Data 501 Fall 2021: Semester End Project

Amit Kumar Srivastava, Priyanka Lalge, Shruti Shukla, Srujan Kumar Nunna Due Date = 12/17/2021

**Introduction**

As part of the MSDA Data 501 final project, our group decided to research on the following questions.

• Is an Oscar winning actor or actress in the cast associated with the IMDB rating of the movie? • Is there a difference in mean audience scores between genres?

• Which variables are associated with, and hence can be used to predict, the Rating of a movie on IMDB?

**Team Work**

First, we started working on the research questions individually, later discussed and picked the best amongst these questions. After this, we decided that each member of the group shall try to come up with the prediction model individually in order to get familiar with all the related concepts. All of us came up with possible solutions for the research questions and then collaborated and enhanced our works to get the best possible solution.

Loading required libraries. . .

library(tidyverse)

## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5 v purrr 0.3.4

## v tibble 3.1.5 v dplyr 1.0.7

## v tidyr 1.1.4 v stringr 1.4.0

## v readr 2.0.2 v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse\_conflicts() -- ## x dplyr::filter() masks stats::filter()

## x dplyr::lag() masks stats::lag()

library(dplyr)

library(car)

## Loading required package: carData

##

## Attaching package: ’car’

1

## The following object is masked from ’package:dplyr’:

##

## recode

## The following object is masked from ’package:purrr’:

##

## some

library(ggcorrplot)

library(GGally)

## Registered S3 method overwritten by ’GGally’:

## method from

## +.gg ggplot2

library(car)

library(MASS)

##

## Attaching package: ’MASS’

## The following object is masked from ’package:dplyr’:

##

## select

**Exploratory data analysis**

We load the data from the url provided. Once the data is loaded we took a glance on the summary and structure of the dataset.

load(url("http://people.math.binghamton.edu/qiao/data501/data/movies.RData")) head(movies)

## # A tibble: 6 x 32

## title title\_type genre runtime mpaa\_rating studio thtr\_rel\_year thtr\_rel\_month ## <chr> <fct> <fct> <dbl> <fct> <fct> <dbl> <dbl> ## 1 Fill~ Feature F~ Drama 80 R Indom~ 2013 4 ## 2 The ~ Feature F~ Drama 101 PG-13 Warne~ 2001 3 ## 3 Wait~ Feature F~ Come~ 84 R Sony ~ 1996 8 ## 4 The ~ Feature F~ Drama 139 PG Colum~ 1993 10 ## 5 Male~ Feature F~ Horr~ 90 R Ancho~ 2004 9 ## 6 Old ~ Documenta~ Docu~ 78 Unrated Shcal~ 2009 1 ## # ... with 24 more variables: thtr\_rel\_day <dbl>, dvd\_rel\_year <dbl>,

## # dvd\_rel\_month <dbl>, dvd\_rel\_day <dbl>, imdb\_rating <dbl>,

## # imdb\_num\_votes <int>, critics\_rating <fct>, critics\_score <dbl>, ## # audience\_rating <fct>, audience\_score <dbl>, best\_pic\_nom <fct>, ## # best\_pic\_win <fct>, best\_actor\_win <fct>, best\_actress\_win <fct>, ## # best\_dir\_win <fct>, top200\_box <fct>, director <chr>, actor1 <chr>, ## # actor2 <chr>, actor3 <chr>, actor4 <chr>, actor5 <chr>, imdb\_url <chr>, ...

2

summary(movies)

## title title\_type genre runtime ## Length:651 Documentary : 55 Drama :305 Min. : 39.0 ## Class :character Feature Film:591 Comedy : 87 1st Qu.: 92.0 ## Mode :character TV Movie : 5 Action & Adventure: 65 Median :103.0 ## Mystery & Suspense: 59 Mean :105.8 ## Documentary : 52 3rd Qu.:115.8 ## Horror : 23 Max. :267.0 ## (Other) : 60 NA’s :1 ## mpaa\_rating studio thtr\_rel\_year ## G : 19 Paramount Pictures : 37 Min. :1970 ## NC-17 : 2 Warner Bros. Pictures : 30 1st Qu.:1990 ## PG :118 Sony Pictures Home Entertainment: 27 Median :2000 ## PG-13 :133 Universal Pictures : 23 Mean :1998 ## R :329 Warner Home Video : 19 3rd Qu.:2007 ## Unrated: 50 (Other) :507 Max. :2014 ## NA’s : 8

## thtr\_rel\_month thtr\_rel\_day dvd\_rel\_year dvd\_rel\_month ## Min. : 1.00 Min. : 1.00 Min. :1991 Min. : 1.000 ## 1st Qu.: 4.00 1st Qu.: 7.00 1st Qu.:2001 1st Qu.: 3.000 ## Median : 7.00 Median :15.00 Median :2004 Median : 6.000 ## Mean : 6.74 Mean :14.42 Mean :2004 Mean : 6.333 ## 3rd Qu.:10.00 3rd Qu.:21.00 3rd Qu.:2008 3rd Qu.: 9.000 ## Max. :12.00 Max. :31.00 Max. :2015 Max. :12.000 ## NA’s :8 NA’s :8

## dvd\_rel\_day imdb\_rating imdb\_num\_votes critics\_rating ## Min. : 1.00 Min. :1.900 Min. : 180 Certified Fresh:135 ## 1st Qu.: 7.00 1st Qu.:5.900 1st Qu.: 4546 Fresh :209 ## Median :15.00 Median :6.600 Median : 15116 Rotten :307 ## Mean :15.01 Mean :6.493 Mean : 57533

