Prob 3, CP 2: Um problema pequeno

15 de maio de 2016

Contextualização

Um grupo de pesquisa, chamado **grouplens**, que atua na área de sistemas de recomendação disponibilizaacesso a dados sobre avaliação de filmes no site do projeto. Os dados coletados para esta pesquisa foram coletados do site http://grouplens.org/datasets/movielens/latest/. Utilizamos para o nosso estudo o dataset **small**, que contém 100.000 avaliações realizadas por 700 usuários sobre 10.000 filmes cadastrados.

Problemas de Pesquisa

Nesta pesquisa queremos investigar dois problemas. O primeiro é verificar se existe alguma quantidade de gêneros num mesmo filme que em média recebe avaliações melhores. Caso exista uma combinação de gêneros que tenha uma média melhor, devemos estimar a diferença nas médias entre essa combinação e filmes com apenas um gênero.

Já o segundo problema é verificar quais gêneros com mais filmes possuem maior variação nas notas atribuídas aos filmes.

Seleção das Variáveis

Com o objetivo de avaliar o comportamento das avaliações, levando em consideração a quantidade de gêneros relacionados aos filmes, selecionamos 2 arquivos do dataset (movies.csv e ratings.csv). O arquivo **movies.csv** está relacionado aos filmes e seus gêneros. Já o arquivo **ratings** traz informações sobre as avaliações dos usuários.

O arquivo **movies.csv** traz uma coluna com informações sobre os gêneros. Os gêneros relativos ao filme vem em uma coluna, chamada **genres**, separados pela barra vertical como por exemplo **Comedy|Drama**. Além disto, não existe no arquivo uma coluna que nos informe a quantidade de gêneros associado a cada filme. Assim precisamos criar esta coluna, calcular levando em consideração esta coluna **genres** e adicioná-la ao dataset.

Para responder a pergunta sobre a variância também utilizamos a coluna **genres** para separar em cada linha diferente o gênero associado ao filmes. Assim, podemos realizar a sumarização necessária para identificar quais são os gêneros com mais filmes associados.

Primeiramente vamos ler o arquivo de origem sobre os filmes.

```
movies <- read.csv(paste ("C:\\Users\\Italo\\Dropbox\\ufcg\\FPCC2\\Atividade",
                          " 03\\Material\\ml-latest-small\\movies.csv",sep = "",
                          collapse = NULL))
```

Utilizaremos a função **str_count** para contar quantos gêneros temos em cada filme. Assim aplicando a função e adicionando a coluna temos:

```
#Conta quantos generos possui cada filme e adiciona a coluna
movies<-mutate(movies,qtgenres=str_count(movies$genres,fixed('|'))+1)
```

O próximo passo foi unir as informações sobre as avaliações dos filmes e seus gêneros em um único dataset. Para tal, utlizamos a função **merge**, unindo os dados pelo campo **movieId**:

