Movie Data Analysis

Amal Alappat, Lingeshwaran.C, Jayalakshmi,Thasneem Therodam Kandi

September 16, 2021

Import data

```
movieData<-read.csv("/Users/aliayyally/Documents/Data Analytics Training/Python/Da
taFiles/movies.csv")
head(movieData)
```

```
##                                             name rating     genre year
## 1                                    The Shining      R     Drama 1980
## 2                                The Blue Lagoon      R Adventure 1980
## 3 Star Wars: Episode V - The Empire Strikes Back     PG    Action 1980
## 4                                      Airplane!     PG    Comedy 1980
## 5                                     Caddyshack      R    Comedy 1980
## 6                                Friday the 13th      R    Horror 1980
##                         released score   votes           director
## 1 June 13, 1980 (United States)   8.4  927000    Stanley Kubrick
## 2  July 2, 1980 (United States)   5.8   65000     Randal Kleiser
## 3 June 20, 1980 (United States)   8.7 1200000     Irvin Kershner
## 4  July 2, 1980 (United States)   7.7  221000       Jim Abrahams
## 5 July 25, 1980 (United States)   7.3  108000       Harold Ramis
## 6   May 9, 1980 (United States)   6.4  123000 Sean S. Cunningham
##                     writer           star        country   budget     gross
## 1            Stephen King Jack Nicholson United Kingdom 19000000  46998772
## 2 Henry De Vere Stacpoole Brooke Shields  United States  4500000  58853106
## 3          Leigh Brackett    Mark Hamill  United States 18000000 538375067
## 4            Jim Abrahams    Robert Hays  United States  3500000  83453539
## 5      Brian Doyle-Murray    Chevy Chase  United States  6000000  39846344
## 6           Victor Miller   Betsy Palmer  United States   550000  39754601
##              company runtime
## 1       Warner Bros.     146
## 2  Columbia Pictures     104
## 3          Lucasfilm     124
## 4 Paramount Pictures      88
## 5     Orion Pictures      98
## 6 Paramount Pictures      95
```

```
dim(movieData)
```

```
## [1] 7668   15
```

```
str(movieData)
```

```
## 'data.frame':    7668 obs. of  15 variables:
##  $ name    : chr  "The Shining" "The Blue Lagoon" "Star Wars: Episode V - The E
mpire Strikes Back" "Airplane!" ...
##  $ rating  : chr  "R" "R" "PG" "PG" ...
##  $ genre   : chr  "Drama" "Adventure" "Action" "Comedy" ...
##  $ year    : int  1980 1980 1980 1980 1980 1980 1980 1980 1980 1980 ...
##  $ released: chr  "June 13, 1980 (United States)" "July 2, 1980 (United States)
" "June 20, 1980 (United States)" "July 2, 1980 (United States)" ...
##  $ score   : num  8.4 5.8 8.7 7.7 7.3 6.4 7.9 8.2 6.8 7 ...
##  $ votes   : int  927000 65000 1200000 221000 108000 123000 188000 330000 10100
0 10000 ...
##  $ director: chr  "Stanley Kubrick" "Randal Kleiser" "Irvin Kershner" "Jim Abra
hams" ...
##  $ writer  : chr  "Stephen King" "Henry De Vere Stacpoole" "Leigh Brackett" "Ji
m Abrahams" ...
##  $ star    : chr  "Jack Nicholson" "Brooke Shields" "Mark Hamill" "Robert Hays"
...
##  $ country : chr  "United Kingdom" "United States" "United States" "United Stat
es" ...
##  $ budget  : int  19000000 4500000 18000000 3500000 6000000 550000 27000000 180
00000 54000000 10000000 ...
##  $ gross   : num  4.70e+07 5.89e+07 5.38e+08 8.35e+07 3.98e+07 ...
##  $ company : chr  "Warner Bros." "Columbia Pictures" "Lucasfilm" "Paramount Pic
tures" ...
##  $ runtime : int  146 104 124 88 98 95 133 129 127 100 ...
```

```
colnames(movieData)
```

```
##  [1] "name"     "rating"   "genre"    "year"     "released" "score"
##  [7] "votes"    "director" "writer"   "star"     "country"  "budget"
## [13] "gross"    "company"  "runtime"
```

```
summary(movieData)
```

```
##      name               rating             genre                year
##  Length:7668        Length:7668        Length:7668        Min.   :1980
##  Class :character   Class :character   Class :character   1st Qu.:1991
##  Mode  :character   Mode  :character   Mode  :character   Median :2000
##                                                           Mean   :2000
##                                                           3rd Qu.:2010
##                                                           Max.   :2020
##
##    released             score           votes            director
##  Length:7668        Min.   :1.90   Min.   :      7   Length:7668
##  Class :character   1st Qu.:5.80   1st Qu.:   9100   Class :character
##  Mode  :character   Median :6.50   Median :  33000   Mode  :character
##                     Mean   :6.39   Mean   :  88108
##                     3rd Qu.:7.10   3rd Qu.:  93000
##                     Max.   :9.30   Max.   :2400000
##                     NA's   :3      NA's   :3
##     writer             star             country              budget
##  Length:7668        Length:7668        Length:7668        Min.   :     3000
##  Class :character   Class :character   Class :character   1st Qu.: 10000000
##  Mode  :character   Mode  :character   Mode  :character   Median : 20500000
##                                                           Mean   : 35589876
##                                                           3rd Qu.: 45000000
##                                                           Max.   :356000000
##                                                           NA's   :2171
##      gross              company             runtime
##  Min.   :3.090e+02   Length:7668        Min.   : 55.0
##  1st Qu.:4.532e+06   Class :character   1st Qu.: 95.0
##  Median :2.021e+07   Mode  :character   Median :104.0
##  Mean   :7.850e+07                      Mean   :107.3
##  3rd Qu.:7.602e+07                      3rd Qu.:116.0
##  Max.   :2.847e+09                      Max.   :366.0
##  NA's   :189                            NA's   :4
```

Import required libraries

```
library(scales)
library(ggplot2)
library(gridExtra)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following object is masked from 'package:gridExtra':
##
##     combine
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(rpart)
#install.packages("rpart.plot")
library(rpart.plot)
library(corrplot)
```

```
## corrplot 0.90 loaded
```

Checking the missing values

```
sum(is.na(movieData))
```

```
## [1] 2370
```

There is 2370 null values

Lets check column wise null values

```
colSums(is.na(movieData))
```

```
##     name   rating    genre     year released    score    votes director
##        0        0        0        0        0        3        3        0
##   writer     star  country   budget    gross  company  runtime
##        0        0        0     2171      189        0        4
```

Proportion of missing values

```
mean(is.na(movieData))
```

```
## [1] 0.02060511
```

Since the the proportion of null values is very small we can simply delete the null data

```
movieData <- na.omit(movieData)
sum(is.na(movieData))
```

```
## [1] 0
```

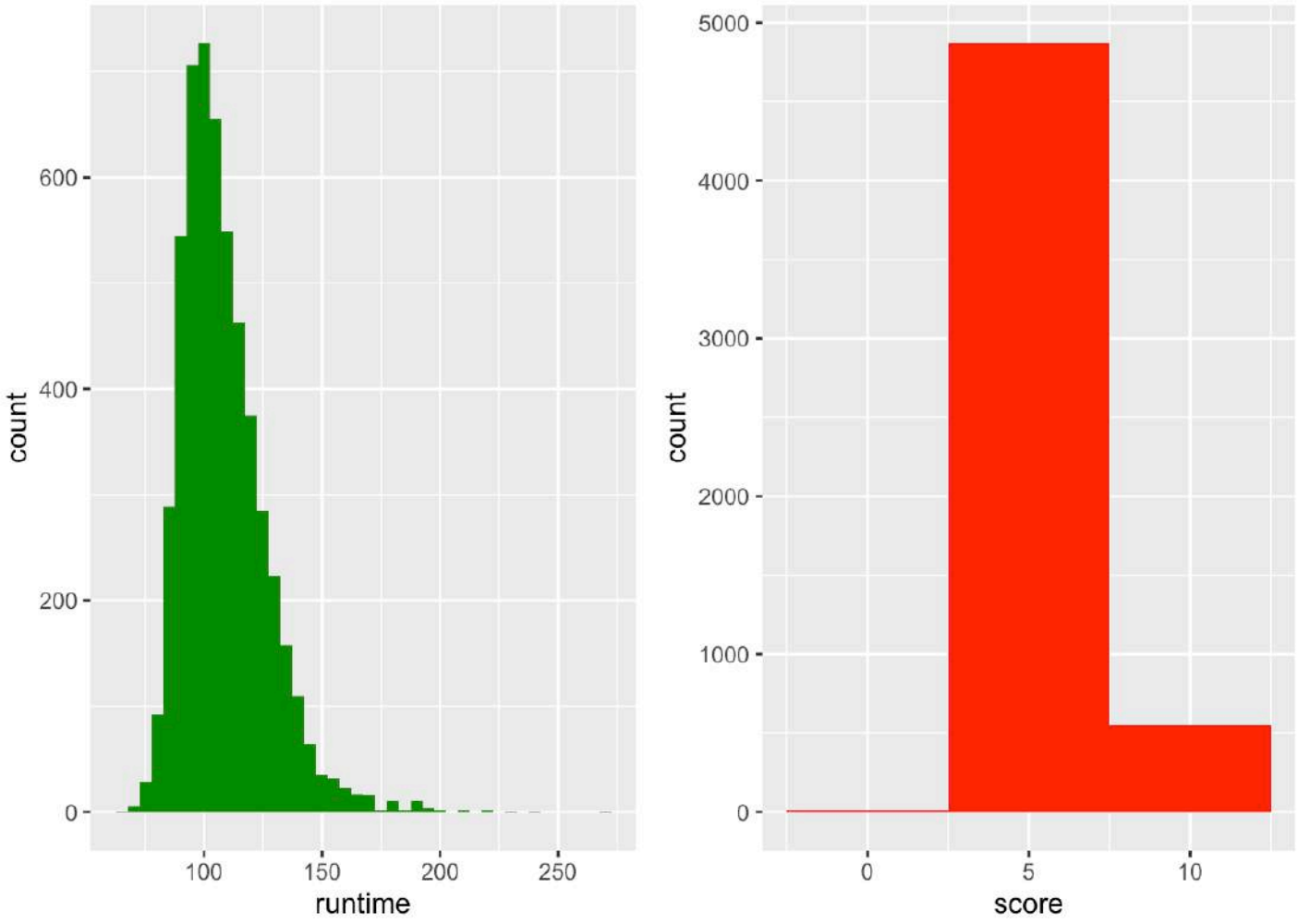
Now we have data without missing values, lest check dimension again

```
dim(movieData)
```

```
## [1] 5435   15
```