## 3rd Qu.:23.00 3rd Qu.:7.300 3rd Qu.: 58300

## Max. :31.00 Max. :9.000 Max. :893008

## NA’s :8

## critics\_score audience\_rating audience\_score best\_pic\_nom best\_pic\_win ## Min. : 1.00 Spilled:275 Min. :11.00 no :629 no :644 ## 1st Qu.: 33.00 Upright:376 1st Qu.:46.00 yes: 22 yes: 7 ## Median : 61.00 Median :65.00

## Mean : 57.69 Mean :62.36

## 3rd Qu.: 83.00 3rd Qu.:80.00

## Max. :100.00 Max. :97.00

##

## best\_actor\_win best\_actress\_win best\_dir\_win top200\_box director ## no :558 no :579 no :608 no :636 Length:651 ## yes: 93 yes: 72 yes: 43 yes: 15 Class :character ## Mode :character ##

##

##

##

## actor1 actor2 actor3 actor4 ## Length:651 Length:651 Length:651 Length:651 ## Class :character Class :character Class :character Class :character

3

## Mode :character Mode :character Mode :character Mode :character

##

##

##

##

## actor5 imdb\_url rt\_url

## Length:651 Length:651 Length:651

## Class :character Class :character Class :character

## Mode :character Mode :character Mode :character

##

##

##

##

str(movies)

## tibble [651 x 32] (S3: tbl\_df/tbl/data.frame)

## $ title : chr [1:651] "Filly Brown" "The Dish" "Waiting for Guffman" "The Age of Innocenc## $ title\_type : Factor w/ 3 levels "Documentary",..: 2 2 2 2 2 1 2 2 1 2 ... ## $ genre : Factor w/ 11 levels "Action & Adventure",..: 6 6 4 6 7 5 6 6 5 6 ... ## $ runtime : num [1:651] 80 101 84 139 90 78 142 93 88 119 ...

## $ mpaa\_rating : Factor w/ 6 levels "G","NC-17","PG",..: 5 4 5 3 5 6 4 5 6 6 ... ## $ studio : Factor w/ 211 levels "20th Century Fox",..: 91 202 167 34 13 163 147 118 88 84 ## $ thtr\_rel\_year : num [1:651] 2013 2001 1996 1993 2004 ...

## $ thtr\_rel\_month : num [1:651] 4 3 8 10 9 1 1 11 9 3 ...

## $ thtr\_rel\_day : num [1:651] 19 14 21 1 10 15 1 8 7 2 ...

## $ dvd\_rel\_year : num [1:651] 2013 2001 2001 2001 2005 ...

## $ dvd\_rel\_month : num [1:651] 7 8 8 11 4 4 2 3 1 8 ...

## $ dvd\_rel\_day : num [1:651] 30 28 21 6 19 20 18 2 21 14 ...

## $ imdb\_rating : num [1:651] 5.5 7.3 7.6 7.2 5.1 7.8 7.2 5.5 7.5 6.6 ... ## $ imdb\_num\_votes : int [1:651] 899 12285 22381 35096 2386 333 5016 2272 880 12496 ... ## $ critics\_rating : Factor w/ 3 levels "Certified Fresh",..: 3 1 1 1 3 2 3 3 2 1 ... ## $ critics\_score : num [1:651] 45 96 91 80 33 91 57 17 90 83 ...

## $ audience\_rating : Factor w/ 2 levels "Spilled","Upright": 2 2 2 2 1 2 2 1 2 2 ... ## $ audience\_score : num [1:651] 73 81 91 76 27 86 76 47 89 66 ...

## $ best\_pic\_nom : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ... ## $ best\_pic\_win : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ... ## $ best\_actor\_win : Factor w/ 2 levels "no","yes": 1 1 1 2 1 1 1 2 1 1 ... ## $ best\_actress\_win: Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ... ## $ best\_dir\_win : Factor w/ 2 levels "no","yes": 1 1 1 2 1 1 1 1 1 1 ... ## $ top200\_box : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ... ## $ director : chr [1:651] "Michael D. Olmos" "Rob Sitch" "Christopher Guest" "Martin Scorsese

## $ actor1 : chr [1:651] "Gina Rodriguez" "Sam Neill" "Christopher Guest" "Daniel Day-Lewis"## $ actor2 : chr [1:651] "Jenni Rivera" "Kevin Harrington" "Catherine O’Hara" "Michelle Pfei## $ actor3 : chr [1:651] "Lou Diamond Phillips" "Patrick Warburton" "Parker Posey" "Winona R## $ actor4 : chr [1:651] "Emilio Rivera" "Tom Long" "Eugene Levy" "Richard E. Grant" ...

## $ actor5 : chr [1:651] "Joseph Julian Soria" "Genevieve Mooy" "Bob Balaban" "Alec McCowen"## $ imdb\_url : chr [1:651] "http://www.imdb.com/title/tt1869425/" "http://www.imdb.com/title/t## $ rt\_url : chr [1:651] "//www.rottentomatoes.com/m/filly\_brown\_2012/" "//www.rottentomatoe

4

**Preprocessing**

The columns are segregated into two lists - cat\_var containing the categorical variables and cont\_vars with the continuous variables. Factor is applied to the categorical variables in the movies dataset.

title, actor1, actor2, actor3, actor4, actor5, imdb\_url, rt\_url columns are not considered at all as these variables doesn’t have much significance as per our research orientation. Further director and studio are also removed as the structure contains 200+ levels in the structure.

thtr\_rel\_year, thtr\_rel\_day, dvd\_rel\_year, dvd\_rel\_day is the list of variables that are not considered as we assumed it would be better to deal with the months rather than year and days.