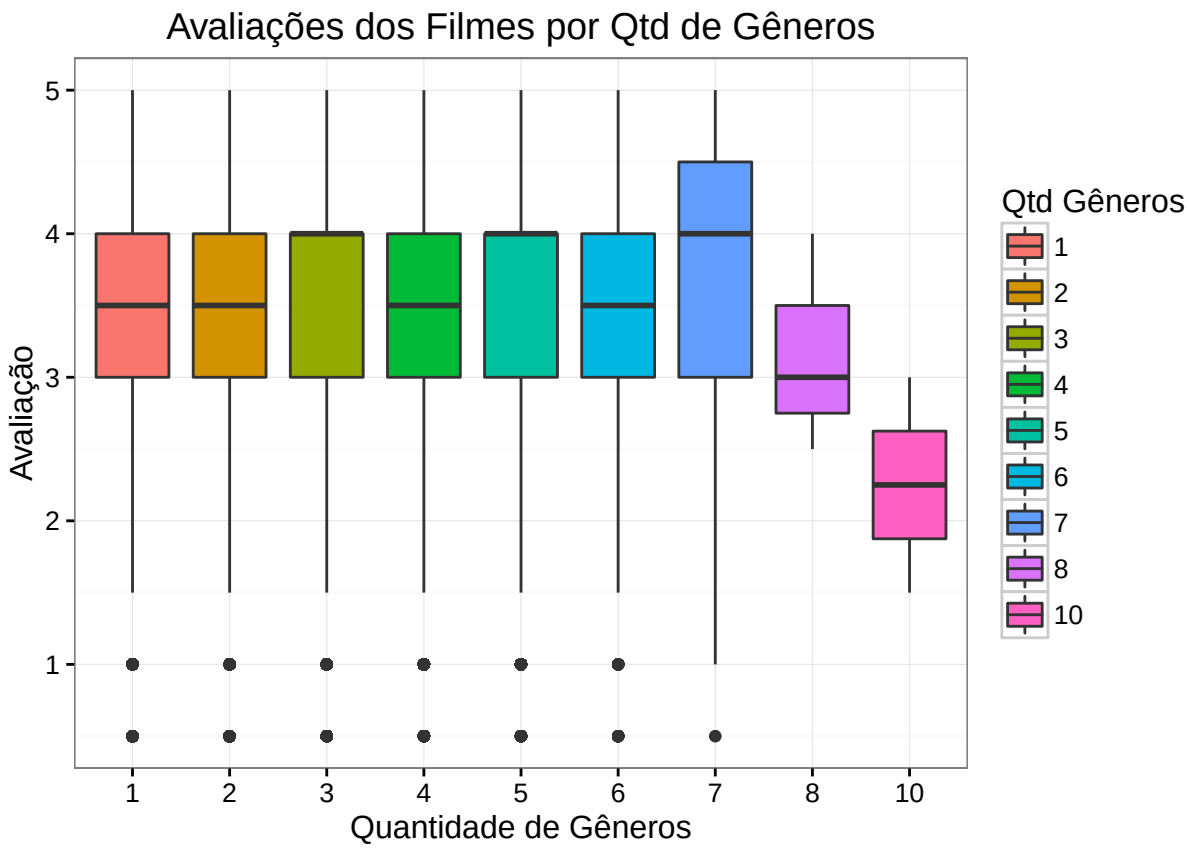
```
#Ler Tabela de Avaliações
ratings <- read.csv(paste ("C:\\Users\\Italo\\Dropbox\\ufcg\\FPCC2\\Atividade",
                           " 03\\Material\\ml-latest-small\\ratings.csv",sep = "",
                           collapse = NULL))

#join de tabelas de avaliações e filmes para juntar a quantidade de gêneros
movies.ratings <-  merge(movies, ratings,by="movieId")
```

Assim, para o realizar o nosso estudo faremos uso de 2 variáveis: **qtgenres** que indica quantos gêneros estão associados ao filme e **rating** que informa qual a avaliação realizada pelo usuário.

Distribuição dos Dados e Identificação de Outliers

Antes de iniciarmos a análise dos dados precisamos verificar como eles estão distribuídos. Assim, uma análise das avaliações para cada gênero foi realizada, como podemos ver no gráfico a seguir.

```
#Box Plot Distribuição para verificar a existência de outliers

p<-ggplot(movies.ratings, aes(factor(qtgenres), rating))
p<-p+geom_boxplot( aes(fill=factor(qtgenres)))
p<-p+labs(x="Quantidade de Gêneros",y="Avaliação")
p<-p+ggtitle("Avaliações dos Filmes por Qtd de Gêneros")
p<-p+ scale_fill_discrete(name="Qtd Gêneros")
p
```



Os dados sugerem que as distribuições para os gêneros 1,2,4,6 e 10 são simétricas, uma vez que as médias são bem próximas das medianas. Já os filmes com combinação de gêneros 3,5 e 7 apresentam a mediana próxima ao Q3 e podemos considerá-las negativamente assimétricas. Já os filmes com combinação de 8 gêneros apresentam a mediana próxima ao Q1, o que podemos dizer que são positivamente assimétricos. Entretanto, ao analisarmos o gráfico, percebemos a existência de *outliers*. Nos filmes entre 1 e 7 gêneros associados, outliers das avaliações com valores iguais ou menores que 1 foram identificados. Assim, uma filtragem para desconsiderar tais outliers foi realizada.