Visualizing duration and score using histogram

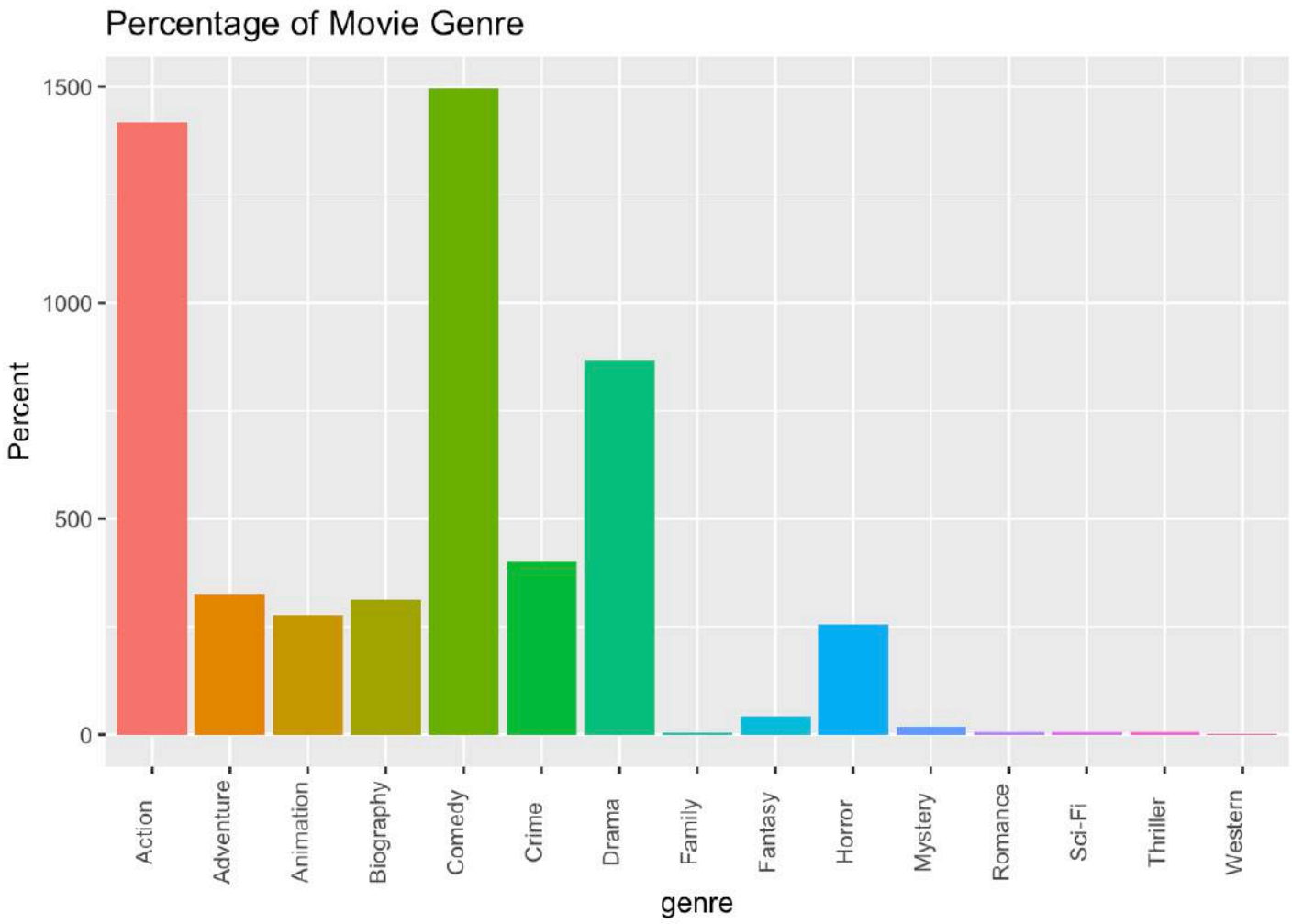
```
options(repr.plot.width=6,repr.plot.height = 4)
g1 <- ggplot(movieData,aes(x=runtime))+geom_histogram(binwidth = 5,fill="green4")
g2 <- ggplot(movieData,aes(x=score))+geom_histogram(binwidth = 5,fill="red")
grid.arrange(g1,g2,nrow=1,ncol=2)
```



Percentage of Movie Genre

```
ggplot(movieData,aes(x=genre,fill=genre))+geom_histogram(stat="count",binwidth = 1
)+
  theme(axis.text.x = element_text(angle = 90,hjust = .5,vjust = 0),legend.positio
n = "none")+
  labs(y="Percent",title="Percentage of Movie Genre")
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```

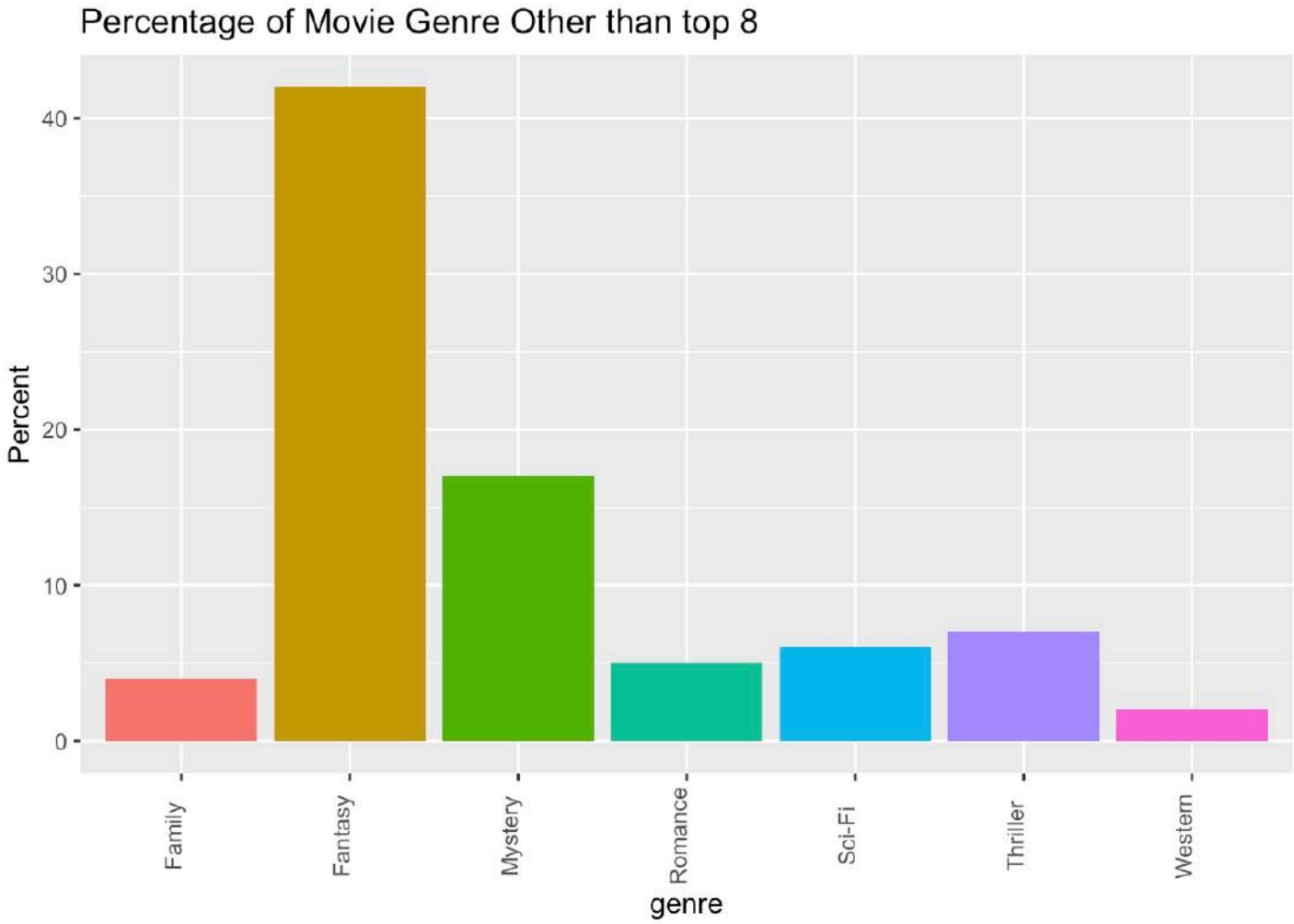


Comedies, Action and Drama are highest genre of movies. Also, Adventure, Animation, Biography, Crime and Horror has considerable count of movies

Now lets check the distribution of other Genre excluding above 8 genres

```
movieData %>% filter(!(genre %in% c("Action","Comedy","Drama","Adventure","Animati
on","Biography","Horror","Crime"))) %>%
  ggplot(aes(x=genre,fill=genre))+geom_histogram(stat="count",binwidth = 1)+
  theme(axis.text.x = element_text(angle = 90,hjust = .5,vjust = 0),legend.positio
n = "none")+
  labs(y="Percent",title="Percentage of Movie Genre Other than top 8")
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```