Lastly, the rows with NA values are removed from the dataset.

names(movies)

## [1] "title" "title\_type" "genre" "runtime"

## [5] "mpaa\_rating" "studio" "thtr\_rel\_year" "thtr\_rel\_month" ## [9] "thtr\_rel\_day" "dvd\_rel\_year" "dvd\_rel\_month" "dvd\_rel\_day" ## [13] "imdb\_rating" "imdb\_num\_votes" "critics\_rating" "critics\_score" ## [17] "audience\_rating" "audience\_score" "best\_pic\_nom" "best\_pic\_win" ## [21] "best\_actor\_win" "best\_actress\_win" "best\_dir\_win" "top200\_box"

## [25] "director" "actor1" "actor2" "actor3"

## [29] "actor4" "actor5" "imdb\_url" "rt\_url"

movies1 <- subset(movies, select = -c(title, studio, thtr\_rel\_year, thtr\_rel\_day, dvd\_rel\_year, dvd\_relsummary(movies1)

## title\_type genre runtime mpaa\_rating

## Documentary : 55 Drama :305 Min. : 39.0 G : 19

## Feature Film:591 Comedy : 87 1st Qu.: 92.0 NC-17 : 2

## TV Movie : 5 Action & Adventure: 65 Median :103.0 PG :118

## Mystery & Suspense: 59 Mean :105.8 PG-13 :133

## Documentary : 52 3rd Qu.:115.8 R :329

## Horror : 23 Max. :267.0 Unrated: 50

## (Other) : 60 NA’s :1

## thtr\_rel\_month dvd\_rel\_month imdb\_rating imdb\_num\_votes

## Min. : 1.00 Min. : 1.000 Min. :1.900 Min. : 180

## 1st Qu.: 4.00 1st Qu.: 3.000 1st Qu.:5.900 1st Qu.: 4546

## Median : 7.00 Median : 6.000 Median :6.600 Median : 15116

## Mean : 6.74 Mean : 6.333 Mean :6.493 Mean : 57533

## 3rd Qu.:10.00 3rd Qu.: 9.000 3rd Qu.:7.300 3rd Qu.: 58300

## Max. :12.00 Max. :12.000 Max. :9.000 Max. :893008

## NA’s :8

## critics\_rating critics\_score audience\_rating audience\_score

## Certified Fresh:135 Min. : 1.00 Spilled:275 Min. :11.00

## Fresh :209 1st Qu.: 33.00 Upright:376 1st Qu.:46.00

## Rotten :307 Median : 61.00 Median :65.00

## Mean : 57.69 Mean :62.36

## 3rd Qu.: 83.00 3rd Qu.:80.00

## Max. :100.00 Max. :97.00

##

## best\_pic\_nom best\_pic\_win best\_actor\_win best\_actress\_win best\_dir\_win

5

## no :629 no :644 no :558 no :579 no :608 ## yes: 22 yes: 7 yes: 93 yes: 72 yes: 43 ##

##

##

##

##

## top200\_box

## no :636

## yes: 15

##

##

##

##

##

movies2 <- movies1 %>% filter(!is.na(runtime), !is.na(dvd\_rel\_month)) summary(movies2)

## title\_type genre runtime mpaa\_rating ## Documentary : 52 Drama :303 Min. : 39.00 G : 18 ## Feature Film:585 Comedy : 87 1st Qu.: 92.25 NC-17 : 2 ## TV Movie : 5 Action & Adventure: 63 Median :103.00 PG :115 ## Mystery & Suspense: 59 Mean :105.93 PG-13 :132 ## Documentary : 49 3rd Qu.:116.00 R :327 ## Horror : 23 Max. :267.00 Unrated: 48 ## (Other) : 58

## thtr\_rel\_month dvd\_rel\_month imdb\_rating imdb\_num\_votes ## Min. : 1.000 Min. : 1.000 Min. :1.9 Min. : 180 ## 1st Qu.: 4.000 1st Qu.: 3.000 1st Qu.:5.9 1st Qu.: 4830 ## Median : 7.000 Median : 6.000 Median :6.6 Median : 15508 ## Mean : 6.737 Mean : 6.341 Mean :6.5 Mean : 58296 ## 3rd Qu.:10.000 3rd Qu.: 9.000 3rd Qu.:7.3 3rd Qu.: 59034 ## Max. :12.000 Max. :12.000 Max. :9.0 Max. :893008 ##

## critics\_rating critics\_score audience\_rating audience\_score ## Certified Fresh:135 Min. : 1.00 Spilled:271 Min. :11.00 ## Fresh :206 1st Qu.: 33.00 Upright:371 1st Qu.:46.00 ## Rotten :301 Median : 61.50 Median :65.00 ## Mean : 57.84 Mean :62.44 ## 3rd Qu.: 83.00 3rd Qu.:80.00 ## Max. :100.00 Max. :97.00 ##

## best\_pic\_nom best\_pic\_win best\_actor\_win best\_actress\_win best\_dir\_win ## no :620 no :635 no :549 no :570 no :599 ## yes: 22 yes: 7 yes: 93 yes: 72 yes: 43 ##

##

##

##

##

## top200\_box

## no :627

## yes: 15

6

##

##

##

##

##

cat\_vars <- c("title\_type", "genre", "mpaa\_rating", "critics\_rating", "audience\_rating", "best\_pic\_nom"cont\_vars <- c("runtime", "imdb\_rating", "imdb\_num\_votes", "critics\_score", "audience\_score") movies2[cat\_vars] = lapply(movies2[cat\_vars], factor)

Now, we start with the exploration of the data. First of all we explore the continuous variables. Here we observed the descriptive summary of the variables as well as the correlation among the variables.

summary(movies2[cont\_vars])