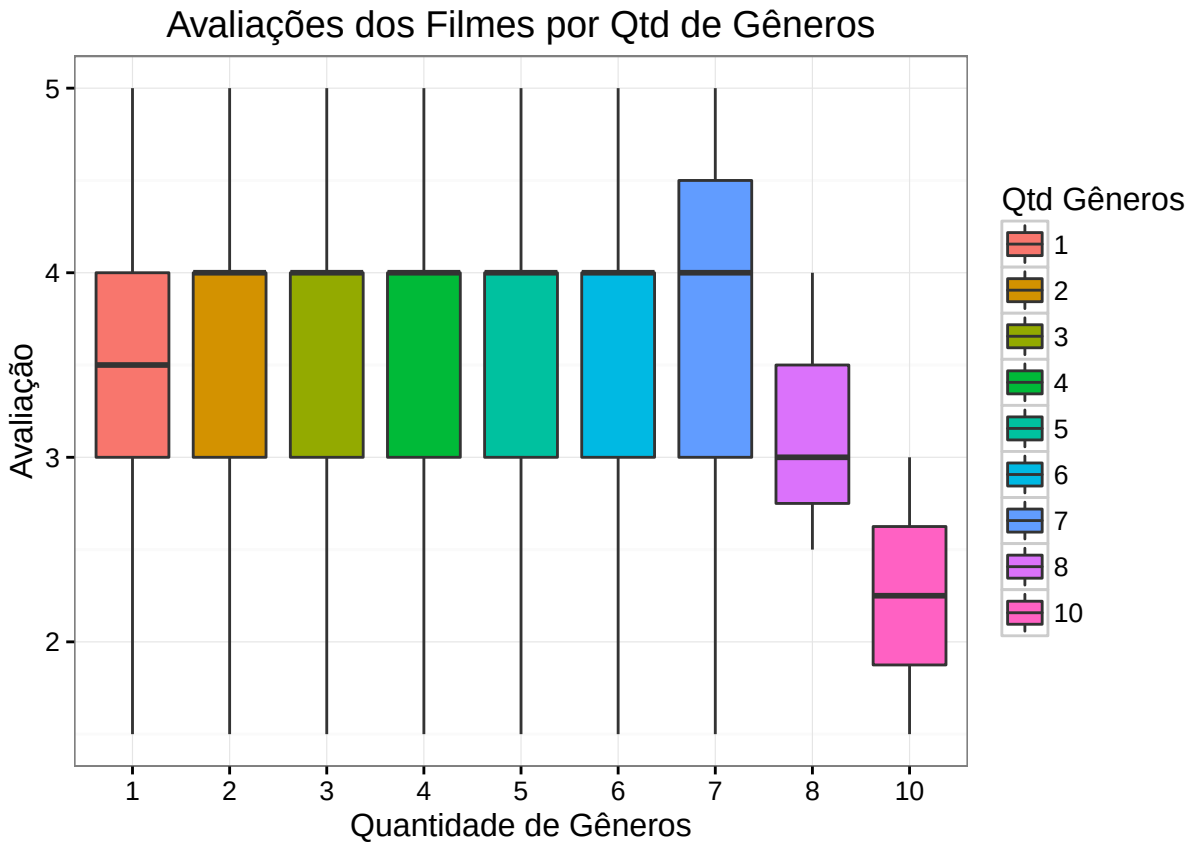
```
#filtra os dados desconsiderando os outliers com valores menores ou iguais a 1
movies.ratings.soutliers <- filter(movies.ratings,rating>1)
```

