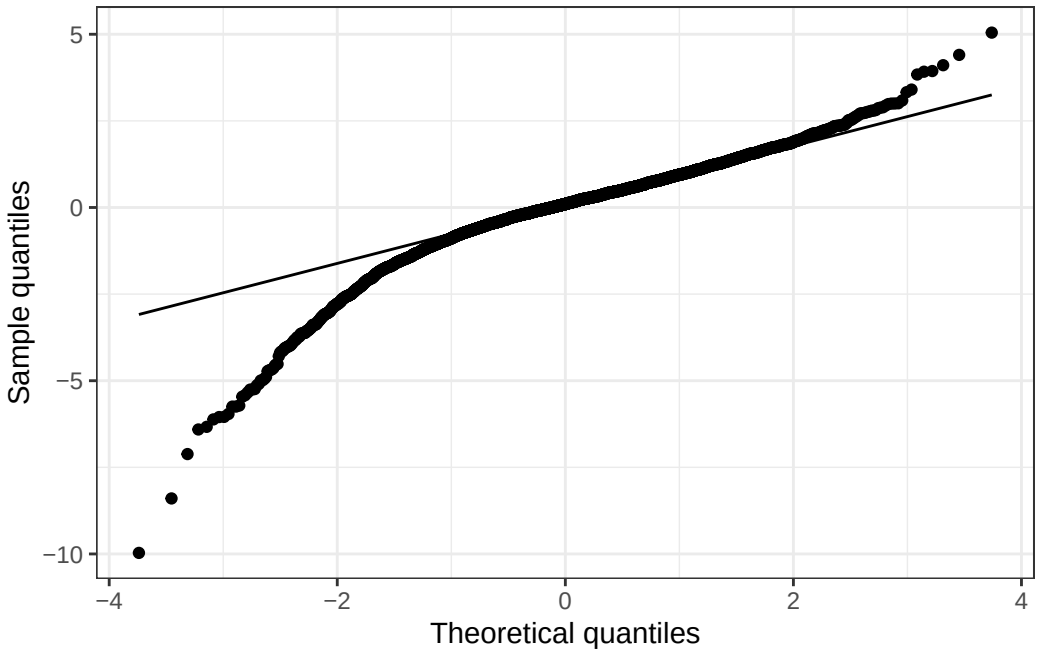
$$\widehat{Gross} = \beta_0 + \beta_1 * log(budget) + \beta_2 * score + \beta_3 * log(votes) + \beta_4 * log(runtime)$$

Residual plots/Assumptions

```
ggplot(m1_aug, aes(x = .fitted, y = .resid)) +
  geom_point() +
  geom_hline(yintercept = 0, color = "darkred") +
  labs(x = "Fitted (predicted) value", y = "Residual") +
  theme_bw()
```




```
ggplot(m1_aug, aes(sample = .resid)) +
  stat_qq() +
  stat_qq_line() +
  theme_bw() +
  labs(x = "Theoretical quantiles",
       y = "Sample quantiles")
```



We may assume that each movie is independent of each other. We can see that the linear assumption is violated since the data is not symmetrically distributed observations around the horizontal axis. The linearity, constant variance, and normality is not satisfied from the above graph. In order to address the normality violation, we did log transformation for our response variable to be `log_gross`. In order to address this problem, we use log transformation to our heavily skewed numeric variables, and in this way, the linearity residual plot is improved, while the constant variance and normality is still violated.