## runtime imdb\_rating imdb\_num\_votes critics\_score

## Min. : 39.00 Min. :1.9 Min. : 180 Min. : 1.00

## 1st Qu.: 92.25 1st Qu.:5.9 1st Qu.: 4830 1st Qu.: 33.00

## Median :103.00 Median :6.6 Median : 15508 Median : 61.50

## Mean :105.93 Mean :6.5 Mean : 58296 Mean : 57.84

## 3rd Qu.:116.00 3rd Qu.:7.3 3rd Qu.: 59034 3rd Qu.: 83.00

## Max. :267.00 Max. :9.0 Max. :893008 Max. :100.00

## audience\_score

## Min. :11.00

## 1st Qu.:46.00

## Median :65.00

## Mean :62.44

## 3rd Qu.:80.00

## Max. :97.00

corr <- round(cor(movies2[cont\_vars]), 1)

head(corr)

## runtime imdb\_rating imdb\_num\_votes critics\_score audience\_score ## runtime 1.0 0.3 0.3 0.2 0.2 ## imdb\_rating 0.3 1.0 0.3 0.8 0.9 ## imdb\_num\_votes 0.3 0.3 1.0 0.2 0.3 ## critics\_score 0.2 0.8 0.2 1.0 0.7 ## audience\_score 0.2 0.9 0.3 0.7 1.0

As per the correlation matrix, we could observe a correlation between imdb\_rating, critics\_score and audience\_score. The variable runtime is not correlated significantly.

*# Visualize the correlation matrix*

*# --------------------------------*

*# method = "square" (default)*

ggcorrplot(corr, method = "circle")

## Warning: ‘guides(<scale> = FALSE)‘ is deprecated. Please use ‘guides(<scale> = ## "none")‘ instead.

7

audience\_score critics\_score

imdb\_num\_votes imdb\_rating

runtime

imdb\_num\_votescritics\_score

Corr

1.0

0.5

0.0

−0.5 −1.0

imdb\_rating

runtime

audience\_score

After the observation of the continuous variables, we proceed towards the categorical variables. As the main question revolves around the imdb\_rating, all the categorical variables are plotted against imdb\_rating.

ggplot(movies2, aes(x = factor(title\_type), y = imdb\_rating)) + geom\_boxplot() 8

g

n

it

a

r

\_

bd

m

i

8

6

4

2

Documentary Feature Film TV Movie factor(title\_type)

ggplot(movies2, aes(x = factor(genre), y = imdb\_rating)) + geom\_boxplot() + theme(axis.text.x = element9

g

n

it

a

r

\_

bd

m

i

8

6

4

2

Art House & InternationalComedyDocumentaryDramaHorror

Mystery & SuspenseOther

Action & AdventureAnimation

Musical & Performing Arts

factor(genre)

Science Fiction & Fantasy

ggplot(movies2, aes(x = factor(mpaa\_rating), y = imdb\_rating)) + geom\_boxplot() 10

g

n

it

a

r

\_

bd

m

i

8

6

4

2

G NC−17 PG PG−13 R Unrated factor(mpaa\_rating)

ggplot(movies2, aes(x = factor(critics\_rating), y = imdb\_rating)) + geom\_boxplot() 11

g

n

it

a

r

\_

bd

m

i

8

6

4

2

Certified Fresh Fresh Rotten factor(critics\_rating)

ggplot(movies2, aes(x = factor(audience\_rating), y = imdb\_rating)) + geom\_boxplot() 12

g

n

it

a

r

\_

bd

m

i

8

6

4

2

Spilled Upright

factor(audience\_rating)

ggplot(movies2, aes(x = factor(best\_pic\_nom), y = imdb\_rating)) + geom\_boxplot() 13

g

n

it

a

r

\_

bd

m

i

8

6

4

2

no yes

factor(best\_pic\_nom)

ggplot(movies2, aes(x = factor(best\_pic\_win), y = imdb\_rating)) + geom\_boxplot() 14

g

n

it

a

r

\_

bd

m

i

8

6

4

2

no yes

factor(best\_pic\_win)

ggplot(movies2, aes(x = factor(best\_actor\_win), y = imdb\_rating)) + geom\_boxplot() 15

g

n

it

a

r

\_

bd

m

i

8

6

4

2

no yes

factor(best\_actor\_win)

ggplot(movies2, aes(x = factor(best\_actress\_win), y = imdb\_rating)) + geom\_boxplot() 16

g

n

it

a

r

\_

bd

m

i

8

6

4

2

no yes

factor(best\_actress\_win)

ggplot(movies2, aes(x = factor(best\_dir\_win), y = imdb\_rating)) + geom\_boxplot() 17

g

n

it

a

r

\_

bd

m

i

8

6

4

2

no yes

factor(best\_dir\_win)

ggplot(movies2, aes(x = factor(top200\_box), y = imdb\_rating)) + geom\_boxplot() 18

g

n

it

a

r

\_

bd

m

i

8

6

4

2

no yes

factor(top200\_box)

ggplot(movies2, aes(x = factor(thtr\_rel\_month), y = imdb\_rating)) + geom\_boxplot() 19

g

n

it

a

r

\_

bd

m

i

8

6

4

2

1 2 3 4 5 6 7 8 9 10 11 12 factor(thtr\_rel\_month)

ggplot(movies2, aes(x = factor(dvd\_rel\_month), y = imdb\_rating)) + geom\_boxplot() 20

g

n

it

a

r

\_

bd

m

i

8

6

4

2

1 2 3 4 5 6 7 8 9 10 11 12 factor(dvd\_rel\_month)

21

**Inference**

The purpose of this section is to use the statistical inference tool of t-test and check the first question of our research, i.e.,

• Is an Oscar winning actor or actress in the cast associated with the IMDB rating of the movie?