Dentro do conjunto de dados, foi verificada a presença de filmes (7 ocorrências) sem gênero relacionado. Neste caso, filtramos também os nossos dados para desconsiderar tais filmes.

```
movies.ratings.soutliers <- filter(movies.ratings.soutliers, genres!= "(no genres listed)")
```

Verficando novamente como ficaram as distribuições podemos perceber uma mudança na combinação de alguns gêneros. Os dados sugerem que as distribuições dos filmes com combinação 1 e 10 de gêneros são simétricas, uma vez que as médias são bem próximas das medianas. Já os filmes com combinação de gêneros 2,3,4,5,6 e 7 apresentam a mediana próxima ao Q3 e podemos considerá-las negativamente assimétricas. Já os filmes com 8 gêneros apresentam a mediana próxima ao Q1, o que podemos dizer que são positivamente assimétricos.

```
p<-ggplot(movies.ratings.soutliers, aes(factor(qtgenres), rating))
p<-p+geom_boxplot( aes(fill=factor(qtgenres)))
p<-p+labs(x="Quantidade de Gêneros",y="Avaliação")
p<-p+ggtitle("Avaliações dos Filmes por Qtd de Gêneros")
p<-p+ scale_fill_discrete(name="Qtd Gêneros")
p
```




```
#Media Geral para cada gênero
 estatisticas = movies.ratings.soutliers %>%
    group_by(qtgenres) %>%
    summarise("Média" = mean(rating), Mediana=median(rating))
```

```
kable(estatisticas, format = "markdown")
```

qtgenres	Média	Mediana
1	3.587480	3.50
2	3.630342	4.00
3	3.655454	4.00
4	3.632569	4.00
5	3.661535	4.00
6	3.654585	4.00
7	3.815517	4.00
8	3.166667	3.00
10	2.250000	2.25

Análise dos Dados

A primeira pergunta que queremos responder é : Existe alguma quantidade de gêneros num mesmo filme que em média recebe avaliações melhores? Logo, precisamos verificar as médias para cada combinação de gênero para verificar se alguma dela recebe valores melhores. No nosso caso vamos adotar um IC de 95%

para nossos cálculos.

Como podemos ver a seguir, os dados, já sem a presença de outliers, referentes a cada combinação de gêneros foram separados em datasets distintos.

```
genero1 <-filter(movies.ratings.soutliers,qtgenres==1)
genero2 <-filter(movies.ratings.soutliers,qtgenres==2)
genero3 <-filter(movies.ratings.soutliers,qtgenres==3)
genero4 <-filter(movies.ratings.soutliers,qtgenres==4)
genero5 <-filter(movies.ratings.soutliers,qtgenres==5)
genero6 <-filter(movies.ratings.soutliers,qtgenres==6)
genero7 <-filter(movies.ratings.soutliers,qtgenres==7)
genero8 <-filter(movies.ratings.soutliers,qtgenres==8)
genero10 <-filter(movies.ratings.soutliers,qtgenres==10)
```

Uma função **bsci**, utilizando **bootstrapping**, foi criada para ser utilizada durante o processo de amostragem. A mesma será utilizada em cada combinação de gênero para computar a média, bem como os máximos e mínimos do IC, verificando se a estimativa obtida inclue a média dentro do IC proposto. Se incluir em menos, o erro está sendo maior que o esperado.

```
bsci<-function(x,B,N,m){
    bstrap <- data.frame(upper = c(), mean = c(), lower = c())
    for (i in 1:B){
        bsample <- sample(x,N,replace=T)
        interval <- CI.percentile(bootstrap(bsample, mean, R = B))
        bstrap <-rbind(bstrap, data.frame(mean = mean(interval),
                                          lower = interval[1],
                                          upper = interval[2]))
    }
    bstrap <- bstrap %>%
        mutate(contem_media = (upper >= m & lower <= m))

    return(bstrap)
}
```