Now, lets check the movies by countries

To understand how many movies produced by countries every year, lets create a data frame of year and countries with their movie count

```
year_movies <- movieData %>%
  group_by(year,country) %>%
  summarize(movie_count=n())%>%
  filter(movie_count>=5)
```

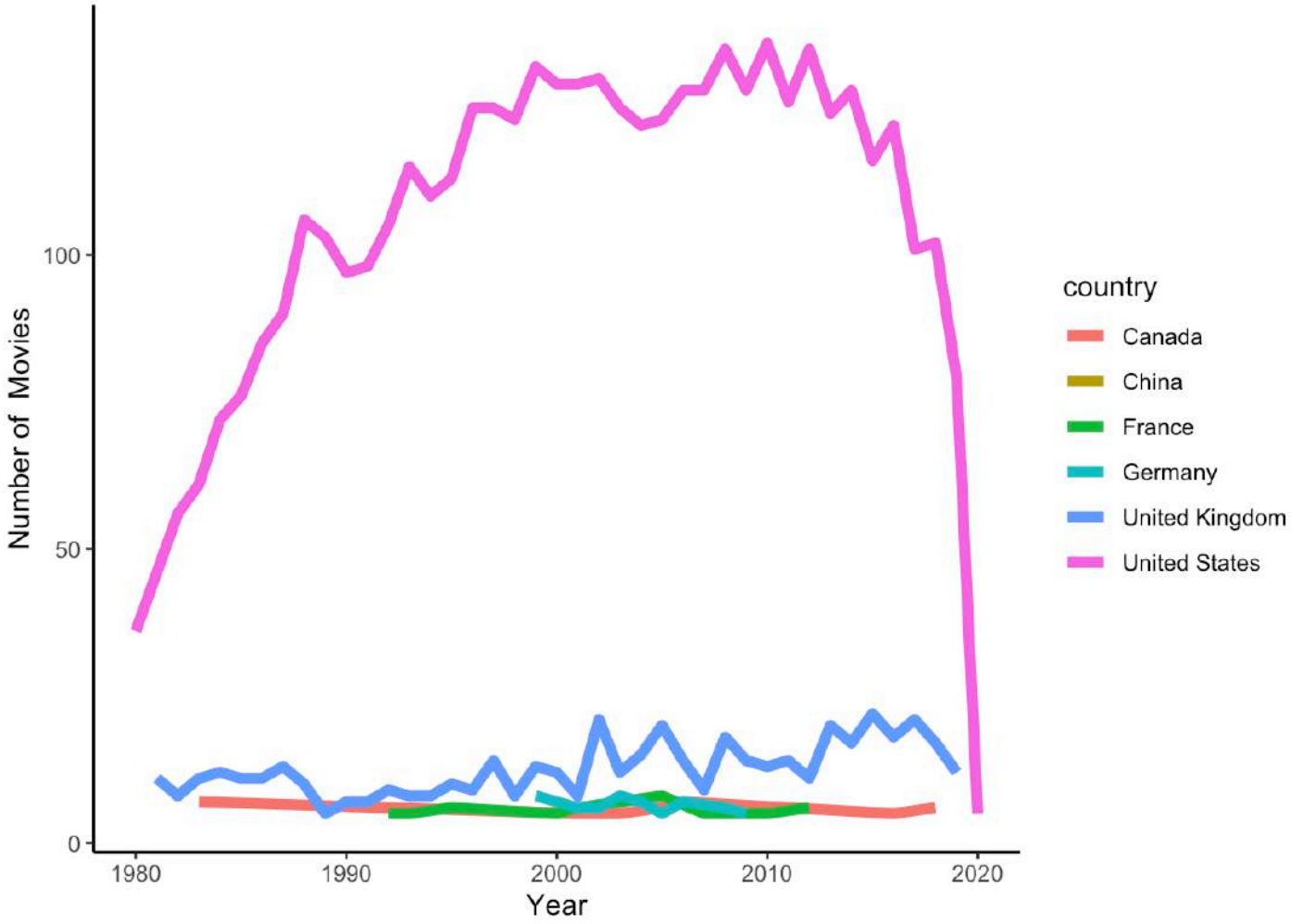
```
## `summarise()` has grouped output by 'year'. You can override using the `.groups
` argument.
```

```
head(year_movies)
```

```
## # A tibble: 6 × 3
## # Groups:   year [4]
##    year country        movie_count
##   <int> <chr>                <int>
## 1  1980 United States           36
## 2  1981 United Kingdom          11
## 3  1981 United States           46
## 4  1982 United Kingdom           8
## 5  1982 United States           56
## 6  1983 Canada                   7
```

Plot the movie number through time and color them with different country

```
options(repr.plot.width=4,repr.plot.height = 4)
ggplot(year_movies,aes(x=year,y=movie_count,colour=country))+
  geom_line(size= 2)+xlab("Year")+ylab("Number of  Movies")+theme_classic()
```



United states movie production was drastically increased (upward) during 1980 and 1990 and also they are the largest players currently

Top ten directors in terms of Score

```
best_director<- movieData %>% group_by(director) %>%
  summarise(mean=mean(score))  %>%
  top_n(10) %>%
  arrange(desc(mean))
```

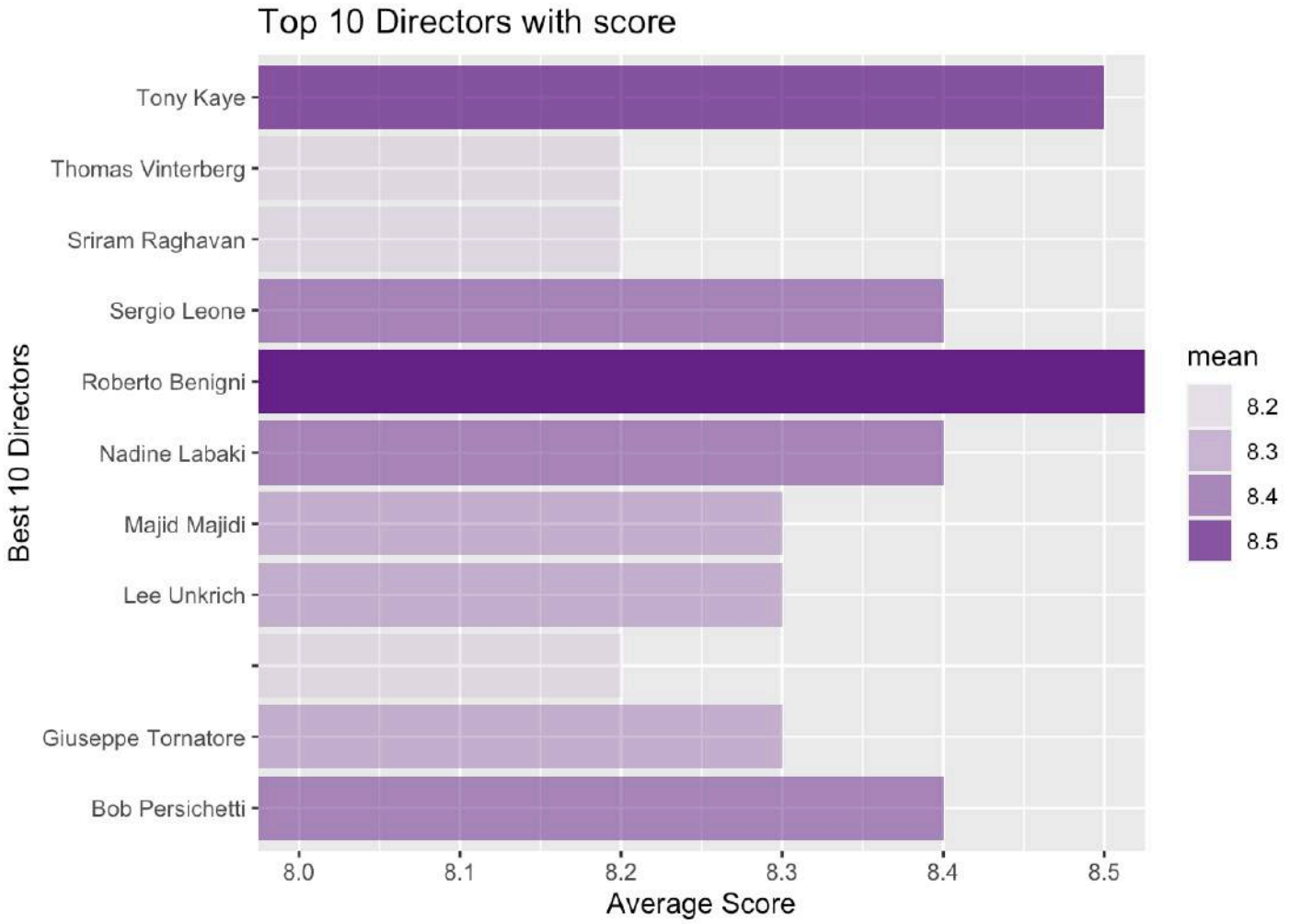
```
## Selecting by mean
```

```
best_director
```

```
## # A tibble: 11 × 2
##    director                   mean
##    <chr>                     <dbl>
##  1 "Roberto Benigni"           8.6
##  2 "Tony Kaye"                 8.5
##  3 "Bob Persichetti"           8.4
##  4 "Nadine Labaki"             8.4
##  5 "Sergio Leone"              8.4
##  6 "Giuseppe Tornatore"        8.3
##  7 "Lee Unkrich"               8.3
##  8 "Majid Majidi"              8.3
##  9 "Juan Jos\xe9 Campanella"   8.2
## 10 "Sriram Raghavan"           8.2
## 11 "Thomas Vinterberg"         8.2
```