To answer the question, first of all, we split the dataset into two subsets. One of the subset contains the movies that casted either a best winning actor or actress. The second set contains the movies that did not cast an oscar winning actor/actress.

movies\_oscar\_cast = movies2[(movies2$best\_actor\_win=='yes' | movies2$best\_actress\_win == 'yes'),] movies\_without\_oscar\_cast = movies2[(movies2$best\_actor\_win=='no' & movies2$best\_actress\_win =='no'),] head(movies\_oscar\_cast)

## # A tibble: 6 x 18

## title\_type genre runtime mpaa\_rating thtr\_rel\_month dvd\_rel\_month imdb\_rating ## <fct> <fct> <dbl> <fct> <fct> <fct> <dbl> ## 1 Feature Fi~ Drama 139 PG 10 11 7.2 ## 2 Feature Fi~ Drama 93 R 11 3 5.5 ## 3 Feature Fi~ Acti~ 127 PG 6 5 6.8 ## 4 Feature Fi~ Come~ 110 R 1 7 7.6 ## 5 Feature Fi~ Drama 96 R 8 12 7

## 6 Feature Fi~ Drama 124 R 6 6 7 ## # ... with 11 more variables: imdb\_num\_votes <int>, critics\_rating <fct>,

## # critics\_score <dbl>, audience\_rating <fct>, audience\_score <dbl>,

## # best\_pic\_nom <fct>, best\_pic\_win <fct>, best\_actor\_win <fct>,

## # best\_actress\_win <fct>, best\_dir\_win <fct>, top200\_box <fct>

Once the data is split, we come up with our null and alternate hypothesis and perform the two-sample t-test.

*H*0: There is no difference in imdb rating for movies casted by oscar won actor/actress *Ha*: There is a difference in imdb rating for movies casted by oscar won actor/actress

Note: Here we assumed variance to be equal in order to simplify our research.

t.test(movies\_oscar\_cast$imdb\_rating, movies\_without\_oscar\_cast$imdb\_rating , alt = "two.sided", conf =

##

## Two Sample t-test

##

## data: movies\_oscar\_cast$imdb\_rating and movies\_without\_oscar\_cast$imdb\_rating ## t = 1.7087, df = 640, p-value = 0.08798

## alternative hypothesis: true difference in means is not equal to 0

## 95 percent confidence interval:

## -0.02581435 0.37187138

## sample estimates:

## mean of x mean of y

## 6.633793 6.460765

Since the p-value ≮ *α* = 0*.*05, hence we fail to reject the *H*0. It implies that there is no difference in imdb rating for movies casted by oscar won actor/actress.

22

**Modeling**

We shall proceed to split the data into two subsets for modeling purpose - train dataset with 70% rows and test dataset with 30% rows of the movies2 dataset. Further, we shall take the help of the model selection techniques to consider the adequate predictor variables. In addition, we will validate the MSE for various models and select the one with the least MSE.

set.seed(1234)

index = sample(c(rep(TRUE,450), rep(FALSE,192)))

mov\_train = movies2[index, ]

mov\_test = movies2[!index, ]

dim(mov\_train)

## [1] 450 18

dim(mov\_test)

## [1] 192 18

With the split of dataset in place, we will start with question 2 and also with the model selection process.

Now, to answer the 2nd question of our project, we shall perform one way anova test using either the pairwise t-test for the genre categorical variable or TukeyHSD. We proceed with the TukeyHSD below.

Since our anova is significant (p-value= 2*e −* 16 *< α* = 0*.*05) hence we performed TukeyHSD which gives the pairwise comparison of means. From the results of the graph we have found that Documentary-Action & Adventure, Drama-Action & Adventure, Musical & Performing Arts-Action & Adventure, Documentary Comedy, Drama-Comedy, Musical & Performing Arts-Comedy, Drama-Documentary, Horror-Documentary, Mystery & Suspense-Documentary, Other-Documentary, Horror-Drama, Musical & Performing Arts-Horror, Mystery & Suspense-Musical & Performing Arts these pairs are significant and their mean difference would not be zero.

mov.aov = aov(audience\_score ~ genre , data = mov\_train)

summary(mov.aov)

## Df Sum Sq Mean Sq F value Pr(>F)

## genre 10 37865 3787 12.24 <2e-16 \*\*\*

## Residuals 439 135855 309

## ---

## Signif. codes: 0 ’\*\*\*’ 0.001 ’\*\*’ 0.01 ’\*’ 0.05 ’.’ 0.1 ’ ’ 1

TukeyHSD(mov.aov)