Com a função definida, aplicamos ela para cada combinação disponível e efetuamos o cálculo o IC da média de cada combinação. Como parâmetros, definimos o tamanho da amostra como N=100 e a quantidade de repetições como R=1000.

```
  saida_g1 <- bsci(genero1$rating,1000,100,mean(genero1$rating))
  saida_g2 <- bsci(genero2$rating,1000,100,mean(genero2$rating))
  saida_g3 <- bsci(genero3$rating,1000,100,mean(genero3$rating))
  saida_g4 <- bsci(genero4$rating,1000,100,mean(genero4$rating))
  saida_g5 <- bsci(genero5$rating,1000,100,mean(genero5$rating))
  saida_g6 <- bsci(genero6$rating,1000,100,mean(genero6$rating))
  saida_g7 <- bsci(genero7$rating,1000,100,mean(genero7$rating))
  saida_g8 <- bsci(genero8$rating,1000,100,mean(genero8$rating))
  saida_g10 <- bsci(genero10$rating,1000,100,mean(genero10$rating))
```

Agora precisamos agrupar as médias das amostras de cada combinação. Assim, criamos uma função para realizar tal procedimento. Abaixo segue sua definição:

```
mediasg95 <- function(x,d,R,idg){
  se <- sd(d$mean)/sqrt(R)
  vlower <-mean(d$mean) -1.96*se
  vupper <-mean(d$mean) +1.96*se
  return(rbind(x, data.frame(genero=idg,mean = mean(d$mean),
                                        lower = vlower,
                                        upper = vupper)))


}
```

Agrupando as médias das amostragens para cada combinação de gênero em um único dataset temos:

```
medias.generos <- data.frame(genero=c(),upper = c(), mean = c(), lower = c())

medias.generos <- mediasg95(medias.generos,saida_g1,1000,1)
medias.generos <- mediasg95(medias.generos,saida_g2,1000,2)
medias.generos <- mediasg95(medias.generos,saida_g3,1000,3)
medias.generos <- mediasg95(medias.generos,saida_g4,1000,4)
medias.generos <- mediasg95(medias.generos,saida_g5,1000,5)
medias.generos <- mediasg95(medias.generos,saida_g6,1000,6)
medias.generos <- mediasg95(medias.generos,saida_g7,1000,7)
medias.generos <- mediasg95(medias.generos,saida_g8,1000,8)
medias.generos <- mediasg95(medias.generos,saida_g10,1000,10)
```