We acknowledge it – limitations XXX

Linear Mixed Model & Random Effect

Since we have the country, genre, and rating these three categorical variables have too many levels, for example, for United States, we have XXX data records, and for XX country, we only have XX variables, so it would make more sense for use to use these three variables as random effects, we want to apply linear mixed model to our datasets to further explore our questions. We want to look at the associations between our interested variables, and random effect due to country, genre, and rating.

```
library(lme4)
m4 <- lm(log_gross ~ log(budget) + genre + rating + score + log(votes) + log(runtime) + co
m5 <- lmer(log_gross ~ 1 + log(budget) + (1 | genre) + (1 | rating) + score + log(votes) +
```

```
summary(m4)$coef
```

```
                                Estimate Std. Error     t value
(Intercept)                  1.65159350 1.33444495  1.23766326
log(budget)                  0.48415938 0.01679556 28.82662187
genreAdventure              -0.09990107 0.06885657 -1.45085749
genreAnimation               0.36501326 0.08981418  4.06409373
genreBiography              -0.16805071 0.07044256 -2.38564188
genreComedy                  0.04168214 0.04211486  0.98972520
genreCrime                  -0.19740631 0.06343845 -3.11177703
genreDrama                  -0.13718647 0.04956153 -2.76800298
genreFamily                  0.24282572 0.53301770  0.45556783
genreFantasy                -0.02665940 0.16930670 -0.15746216
genreHorror                  0.36551537 0.07703046  4.74507549
genreMystery                -0.20584854 0.26109930 -0.78839177
genreRomance                -1.13612775 0.47609923 -2.38632552
genreSci-Fi                  0.06207227 0.43428882  0.14292855
genreThriller                0.66645719 0.40219271  1.65705933
genreWestern                 0.76877876 0.75150656  1.02298344
ratingG                     -1.31830610 1.06987668 -1.23220379
ratingNC-17                 -2.69746979 1.10876553 -2.43285863
ratingNot Rated             -3.23065085 1.07862775 -2.99514903
ratingPG                    -1.48612473 1.06389765 -1.39686814
ratingPG-13                 -1.78956707 1.06418804 -1.68162674
ratingR                     -2.15699794 1.06410119 -2.02706093
ratingTV-MA                 -3.53389750 1.30564570 -2.70662823
ratingUnrated               -2.70343402 1.09542227 -2.46793779
ratingX                     -2.86685730 1.50556554 -1.90417303
score                       -0.14146189 0.02143425 -6.59980726
log(votes)                   0.74078724 0.01396595 53.04238997
log(runtime)                 0.62196874 0.12708166  4.89424489
countryAruba                -0.86524751 1.22632811 -0.70555955
countryAustralia            -0.68960979 0.63524612 -1.08557892
countryAustria              -0.69233063 0.96881264 -0.71461767
countryBelgium              -2.40100570 0.81377481 -2.95045467
countryBrazil               -0.78806069 0.86621728 -0.90977253
countryCanada               -0.78904225 0.62207141 -1.26841106
countryChile                -0.64255223 1.22680505 -0.52376067
countryChina                 0.08393939 0.64937242  0.12926233
countryColombia              0.27767673 1.22498779  0.22667714
countryCzech Republic       -1.10919940 0.73391278 -1.51135043
countryDenmark              -1.33479773 0.69940361 -1.90847988
```

```
countryFederal Republic of Yugoslavia -4.51585701 1.23884387 -3.64521884
countryFinland                         0.09464640 0.97179027  0.09739386
countryFrance                         -1.13927248 0.62229328 -1.83076455
countryGermany                        -0.88037146 0.62517681 -1.40819596
countryHong Kong                      -0.31735357 0.65134418 -0.48722869
countryHungary                        -0.68933412 1.22649268 -0.56203688
countryIceland                        -2.76922855 0.96937165 -2.85672532
countryIndia                           0.07689255 0.73119890  0.10515956
countryIndonesia                      -1.43155327 0.96846171 -1.47817229
countryIran                           -0.38380320 0.87136468 -0.44046220
countryIreland                        -0.82094370 0.65600686 -1.25142548
countryIsrael                         -0.28139878 1.22715527 -0.22930984
countryItaly                          -1.64351788 0.65950678 -2.49204093
countryJamaica                         0.77864579 1.22533770  0.63545404
countryJapan                          -0.70436526 0.64584653 -1.09060779
countryKenya                          -2.25745173 1.22709109 -1.83967738
countryLebanon                         1.16873044 1.22456832  0.95440199
countryMalta                           0.24534579 1.22609240  0.20010384
countryMexico                          0.02591561 0.69164021  0.03746978
countryNetherlands                    -0.83310327 0.86845358 -0.95929511
countryNew Zealand                    -0.86672407 0.66222856 -1.30879898
countryNorway                         -1.04709796 0.86832299 -1.20588533
countryPanama                         -1.21186358 1.22708049 -0.98759909
countryPortugal                       -1.68622686 1.22566802 -1.37576149
countryRepublic of Macedonia          -0.61805137 1.23786420 -0.49928851
countryRussia                         -0.20726705 0.81039961 -0.25575906
countrySouth Africa                   -0.43242735 0.81099378 -0.53320673
countrySouth Korea                    -0.92665698 0.68005359 -1.36262346
countrySpain                          -0.64021639 0.66271448 -0.96605162
countrySweden                         -0.92469479 0.77509782 -1.19300400
countrySwitzerland                    -1.19137615 0.81094534 -1.46912017
countryTaiwan                          0.69653498 0.87249692  0.79832372
countryThailand                       -3.41401578 1.22559917 -2.78558917
countryUnited Arab Emirates           -0.91296776 0.96993643 -0.94126556
countryUnited Kingdom                 -0.84947495 0.61543260 -1.38028916
countryUnited States                  -0.55506434 0.61389799 -0.90416380
countryWest Germany                   -0.68830581 0.86694437 -0.79394461
countryYugoslavia                     -0.29865554 1.22571246 -0.24365873
                                           Pr(>|t|)
(Intercept)                            2.158953e-01
log(budget)                           5.632581e-170
genreAdventure                         1.468782e-01
genreAnimation                         4.891113e-05
```