## Tukey multiple comparisons of means

## 95% family-wise confidence level

##

## Fit: aov(formula = audience\_score ~ genre, data = mov\_train)

##

## $genre

## diff lwr ## Animation-Action & Adventure 10.0353535 -10.7892843

23

## Art House & International-Action & Adventure 10.4909091 -9.4505833 ## Comedy-Action & Adventure 1.4454545 -10.0677715 ## Documentary-Action & Adventure 29.7372506 17.3812388 ## Drama-Action & Adventure 14.2686342 4.8347696 ## Horror-Action & Adventure -10.6948052 -28.1615010 ## Musical & Performing Arts-Action & Adventure 26.5909091 6.6494167 ## Mystery & Suspense-Action & Adventure 4.5665188 -7.7894929 ## Other-Action & Adventure 9.3181818 -9.8705282 ## Science Fiction & Fantasy-Action & Adventure 16.3409091 -13.3861126 ## Art House & International-Animation 0.4555556 -25.6987180 ## Comedy-Animation -8.5898990 -29.0578563 ## Documentary-Animation 19.7018970 -1.2517168 ## Drama-Animation 4.2332807 -15.1414634 ## Horror-Animation -20.7301587 -45.0502814 ## Musical & Performing Arts-Animation 16.5555556 -9.5987180 ## Mystery & Suspense-Animation -5.4688347 -26.4224485 ## Other-Animation -0.7171717 -26.3021184 ## Science Fiction & Fantasy-Animation 6.3055556 -27.9008580 ## Comedy-Art House & International -9.0454545 -28.6141759 ## Documentary-Art House & International 19.2463415 -0.8298013 ## Drama-Art House & International 3.7777251 -14.6444975 ## Horror-Art House & International -21.1857143 -44.7540393 ## Musical & Performing Arts-Art House & International 16.1000000 -9.3567005 ## Mystery & Suspense-Art House & International -5.9243902 -26.0005330 ## Other-Art House & International -1.1727273 -26.0441381 ## Science Fiction & Fantasy-Art House & International 5.8500000 -27.8260494 ## Documentary-Comedy 28.2917960 16.5468923 ## Drama-Comedy 12.8231797 4.2052020 ## Horror-Comedy -12.1402597 -29.1801306 ## Musical & Performing Arts-Comedy 25.1454545 5.5767332 ## Mystery & Suspense-Comedy 3.1210643 -8.6238394 ## Other-Comedy 7.8727273 -10.9282921 ## Science Fiction & Fantasy-Comedy 14.8954545 -14.5828011 ## Drama-Documentary -15.4686163 -25.1838719 ## Horror-Documentary -40.4320557 -58.0523239 ## Musical & Performing Arts-Documentary -3.1463415 -23.2224842 ## Mystery & Suspense-Documentary -25.1707317 -37.7429000 ## Other-Documentary -20.4190687 -39.7476740 ## Science Fiction & Fantasy-Documentary -13.3963415 -43.2138566 ## Horror-Drama -24.9634394 -40.6733289 ## Musical & Performing Arts-Drama 12.3222749 -6.0999477 ## Mystery & Suspense-Drama -9.7021154 -19.4173709 ## Other-Drama -4.9504524 -22.5550472 ## Science Fiction & Fantasy-Drama 2.0722749 -26.6576918 ## Musical & Performing Arts-Horror 37.2857143 13.7173893 ## Mystery & Suspense-Horror 15.2613240 -2.3589442 ## Other-Horror 20.0129870 -2.9219082 ## Science Fiction & Fantasy-Horror 27.0357143 -5.2365439 ## Mystery & Suspense-Musical & Performing Arts -22.0243902 -42.1005330 ## Other-Musical & Performing Arts -17.2727273 -42.1441381 ## Science Fiction & Fantasy-Musical & Performing Arts -10.2500000 -43.9260494 ## Other-Mystery & Suspense 4.7516630 -14.5769423 ## Science Fiction & Fantasy-Mystery & Suspense 11.7743902 -18.0431249 ## Science Fiction & Fantasy-Other 7.0227273 -26.2130934

24

## upr p adj ## Animation-Action & Adventure 30.85999139 0.8982375 ## Art House & International-Action & Adventure 30.43240150 0.8339212 ## Comedy-Action & Adventure 12.95868056 0.9999988 ## Documentary-Action & Adventure 42.09326229 0.0000000 ## Drama-Action & Adventure 23.70249884 0.0000736 ## Horror-Action & Adventure 6.77189060 0.6621600 ## Musical & Performing Arts-Action & Adventure 46.53240150 0.0009942 ## Mystery & Suspense-Action & Adventure 16.92253058 0.9827584 ## Other-Action & Adventure 28.50689184 0.8935579 ## Science Fiction & Fantasy-Action & Adventure 46.06793083 0.7922139 ## Art House & International-Animation 26.60982910 1.0000000 ## Comedy-Animation 11.87805836 0.9575830 ## Documentary-Animation 40.65551081 0.0868992 ## Drama-Animation 23.60802472 0.9997842 ## Horror-Animation 3.58996395 0.1780939 ## Musical & Performing Arts-Animation 42.70982910 0.6150129 ## Mystery & Suspense-Animation 15.48477911 0.9989613 ## Other-Animation 24.86777498 1.0000000 ## Science Fiction & Fantasy-Animation 40.51196908 0.9999546 ## Comedy-Art House & International 10.52326682 0.9208281 ## Documentary-Art House & International 39.32248424 0.0736899 ## Drama-Art House & International 22.19994769 0.9998788 ## Horror-Art House & International 2.38261075 0.1236044 ## Musical & Performing Arts-Art House & International 41.55670054 0.6162797 ## Mystery & Suspense-Art House & International 14.15175253 0.9970638 ## Other-Art House & International 23.69868352 1.0000000 ## Science Fiction & Fantasy-Art House & International 39.52604941 0.9999739 ## Documentary-Comedy 40.03669968 0.0000000 ## Drama-Comedy 21.44115728 0.0001071 ## Horror-Comedy 4.89961111 0.4320268 ## Musical & Performing Arts-Comedy 44.71417591 0.0019023 ## Mystery & Suspense-Comedy 14.86596798 0.9987871 ## Other-Comedy 26.67374662 0.9582176 ## Science Fiction & Fantasy-Comedy 44.37371022 0.8664846 ## Drama-Documentary -5.75336076 0.0000209 ## Horror-Documentary -22.81178755 0.0000000 ## Musical & Performing Arts-Documentary 16.92980131 0.9999901 ## Mystery & Suspense-Documentary -12.59856343 0.0000000 ## Other-Documentary -1.09046347 0.0283571 ## Science Fiction & Fantasy-Documentary 16.42117370 0.9337288 ## Horror-Drama -9.25354993 0.0000220 ## Musical & Performing Arts-Drama 30.74449746 0.5316015 ## Mystery & Suspense-Drama 0.01314022 0.0506577 ## Other-Drama 12.65414239 0.9980375 ## Science Fiction & Fantasy-Drama 30.80224158 1.0000000 ## Musical & Performing Arts-Horror 60.85403932 0.0000246 ## Mystery & Suspense-Horror 32.88159224 0.1604488 ## Other-Horror 42.94788220 0.1525896 ## Science Fiction & Fantasy-Horror 59.30797244 0.1983985 ## Mystery & Suspense-Musical & Performing Arts -1.94824747 0.0183687 ## Other-Musical & Performing Arts 7.59868352 0.4725291 ## Science Fiction & Fantasy-Musical & Performing Arts 23.42604941 0.9962125 ## Other-Mystery & Suspense 24.08026823 0.9993832