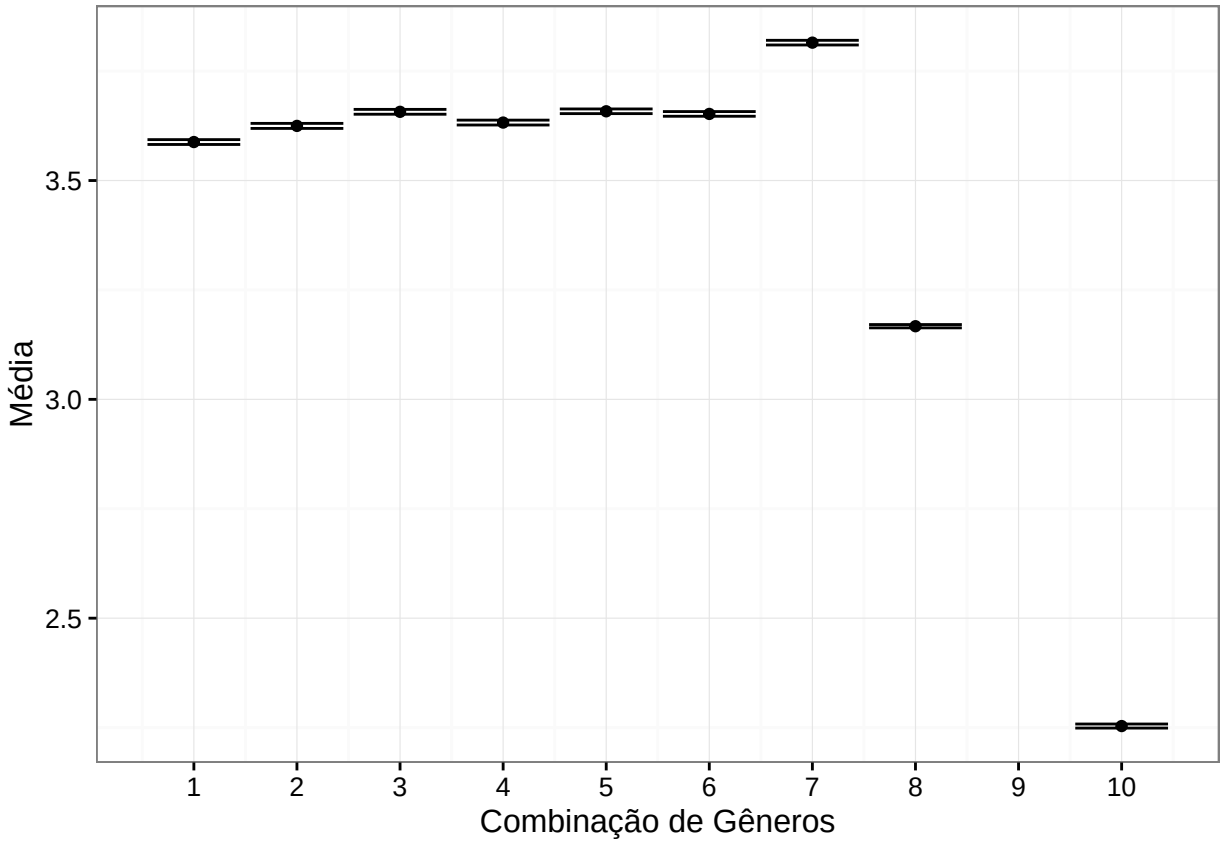
Ordenando as médias pelo valor da média em ordem decrescente, temos:

```
 arrange(medias.generos, desc(mean))
```

```
##   genero     mean    lower    upper
## 1      7 3.815004 3.809671 3.820336
## 2      5 3.658182 3.652838 3.663527
## 3      3 3.656989 3.651441 3.662537
## 4      6 3.652182 3.646839 3.657525
## 5      4 3.632408 3.626853 3.637963
## 6      2 3.624778 3.619026 3.630530
## 7      1 3.587926 3.582379 3.593472
## 8      8 3.167128 3.163360 3.170896
## 9     10 2.253553 2.248938 2.258167
```

Analisando o gráfico a seguir, onde as médias para cada combinação de gênero são exibidas que a combinação com 7 gêneros possui uma avaliação média melhor que as demais.

```
medias.generos %>%
  ggplot(aes(x = genero, y = mean)) +
  geom_point() +
  geom_errorbar(aes(ymin = lower, ymax = upper)) +
  scale_x_continuous(breaks = 1:10) +
  xlab("Combinação de Gêneros") +
  ylab("Média")
```



Assim, podemos estimar com uma confiança de 95% que a combinação com 7 gêneros possui a melhor avaliação média dentre as combinações apresentadas.

O próximo passo agora é estimar a diferença nas médias entre essa combinação e filmes com apenas um gênero. Uma função foi criada para calcular a diferença entre as amostras das duas combinações. Segue abaixo uma descrição da mesma:

```
differenca_medias <- function(x, y,N){
  boot_x <- sample(x, size = N, replace = TRUE) # aqui é o bootstrap
  boot_y <- sample(y, size = N, replace = TRUE) # de novo!
  return(mean(boot_x) - mean(boot_y))
}
```