Plotting the top 10 directors

```
ggplot(best_director,aes(x=director, y= mean,alpha=mean))+
  geom_bar(stat = "identity",fill="darkorchid4")+labs(x="Best 10 Directors",y="Ave
rage Score")+
  ggtitle("Top 10 Directors with score")+coord_flip(ylim = c(8,8.5))
```



Listing the top 10 actors based on the average movie score

```
best_actor<-movieData %>%
  group_by(star)%>%
  summarise(mean= mean(score))%>%
  top_n(10)%>%
  arrange(desc(mean))
```

```
## Selecting by mean
```

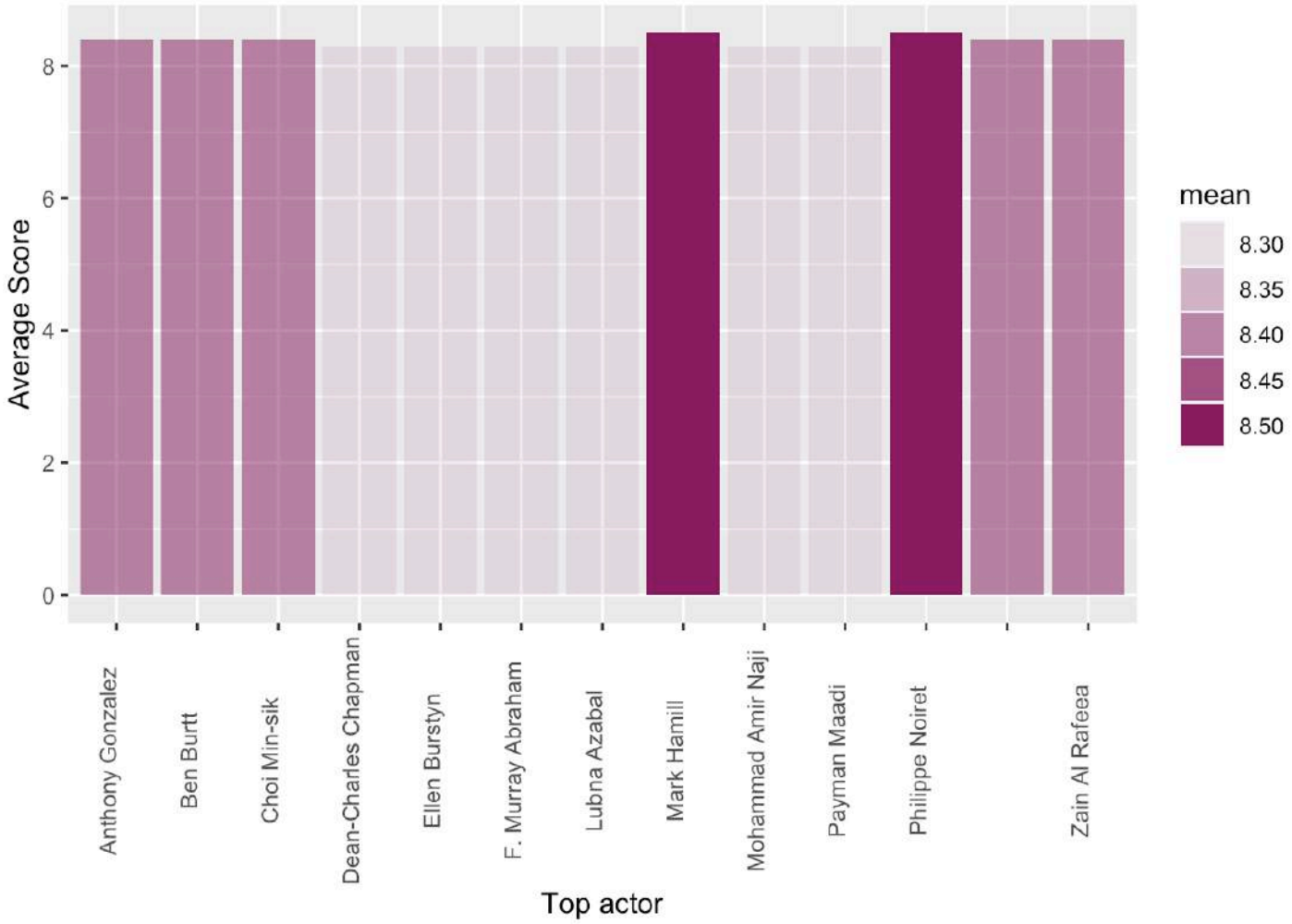
```
best_actor
```

```
## # A tibble: 13 × 2
##    star                    mean
##    <chr>                  <dbl>
##  1 "Mark Hamill"            8.5
##  2 "Philippe Noiret"        8.5
##  3 "Anthony Gonzalez"       8.4
##  4 "Ben Burtt"              8.4
##  5 "Choi Min-sik"           8.4
##  6 "Ulrich M\xfche"         8.4
##  7 "Zain Al Rafeea"         8.4
##  8 "Dean-Charles Chapman"   8.3
##  9 "Ellen Burstyn"          8.3
## 10 "F. Murray Abraham"      8.3
## 11 "Lubna Azabal"           8.3
## 12 "Mohammad Amir Naji"     8.3
## 13 "Payman Maadi"           8.3
```

```
ggtitle("Top 10 actor with average score")+coord_flip(ylim=c(8.0,9.0))
```

```
## NULL
```

```
ggplot(best_actor, aes(x = star, y = mean, alpha = mean))+
  theme(axis.text.x = element_text(angle = 90,hjust = .5,vjust = 0))+
  geom_bar(stat = "identity",fill="maroon4") + labs(x = "Top actor", y = "Average
Score")
```



Top movie country with average score

```
best_moviecountry<-movieData %>%
  group_by(country)%>%
  summarise(mean= mean(score))%>%
  top_n(10)%>%
  arrange(desc(mean))
```

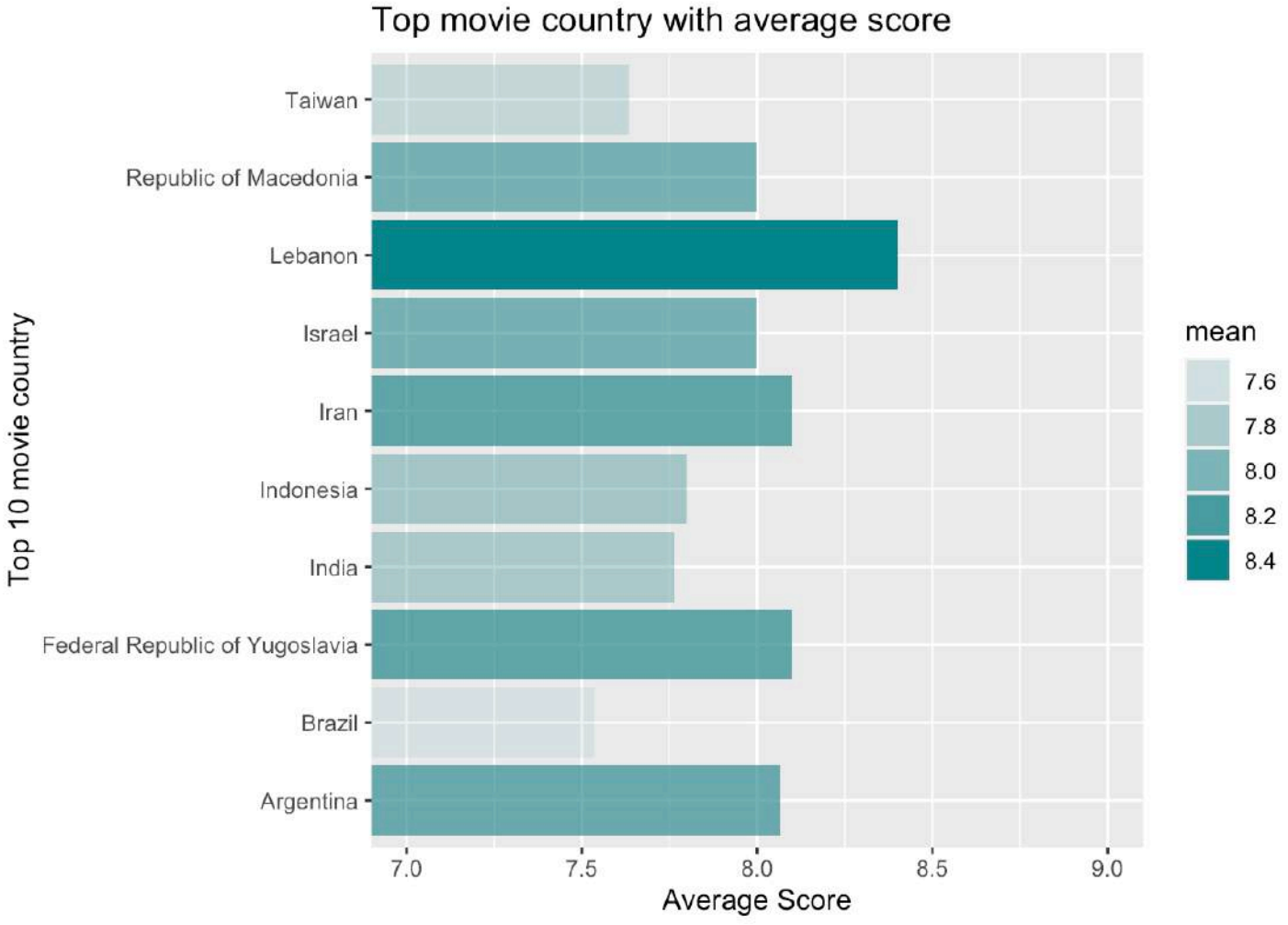
```
## Selecting by mean
```

```
best_moviecountry
```

```
## # A tibble: 10 × 2
##    country                          mean
##    <chr>                           <dbl>
##  1 Lebanon                          8.4
##  2 Iran                             8.1
##  3 Federal Republic of Yugoslavia   8.1
##  4 Argentina                        8.07
##  5 Israel                           8
##  6 Republic of Macedonia            8
##  7 Indonesia                        7.8
##  8 India                            7.76
##  9 Taiwan                           7.63
## 10 Brazil                           7.53
```