25

## Science Fiction & Fantasy-Mystery & Suspense 41.59190541 0.9721543

## Science Fiction & Fantasy-Other 40.25854796 0.9998407

plot(TukeyHSD(mov.aov), las=1, cex.axis=0.7)

**95% family−wise confidence level**

Animation−Action & Adventure

ocumentary−Action & Adventure

forming Arts−Action & Adventure

n & Fantasy−Action & Adventure

Documentary−Animation

cal & Performing Arts−Animation

nce Fiction & Fantasy−Animation

rama−Art House & International

pense−Art House & International

Documentary−Comedy

sical & Performing Arts−Comedy

ence Fiction & Fantasy−Comedy

& Performing Arts−Documentary

Fiction & Fantasy−Documentary

Mystery & Suspense−Drama

usical & Performing Arts−Horror

cience Fiction & Fantasy−Horror

ntasy−Musical & Performing Arts

Science Fiction & Fantasy−Other

−60 −40 −20 0 20 40 60

Differences in mean levels of genre

*# pairwise.t.test(mov\_train$audience\_score, mov\_train$genre, p.adjust.method = "bonferroni")*

Now, let us proceed with the modeling.

full <- lm(imdb\_rating ~ ., data = mov\_train)

null <- lm(imdb\_rating ~ 1, data = mov\_train)

X <- model.matrix(full)[,-1]

*# both BIC models with forward/backward steps*

both\_BIC = step(null, list(lower = ~ 1, upper = formula(full)), trace = F,

direction = 'both', k = log(nrow(X)))

both\_backward\_BIC = step(full, list( upper = null), trace = F,

direction = 'both', k = log(nrow(X)))

*# both BIC models with forward/backward steps*

both\_AIC = step(null, list(lower = ~ 1, upper = formula(full)), trace = F,

direction = 'both', k = 2)

26

both\_backward\_AIC = step(full, list( upper = null), trace = F,

direction = 'both', k = 2)

MSE.BIC.forward = mean((predict(both\_BIC, mov\_test) - mov\_test$imdb\_rating)ˆ2) MSE.BIC.backward = mean((predict(both\_backward\_BIC, mov\_test) - mov\_test$imdb\_rating)ˆ2)

MSE.AIC.forward = mean((predict(both\_AIC, mov\_test) - mov\_test$imdb\_rating)ˆ2) MSE.AIC.backward = mean((predict(both\_backward\_AIC, mov\_test) - mov\_test$imdb\_rating)ˆ2)

data.frame(MSE.BIC.forward, MSE.BIC.backward, MSE.AIC.forward, MSE.AIC.backward)

## MSE.BIC.forward MSE.BIC.backward MSE.AIC.forward MSE.AIC.backward ## 1 0.1846557 0.1786505 0.1765288 0.1765288

formula(both\_BIC) *# imdb\_num\_votes, critics\_rating*

## imdb\_rating ~ audience\_score + critics\_score + runtime + audience\_rating formula(both\_backward\_BIC)

## imdb\_rating ~ runtime + imdb\_num\_votes + critics\_rating + critics\_score + ## audience\_rating + audience\_score

formula(both\_AIC)

## imdb\_rating ~ audience\_score + critics\_score + genre + imdb\_num\_votes + ## audience\_rating + critics\_rating + runtime

formula(both\_backward\_AIC)

## imdb\_rating ~ genre + runtime + imdb\_num\_votes + critics\_rating + ## critics\_score + audience\_rating + audience\_score

We select the both\_AIC model with least MSE. Now, we shall proceed to diagnose the model to improve it. From the diagnostic plots, we could observe that linearity assumption of the model doesn’t hold and a possible heteroscedasticity is observed.

par(mfrow = c(2,2))

plot(both\_AIC)

27

s

la

u

di

s

e

R

s

la

u

di

s

e

r

d e

z

i

dr

a

d n

a

tS

0

2 −

5 .

1

0 .

0

Residuals vs Fitted

242 83 154

4 5 6 7 8 9 Fitted values

Scale−Location

242 83 154

4 5 6 7 8 9 Fitted values

s

la

u

di

s

e

r

d

e

z

i

dr

a

dn

a

tS

s

la

u

di

s

e

r

d

e

z

i

dr

a

dn

a

tS

Normal Q−Q

0

4

1542283

−

−3 −2 −1 0 1 2 3 Theoretical Quantiles

Residuals vs Leverage

23

2

2

345

Cook's distance 0.5

−

86

6

−

0.00 0.10 0.20

Leverage

Let us apply the transformation to remove heteroscedasticity and linearize the model.

par(mfrow = c(1,2))

boxcox(both\_AIC, plotit=T)

boxcox(both\_AIC, plotit=T, lambda=seq(1.5,3,by=0.05))

28

d o

o

hil

e

ki

L

−

g

o

l

0 02

−

0 04

−

0 06

−

0 08

−

95%

−2 −1 0 1 2 λ

d o

o

hil

e

ki

L

−

g

o

l

0 9

−

0 01

−

0 11

−

0 21

−

0 31

−

95%

1.5 2.0 2.5 3.0 λ

As per boxcox plot, we shall consider a transformation of 10/4 (= 2*.*5) for the response variable.