genreBiography	1.708396e-02
genreComedy	3.223532e-01
genreCrime	1.869456e-03
genreDrama	5.659513e-03
genreFamily	6.487193e-01
genreFantasy	8.748866e-01
genreHorror	2.138564e-06
genreMystery	4.305025e-01
genreRomance	1.705227e-02
genreSci-Fi	8.863520e-01
genreThriller	9.756616e-02
genreWestern	3.063620e-01
ratingG	2.179272e-01
ratingNC-17	1.501273e-02
ratingNot Rated	2.755683e-03
ratingPG	1.625112e-01
ratingPG-13	9.269965e-02
ratingR	4.270571e-02
ratingTV-MA	6.818622e-03
ratingUnrated	1.362047e-02
ratingX	5.694128e-02
score	4.514836e-11
log(votes)	0.000000e+00
log(runtime)	1.015828e-06
countryAruba	4.804928e-01
countryAustralia	2.777143e-01
countryAustria	4.748765e-01
countryBelgium	3.186836e-03
countryBrazil	3.629835e-01
countryCanada	2.047064e-01
countryChile	6.004667e-01
countryChina	8.971549e-01
countryColombia	8.206835e-01
countryCzech Republic	1.307583e-01
countryDenmark	5.638272e-02
countryFederal Republic of Yugoslavia	2.697025e-04
countryFinland	9.224173e-01
countryFrance	6.719133e-02
countryGermany	1.591312e-01
countryHong Kong	6.261163e-01
countryHungary	5.741145e-01
countryIceland	4.296879e-03
countryIndia	9.162531e-01

```
countryIndonesia                        1.394206e-01
countryIran                             6.596202e-01
countryIreland                          2.108340e-01
countryIsrael                           8.186369e-01
countryItaly                            1.273120e-02
countryJamaica                          5.251595e-01
countryJapan                            2.754947e-01
countryKenya                            6.587103e-02
countryLebanon                          3.399234e-01
countryMalta                            8.414070e-01
countryMexico                           9.701118e-01
countryNetherlands                      3.374535e-01
countryNew Zealand                      1.906587e-01
countryNorway                           2.279151e-01
countryPanama                           3.233937e-01
countryPortugal                         1.689531e-01
countryRepublic of Macedonia            6.175967e-01
countryRussia                           7.981467e-01
countrySouth Africa                     5.939126e-01
countrySouth Korea                      1.730586e-01
countrySpain                            3.340621e-01
countrySweden                           2.329208e-01
countrySwitzerland                      1.418590e-01
countryTaiwan                           4.247181e-01
countryThailand                         5.361876e-03
countryUnited Arab Emirates             3.466113e-01
countryUnited Kingdom                   1.675553e-01
countryUnited States                    3.659493e-01
countryWest Germany                     4.272629e-01
countryYugoslavia                       8.075045e-01
```

```
summary(m5)
```

```
Linear mixed model fit by REML ['lmerMod']
Formula: log_gross ~ 1 + log(budget) + (1 | genre) + (1 | rating) + score +
    log(votes) + log(runtime) + (1 | country)
   Data: data

REML criterion at convergence: 16151.1

Scaled residuals:
```

```
    Min      1Q  Median      3Q     Max
-9.4123 -0.4308  0.0930  0.5753  4.3269

Random effects:
 Groups   Name        Variance Std.Dev.
 country  (Intercept) 0.16092  0.4012
 genre    (Intercept) 0.04408  0.2100
 rating   (Intercept) 0.53272  0.7299
 Residual             1.12697  1.0616
Number of obs: 5423, groups:  country, 50; genre, 15; rating, 10

Fixed effects:
             Estimate Std. Error t value
(Intercept)  -1.11070    0.56498  -1.966
log(budget)   0.48409    0.01655  29.244
score        -0.14208    0.02127  -6.680
log(votes)    0.74265    0.01390  53.415
log(runtime)  0.57954    0.12531   4.625

Correlation of Fixed Effects:
            (Intr) lg(bd) score  lg(vt)
log(budget)  0.014
score        0.128  0.354
log(votes)   0.009 -0.442 -0.556
log(runtim) -0.761 -0.434 -0.385  0.075
```

```
r.rmse <- sqrt(mean(residuals(m5)^2))
r.rmse
```

```
[1] 1.057631
```

```
m4_aug <- augment(m4)
rmse(m4_aug, truth = `log_gross`, estimate = .fitted)
```

```
# A tibble: 1 x 3
  .metric .estimator .estimate
  <chr>   <chr>          <dbl>
1 rmse    standard        1.05
```