Plotting the top movie country with average score

```
moviecountry<-ggplot(best_moviecountry, aes(x = country, y = mean, alpha = mean))+
  geom_bar(stat = "identity",fill = "turquoise4") + labs(x = "Top 10 movie country
", y = "Average Score") +
  ggtitle("Top movie country with average score")+coord_flip(ylim=c(7.0,9.0))
moviecountry
```

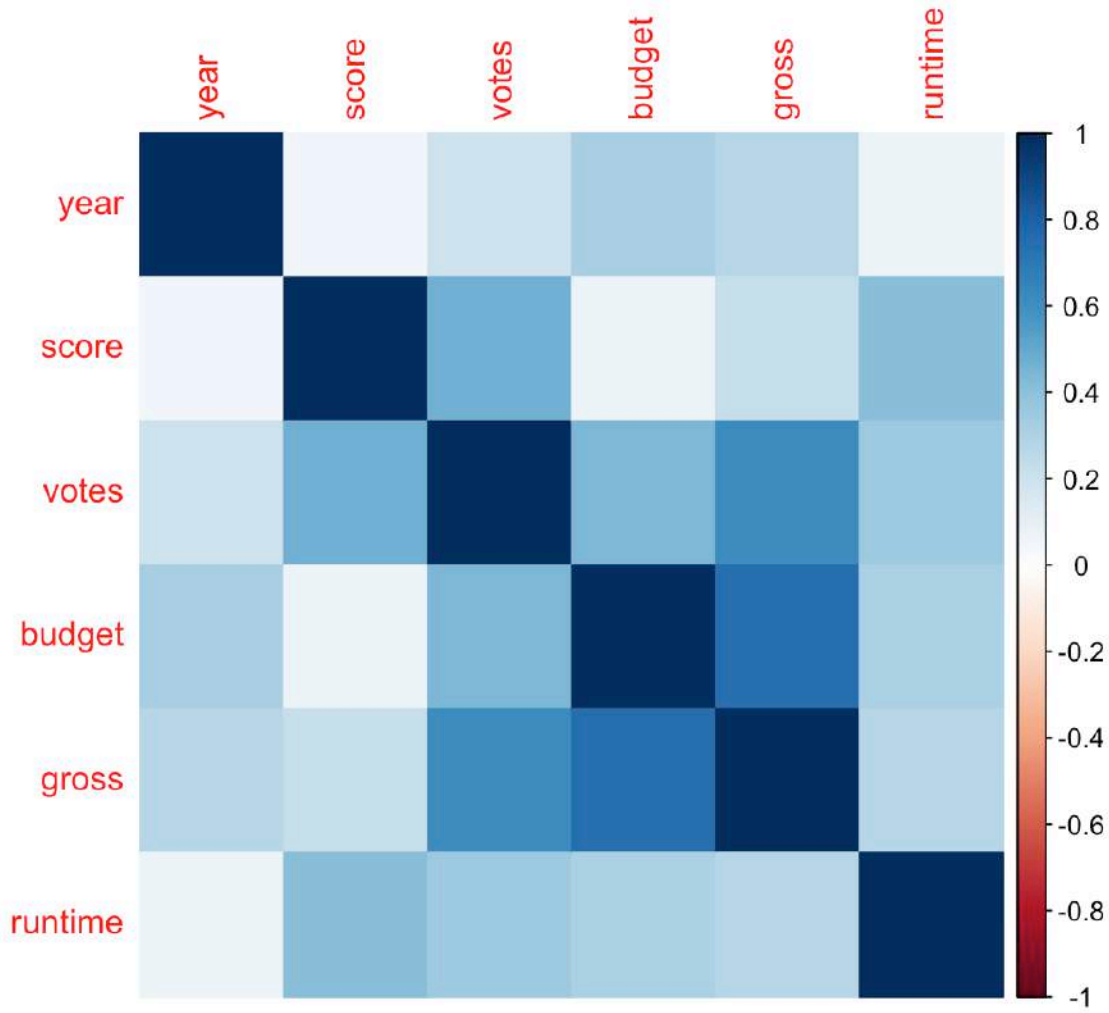


Relationship between variables

```
numeric_col <- sapply(movieData, is.numeric)
movie_numeric<- movieData[,numeric_col]
```

create correlation matrix

```
Correlation<-cor(movie_numeric)
corrplot(Correlation, method = "color")
```




```
Correlation
```

```
##                year      score     votes     budget     gross    runtime
## year    1.00000000 0.05539076 0.2058518 0.32779259 0.2743544 0.07420276
## score   0.05539076 1.00000000 0.4737885 0.07182096 0.2220997 0.41457958
## votes   0.20585175 0.47378855 1.0000000 0.44003510 0.6148948 0.35243685
## budget  0.32779259 0.07182096 0.4400351 1.00000000 0.7404100 0.31859490
## gross   0.27435438 0.22209969 0.6148948 0.74040999 1.0000000 0.27559627
## runtime 0.07420276 0.41457958 0.3524368 0.31859490 0.2755963 1.00000000
```

Here from correlation, we noticed that, vote, gross and runtime are highly correlated with

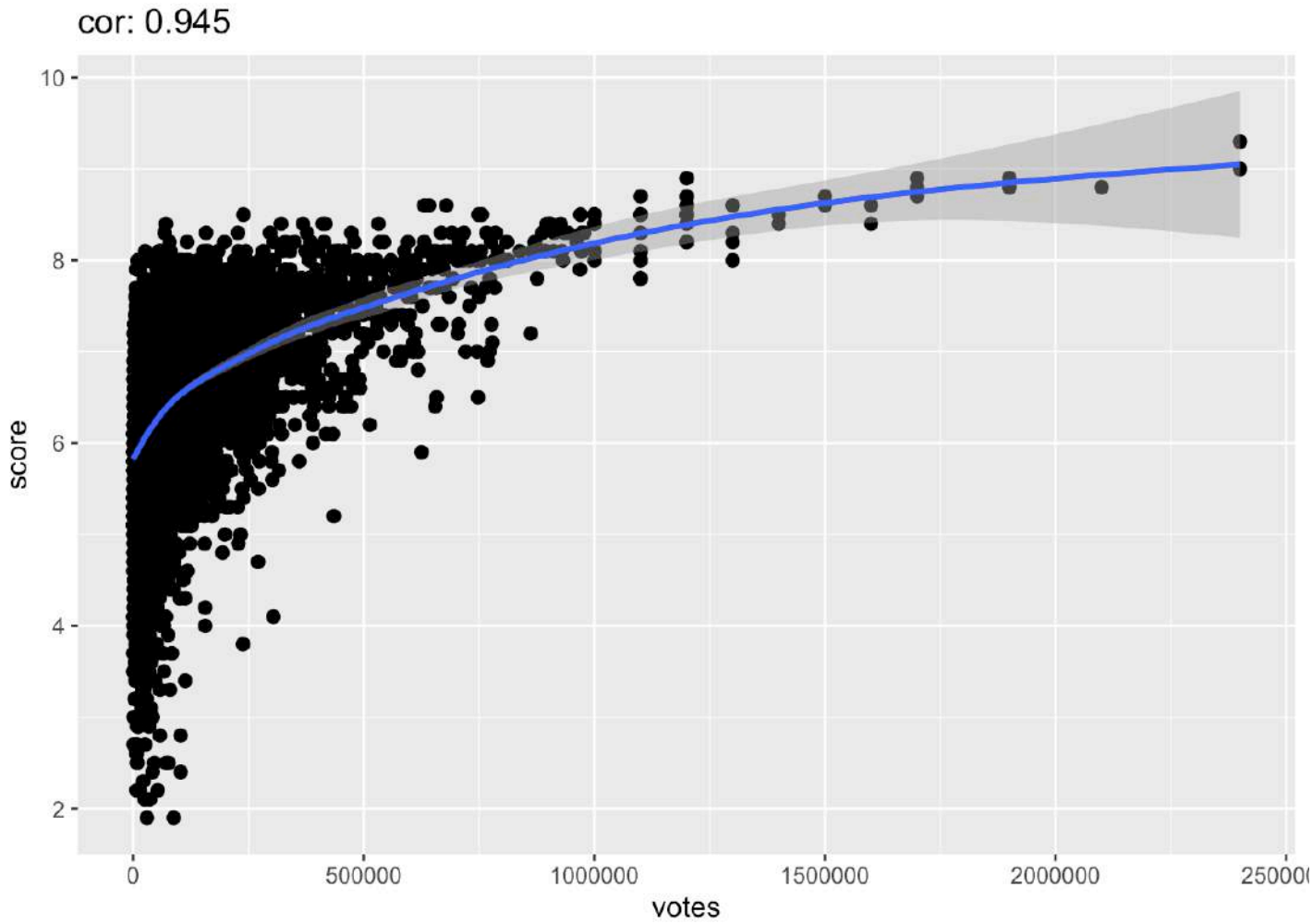
score

Lets create scatter plots to understand the relationship of the above variables better

```
ggplot(movieData, aes(x =votes, y =score))+
  geom_point(size=2) +
  stat_smooth(methos = lm, se = F, color = "red")+geom_smooth()+
  labs(title = "Votes Vs Score",
       x = "votes", y = "score")+ ggtitle(paste("cor:", 0.945))
```

```
## Warning: Ignoring unknown parameters: methos
```