*# transformed model*

mod1 <- lm(imdb\_rating ˆ 2.5 ~ audience\_score + critics\_score + genre + imdb\_num\_votes + audience\_rating + critics\_rating + runtime, data = mov\_train)

*#Diagnostic Plots*

par(mfrow = c(2,2))

plot(mod1)

29

s

la

u

di

s

e

R

s

la

u

di

s

e

r

d e

z

i

dr

a

d n

a

tS

0

0 8

−

5 .

1

0 .

0

Residuals vs Fitted

15483 86

50 100 150 200 Fitted values

Scale−Location

15483 86

50 100 150 200 Fitted values

s

la

u

di

s

e

r

d

e

z

i

dr

a

dn

a

tS

s

la

u

di

s

e

r

d

e

z

i

dr

a

dn

a

tS

Normal Q−Q

0

154 8386

4

−

−3 −2 −1 0 1 2 3 Theoretical Quantiles

Residuals vs Leverage

23

0

405 107

Cook's distance0.5

4

−

0.00 0.10 0.20

Leverage

Let us now check for collinearity. As per below, here we observe that, VIF *<* 10 & *κp <* 15. Hence, collinearity is ok.

*# VIF*

car::vif(mod1)

## GVIF Df GVIF^(1/(2\*Df))

## audience\_score 5.895377 1 2.428040

## critics\_score 6.117419 1 2.473342

## genre 1.820407 10 1.030406

## imdb\_num\_votes 1.615459 1 1.271007

## audience\_rating 4.148237 1 2.036722

## critics\_rating 5.393049 2 1.523907

## runtime 1.339764 1 1.157482

*# condition index*

X = model.matrix(mod1)[,-1]

R = cor(X)

ev = eigen(R)$val

sqrt(ev[1]\*evˆ(-1))

## [1] 1.000000 1.450132 1.663677 1.714164 1.858079 1.880353 1.882544 1.893225 ## [9] 1.904681 1.974370 1.998278 2.334482 2.688485 3.223667 5.295403 5.507654 ## [17] 6.975310

30

Let us check for autocorrelation even though we could skip it. From the graph, we could see that no correlation exists.

acf(resid(mod1))

**Series resid(mod1)**

0

.

1

8

.

0

6

.

0

F

C

4

.

A

0

2

.

0

0

.

0

0 5 10 15 20 25

Lag

Now, our modelling step is complete.

31

**Prediction**

First, let us test the prediction with one of the existing row in the mov\_test dataset. In this case, let us consider the movie Locke. The dataframe is created with the values relevant to Locke movie.

new\_df = data.frame(audience\_score = 71 ,

critics\_score = 91,

genre = 'Mystery & Suspense',

imdb\_num\_votes = 82851,

audience\_rating = 'Upright',

critics\_rating = 'Certified Fresh',

runtime = 85)

Based on the prediction, we could observe that the fitted value is 7.121779 which is ~ equal to the original value of 7.1. Hence we can say that the model prediction is working properly.

predict(mod1 , newdata = new\_df, interval = 'prediction')ˆ(1/2.5)

## fit lwr upr

## 1 7.121779 6.422304 7.730952

Now let us predict for a movie not from the datasets. In this case, let us consider the movie Dune. The dataframe is created with the values relevant to Dune movie.

new\_df1 = data.frame(audience\_score = 90,

critics\_score = 83,

genre = 'Science Fiction & Fantasy',

imdb\_num\_votes = 390470,

audience\_rating = 'Upright',

critics\_rating = 'Certified Fresh',

runtime = 155)

Based on the prediction, we could observe that the fitted value is 7.9 which is very close to the original value of 8.2. Hence we can say that the model prediction is working properly.

predict(mod1 , newdata = new\_df1, interval = 'prediction')ˆ(1/2.5)

## fit lwr upr

## 1 7.921116 7.273341 8.497836

Now let us predict for a movie not from the datasets. In this case, let us consider the movie RUN. The dataframe is created with the values relevant to RUN movie.

new\_df2 = data.frame(audience\_score = 74,

critics\_score = 88,

genre = 'Mystery & Suspense',

imdb\_num\_votes = 62456,

audience\_rating = 'Upright',

critics\_rating = 'Certified Fresh',

runtime = 90)

32

Based on the prediction, we could observe that the fitted value is 7.1 which is very close to the original value of 6.7. Hence we can say that the model prediction is working properly.

predict(mod1 , newdata = new\_df2, interval = 'prediction')ˆ(1/2.5)

## fit lwr upr

## 1 7.170732 6.480396 7.773505

**Conclusion**

While researching on the topics mentioned in the report, first we found many insignificant variables which were removed from the dataset in the pre-processing step. We have also removed the rows having ‘NA’ values. For the first research question, we used two sample t-test and determined that there is no association between Oscar winning actor or actress with the IMDB rating of the movies.

For the second research question we have performed TukeyHSD test which suggests that there is a difference between mean audience score between genres. We have found few such pairs in the results.

For the third research question we found that audience\_score, critics\_score, genre, imdb\_num\_votes, au dience\_rating, critics\_rating and runtime are associated and can be used to predict the rating of a movie on IMBD. We have used both\_AIC model with least MSE. We have tested our model on the given dataset values as well as with the values apart from the given dataset and we obtain a good accuracy which falls within the 95% confidence interval.

For the model built, we didn’t consider the interaction terms as well as different model approaches like lasso and ridge regression. In the future studies, these could be considered while model building and see if there is an improvement in prediction.

One of the shortcomings of the dataset is that the data available is related to the movies of USA and hence for the movies outside of the USA, there could be possible bias if this model is used.

33