While adjusting for the random intercept based on country variable and holding all other variables constant, we noticed that the estimated variance is 0.1623 which means that all the country-specific intercept is distributed around the model's overall intercept within estimated variance of 0.1623, given that our unit is log dollars, and our estimated variance is quite small, we may expect that the country is similar. In terms of our interest in knowing whether a movie is successful or not, production of country may not be a big effect on the movie's success.

Since we noticed that genre has the smallest estimated variance (0.08254), we may say that compared to the difference between ratings and countries, the difference between genres is smaller, while rating has the biggest difference. Given that our unit is log dollars, and all of our estimated variance is quite small, the differences between these variables should be similar to each other.

Interested Hypothesis Test 1:

Question 1: Is there evidence to suggest that log(Budget) has an effect on the success of the movies? Null hypothesis: $p_1 = 0$ There isn't sufficient evidence to suggest that budget is associated with movies' gross, while controlling for all of the variables. Alternative hypothesis: $p_1 \neq 0$ There is sufficient evidence to suggest that budget is associated with movies' gross, while controlling for all of the variables.

We use significance level of 0.05. Since the t-statistics is 23.227 and the p-value is $<$ 2e-16 which is much smaller than our significance level, so we reject the null hypothesis since there's sufficient evidence, and thus there's sufficient evidence to suggest to log(budget) does have an effect on the success of the movies (log(gross)).