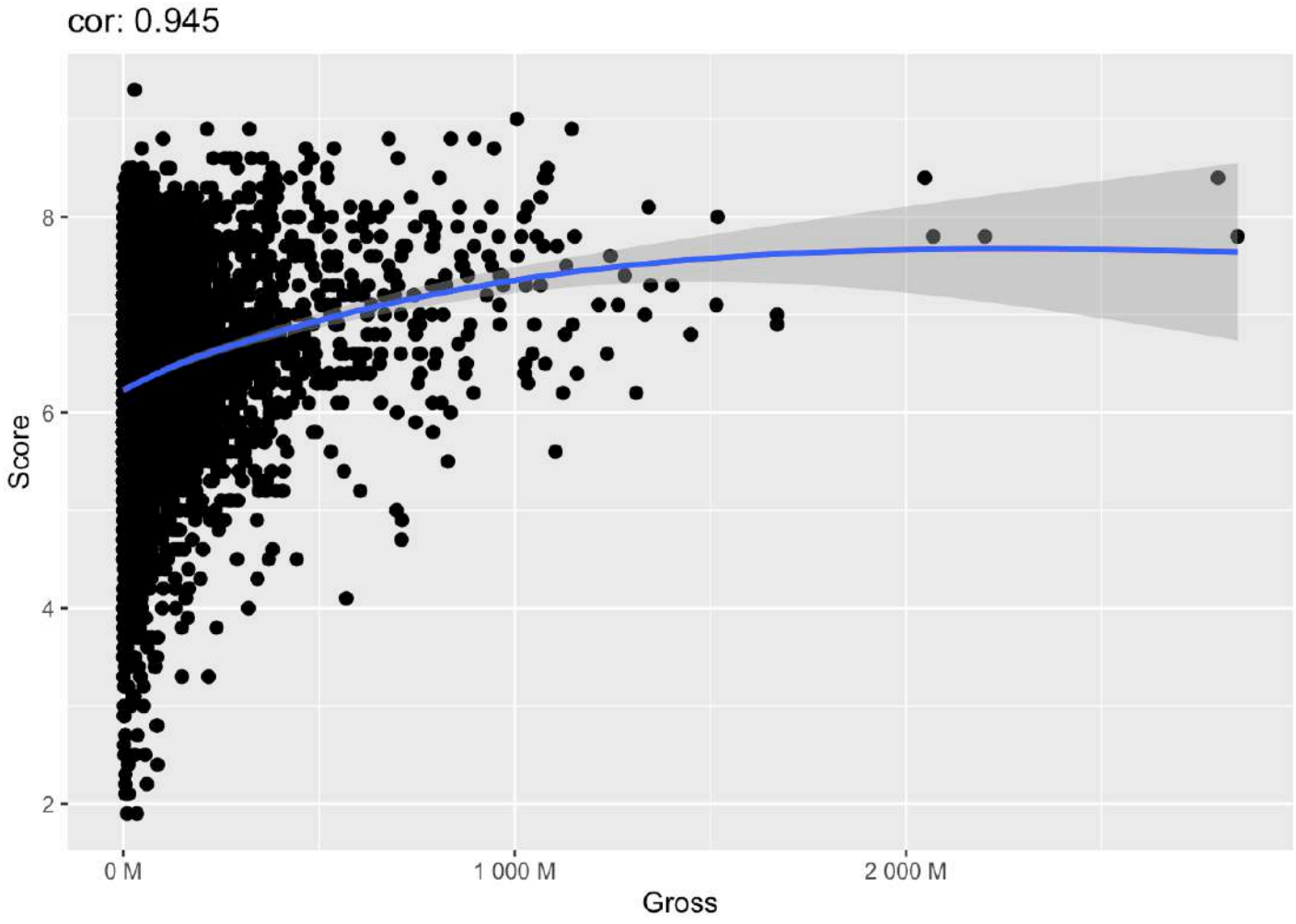
```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



```
ggplot(movieData, aes(x =gross, y =score))+
  geom_point(size=2) +
  stat_smooth(methos = lm, se = F, color = "red")+geom_smooth()+
  scale_x_continuous(labels = unit_format(unit = "M", scale = 1e-6))+
  labs(title = "Gross Vs Score",
       x = "Gross", y = "Score")+ ggtitle(paste("cor:", 0.945))
```