Agora para a um número de repetições R=1000 e um número de amostras N=100, iremos estimar o IC para a diferença das médias entre as combinações de gêneros.

```
calculos_diferencas = data_frame(i = 1:1000)
for(i in seq(1, 1000)){
  boot_x <- sample(genero7$rating, size = 100, replace = TRUE) # aqui é o bootstrap
  boot_y <- sample(genero1$rating, size = 100, replace = TRUE) # de novo!
  diff = mean(boot_x) - mean(boot_y);

  calculos_diferencas = calculos_diferencas %>%
  rowwise() %>%
  mutate(diferenca = diff)
}
```

```r
# IC com 95%:
alpha = .05
qd <-quantile(calculos_diferencas$diferenca, probs = c(alpha/2, 1 - alpha/2))

se <- sd(calculos_diferencas$diferenca)/sqrt(1000)
vlower <-mean(calculos_diferencas$diferenca) -1.96*se
vupper <-mean(calculos_diferencas$diferenca) +1.96*se
```

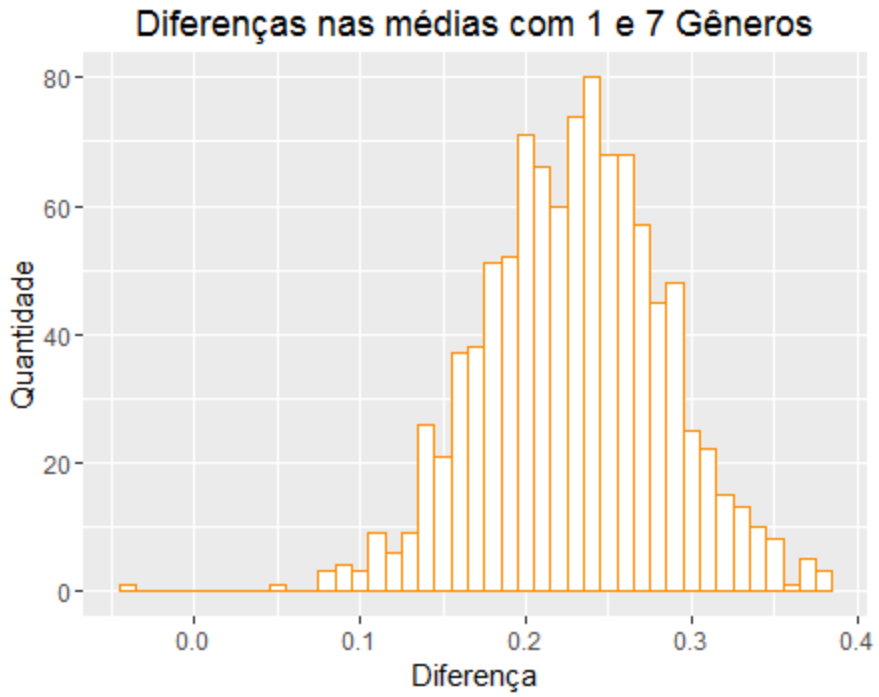
A média das diferenças entre a combinação de 7 gêneros e 1 gênero foi calculada. Com uma confiança de 95%, a mediana e a média das médias das diferenças respectivamente, com limite seu superior e limite inferior são ilustrados na tabela abaixo.

```r
tcdiferencas <- data.frame(Mediana=c(),upper = c(), "Média" = c(), lower = c())



tcdiferencas <- (rbind(tcdiferencas, data.frame(Mediana=median(calculos_diferencas$diferenca),"Média" =
                                          lower = vlower,
                                          upper = vupper)))
```

```r
kable(tcdiferencas, format = "markdown")
```

Mediana	lower	Média	upper
0.2299832	0.2256701	0.2290403	0.2324104

Podemos observar que a distribuição no histograma das diferenças das médias é normal,visto que a média está muito próxima da mediana.



“‘

A terceira pergunta diz respeito a variância. Nós queremos investigar dentre os 10 gêneros que têm mais filmes, quais possuem maior variação nas notas atribuídas a seus filmes? Como vimos anteriormente, precisamos de alguma forma separar os gêneros em linhas para iniciarmos a contagem para verificar quais gêneros possuem mais filmes. Selecionar as colunas **title** e **genres** temos: .

```
movies.genre <- select(movies,title,genres)
```

Selecionadas as colunas, agora vamos separar os gêneros por linha. Para isto utilizaremos as funções **strsplit** e **unnest**.