In terms of qualitatively addressing this issue: in our model, we know that every 1 million increase in our budget, the gross value is expected to increase by e^{cc} dollars. ## Interested Hypothesis Test 2: Aim 2: Is there evidence to suggest that log(Score) has an effect on the success of the movies? Null hypothesis: $p_2 = 0$ There isn't sufficient evidence to suggest that Score is associated with movies' gross, while controlling for all of the variables. Alternative hypothesis: $p_2 \neq 0$ There is sufficient evidence to suggest that Score is associated with movies' gross, while controlling for all of the variables.

We use significance level of 0.05. Since the t-statistics is 23.227 and the p-value is 3.06e-09 which is much smaller than our significance level, so we reject the null hypothesis since there's sufficient evidence, and thus there's sufficient evidence to suggest to genre does have an effect on the success of the movies.

Summary based on Hypothesis:

Based on our previous two hypothesis, we noticed that XXXXX.

Predicting moves' gross

Based on all the models we've done above, we've analyzed the relationship between different variables that we're interested in. In order to make our findings applicable for future use, we'd like to make a prediction on movies' gross based on our current variables.

```
set.seed(123)
dim(data)
```

```
[1] 5423   17
```

```
indices <- sample(1:5423, size = 5423 * 0.8, replace = F)
train.data  <- data %>%
  slice(indices)
test.data <- data %>%
  slice(-indices)
dim(train.data)
```

```
[1] 4338   17
```

```
library(caret)
cv_method <- trainControl(method = "cv", number = 10,
                          repeats = 5)
m1 <- train(log_gross ~ budget + genre + rating + score + votes + runtime +
              country, data = data, method = "lm", trControl = cv_method)
m2 <- train(log_gross ~ budget + genre + rating + score + votes + runtime +
              country, data = data, method = "lm", trControl = cv_method)

print(m2)
```

```
Linear Regression

5423 samples
   7 predictor

No pre-processing
Resampling: Cross-Validated (10 fold)
Summary of sample sizes: 4883, 4879, 4882, 4881, 4881, 4880, ...
Resampling results:
```

```
  RMSE    Rsquared   MAE
  1.3736  0.4729455  1.0154

Tuning parameter 'intercept' was held constant at a value of TRUE
```

```r
library(caret)
cv_method <- trainControl(method = "cv", number = 10,
                          repeats = 5)
m3 <- train(log_gross ~ log(budget) + genre + rating + log(score) + log(votes) +
              log(runtime) + country,
            data = data, method = "lm", trControl = cv_method)
m4 <- train(log_gross ~ log(budget) + genre + rating + log(score) + log(votes) +
              log(runtime) + country,
            data = data, method = "lm", trControl = cv_method)

print(m4)
```

```
Linear Regression

5423 samples
   7 predictor

No pre-processing
Resampling: Cross-Validated (10 fold)
Summary of sample sizes: 4881, 4879, 4879, 4883, 4881, 4881, ...
Resampling results:

  RMSE      Rsquared   MAE
  1.083085  0.6717084  0.7602062

Tuning parameter 'intercept' was held constant at a value of TRUE
```

The RMSE from first model is XXX; while the RMSE from the second model is XXXX. Since the second one is smaller, the second one is better performing model in predicting the gross values of the movie. XXXX Since better, we might use log/ or fewer variables in predicting our gross values.

Results

Explain the reasoning for the type of model you're fitting, predictor variables considered for the model including any interactions. Additionally, show how you arrived at the final model by describing the model selection process, interactions considered, variable transformations (if needed), assessment of conditions and diagnostics, and any other relevant considerations that were part of the model fitting process.

Discussion & Limitations

Summary + statistical arguments to support my conclusions + future limitations/future ideads

Variable selection, Rescaling predictor variables, linearity assumptions,

Sources

https://towardsdatascience.com/feature-selection-in-machine-learning-using-lasso-regression-7809c7c2771a https://stackoverflow.com/questions/13646654/root-mean-square-error-in-r-mixed-effect-model