```
## Warning: Ignoring unknown parameters: methos
```

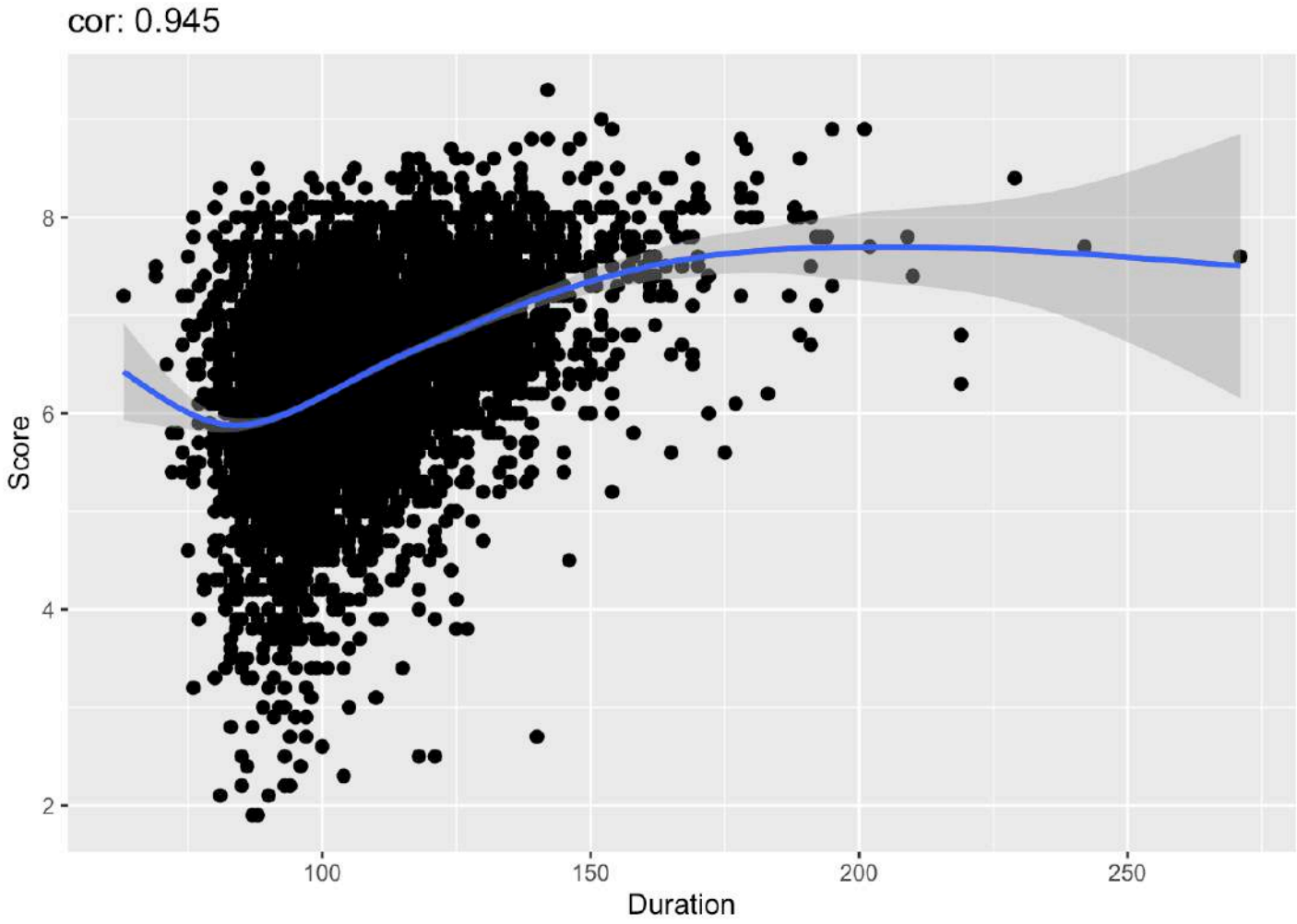
```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```




```
ggplot(movieData, aes(x =runtime, y =score))+
  geom_point(size=2) +
  stat_smooth(methos = lm, se = F, color = "red")+geom_smooth()+
  labs(title = "Duration Vs. Score",
       x = "Duration", y = "Score")+ ggtitle(paste("cor:", 0.945))
```

```
## Warning: Ignoring unknown parameters: methos
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```

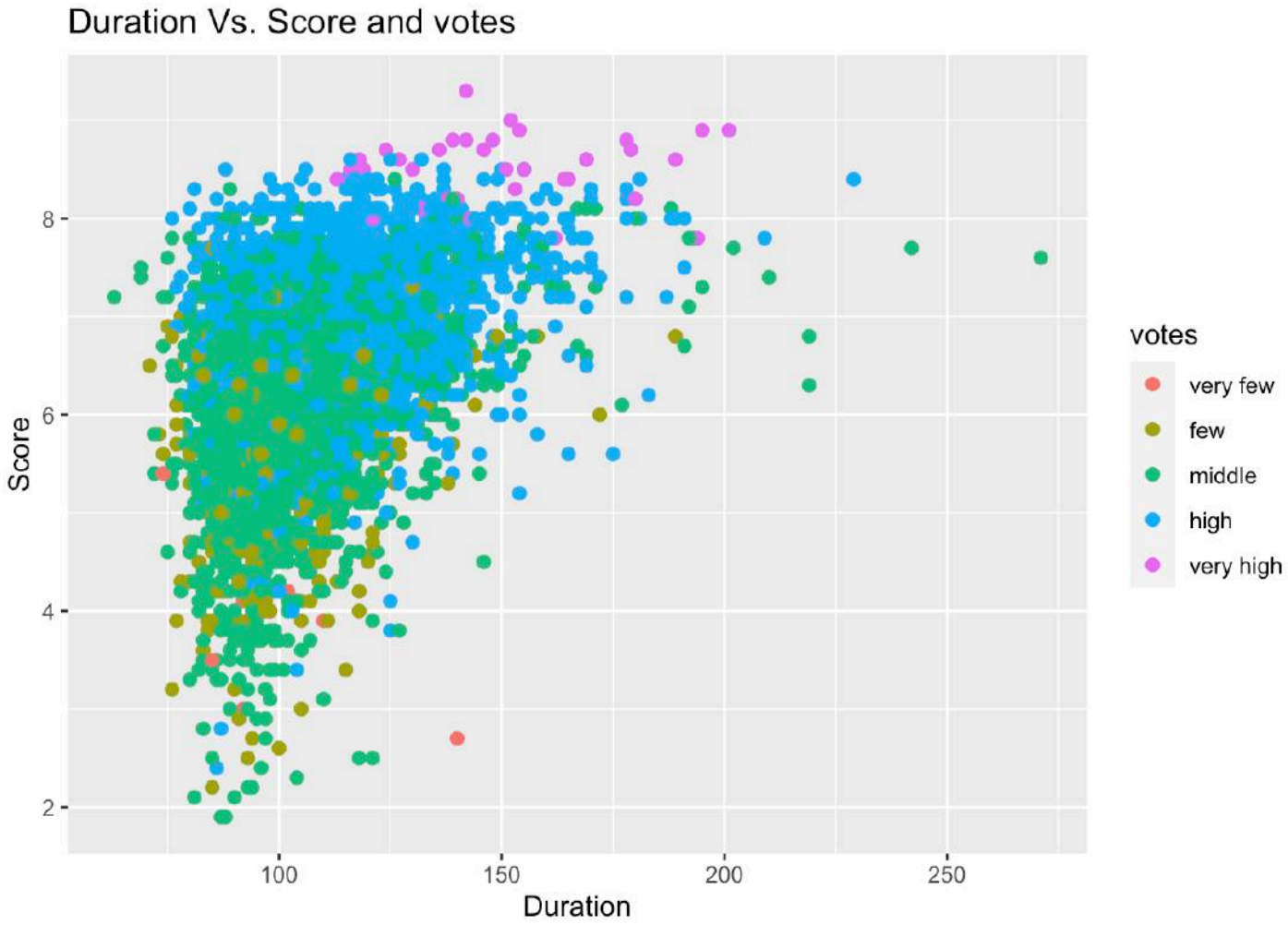


Lets categories the votes and then use it for colouring the above plots

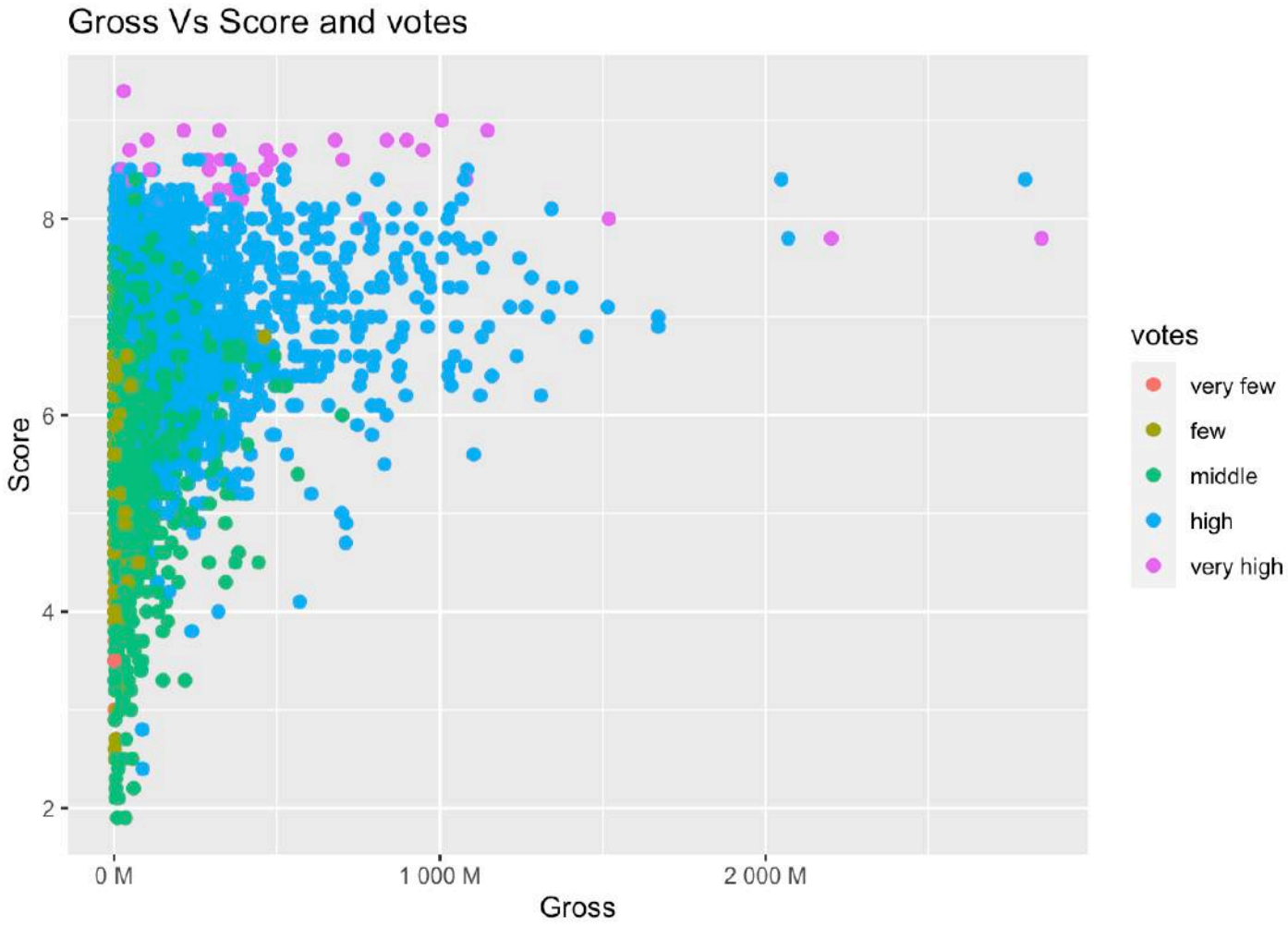
```
movieDataC<-movieData
movieDataC$votes<-cut(movieDataC$votes, breaks = c(7,1000,10000,100000,1000000,240
0000), labels = c("very few","few","middle","high","very high"))
summary(movieDataC$category)
```

```
## Length  Class   Mode
##      0   NULL   NULL
```

```
ggplot(movieDataC, aes(x =runtime, y =score))+
  geom_point(size=2, aes(colour=votes)) +
  labs(title = "Duration Vs. Score and votes",
       x = "Duration", y = "Score")
```




```
ggplot(movieDataC, aes(x =gross, y =score))+
  geom_point(size=2,aes(colour=votes))+
  scale_x_continuous(labels = unit_format(unit = "M", scale = 1e-6))+
  labs(title = "Gross Vs Score and votes",
       x = "Gross", y = "Score")
```



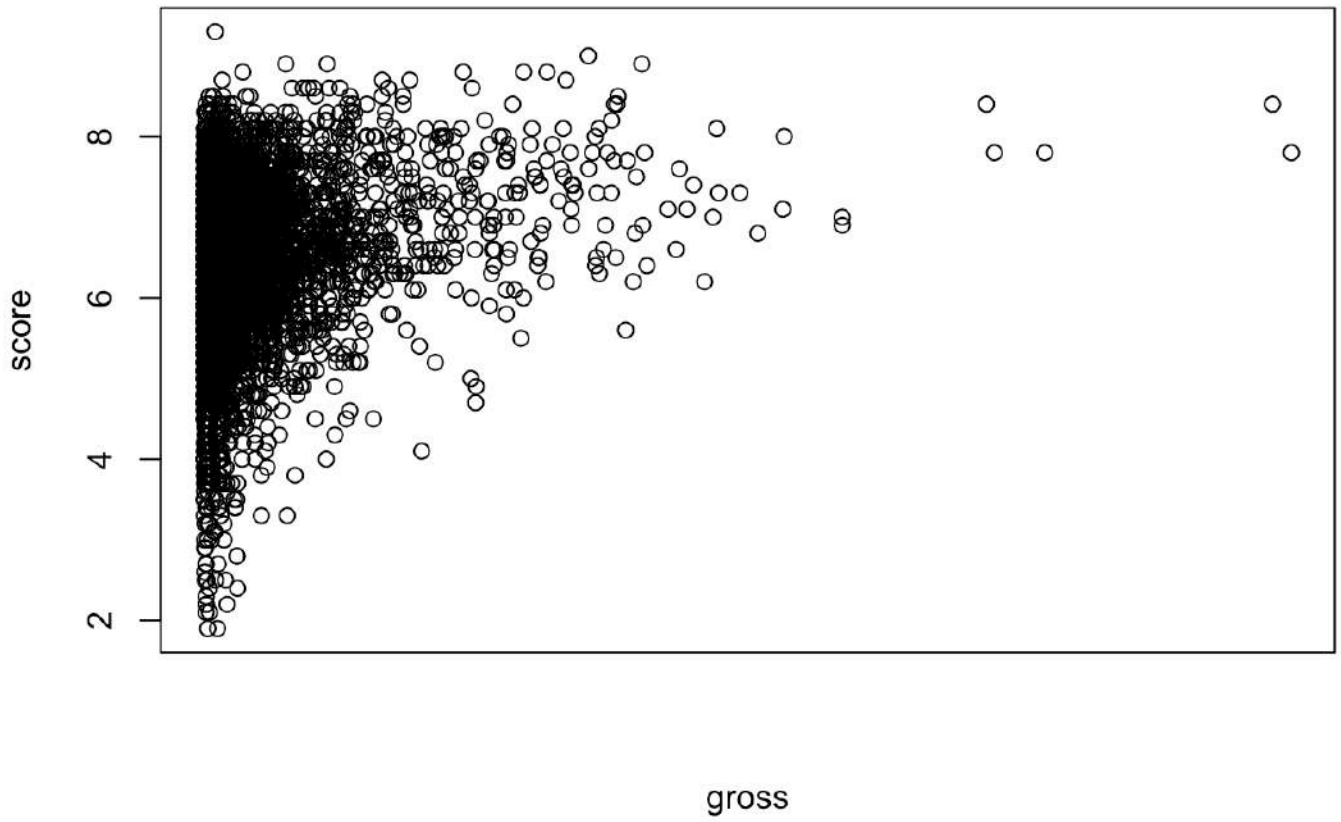
Conclusion:

Comedies, Action and Drama are mostly produced genre of movies United states hold the highest stakes in movie production, US movie production was drastically increased (upward) during 1980 and 1990 and also they are the largest players currently Roberto Benigni of Italy has the highest movie score average, 8.6 among the directors Among the actors, “Mark Hamill” and “Philippe Noiret” are having the highest avg move score Among the countries, Lebanon leads the chart of average move score. Surpisingly US not in the list of top 10 vote, gross and runtime are the most determine factors of getting movie score #### Linear Regression #### Now, we identify the variables that are determinig the score, Let’s select the determining variables

```
movies_important_variables = movieDataC[, c("score",
                                            "votes",
                                            "gross",
                                            "runtime")]

#regression plot - score vs gross

plot(score~gross, data=movies_important_variables,xaxt="none")
```




```
li = lm(score~gross, data=movies_important_variables)
print(li)
```

```
##
## Call:
## lm(formula = score ~ gross, data = movies_important_variables)
##
## Coefficients:
## (Intercept)        gross
##   6.275e+00    1.143e-09
```

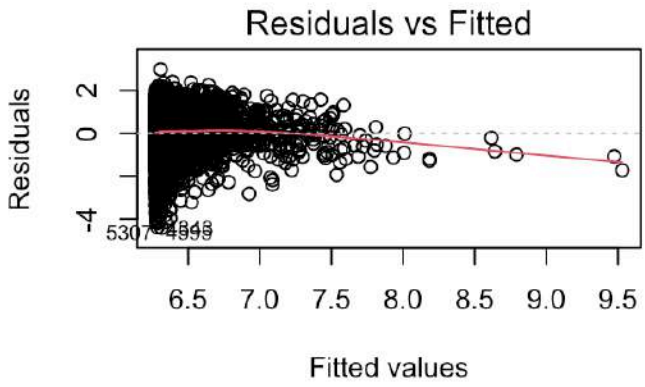
```
print(summary(li))
```

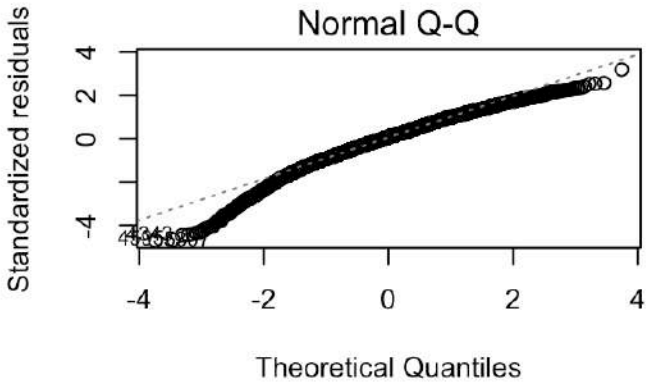
```
##
## Call:
## lm(formula = score ~ gross, data = movies_important_variables)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.4149 -0.5522  0.0650  0.6600  2.9920
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 6.275e+00  1.454e-02  431.48   <2e-16 ***
## gross       1.143e-09  6.809e-11   16.79   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9392 on 5433 degrees of freedom
## Multiple R-squared:  0.04933,    Adjusted R-squared:  0.04915
## F-statistic: 281.9 on 1 and 5433 DF,  p-value: < 2.2e-16
```

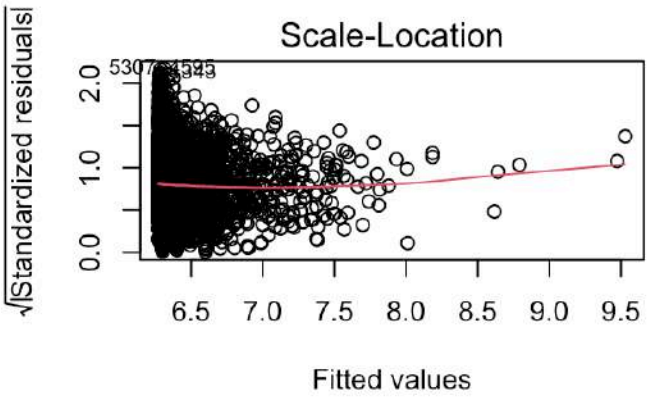
```
print(par(mfrow=c(2,2)))
```

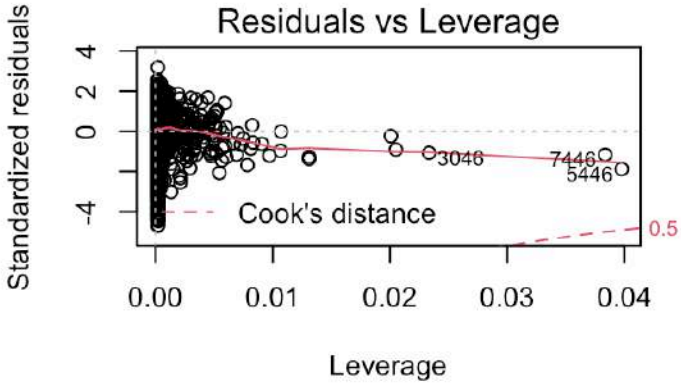
```
## $mfrow
## [1] 1 1
```

```
plot(li)
```



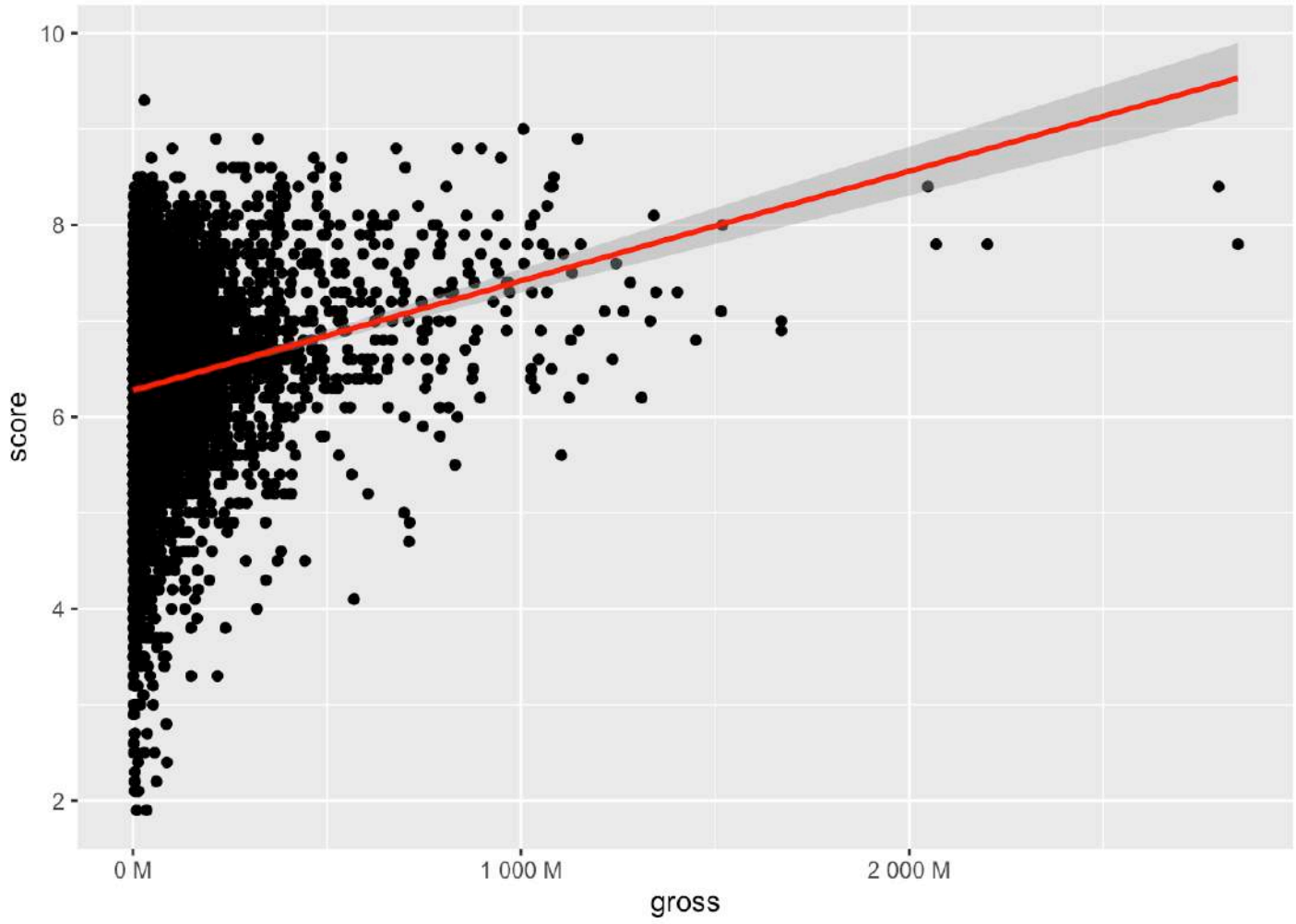







```
i<-ggplot(movies_important_variables,aes(x=gross,y=score))+geom_point()
i<-i+geom_smooth(method='lm',col="red")
i<- i + scale_x_continuous(labels = unit_format(unit = "M", scale = 1e-6))
print(i)
```

```
## `geom_smooth()` using formula 'y ~ x'
```


10
8
6
4
2
score
0 M
1 000 M
2 000 M
gross