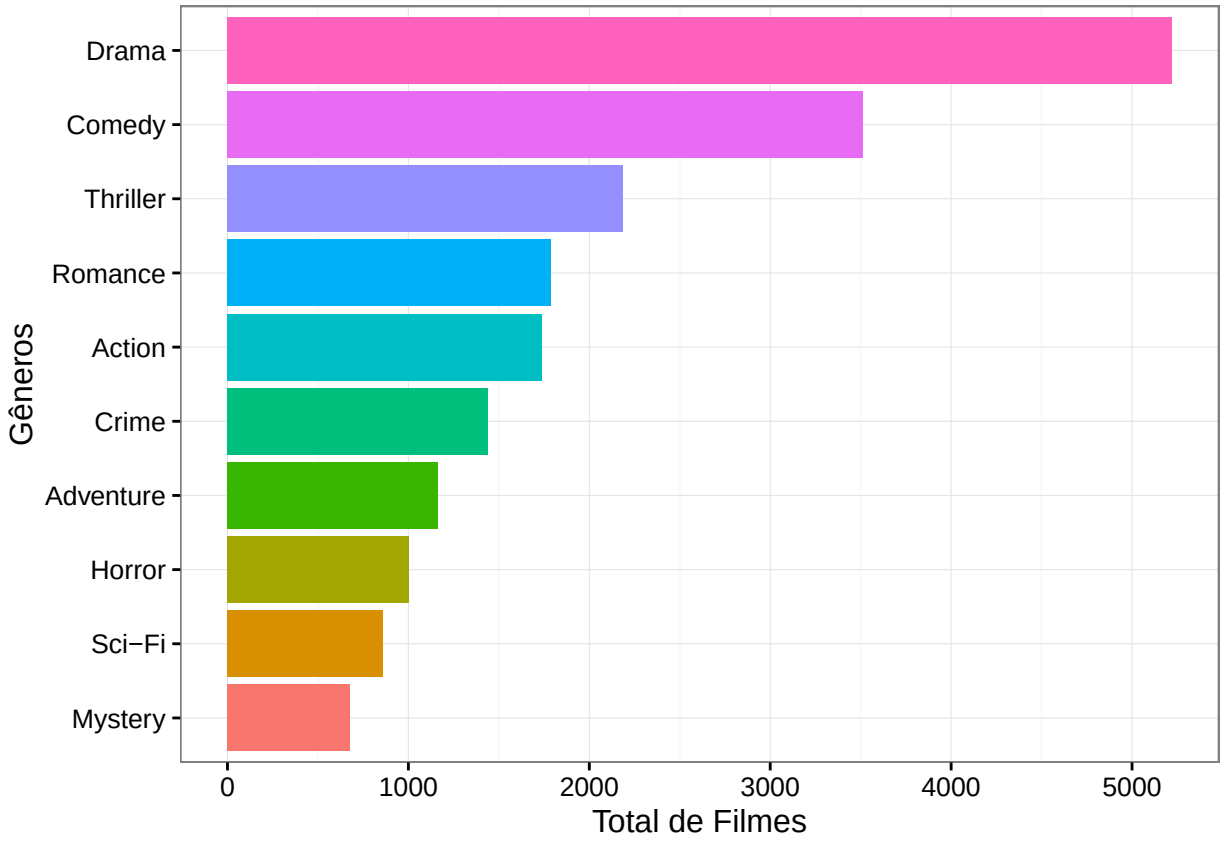
```
movies.genre <- movies.genre %>%
    mutate(qtd=1, genres = strsplit(as.character(genres), "\\|")) %>%
    unnest(genres)
```

O próximo passo agora é calcular quais são os gêneros com mais filmes e filtrar os 10 primeiros.

```
#soma e agrupa por genero
estatisticas_fgeneros = movies.genre %>%
    group_by(genres) %>%
    summarise(qtd = sum(qtd))

#ordena pelos generos com mais filmes.
estatisticas_fgeneros <- arrange(estatisticas_fgeneros,desc(qtd))

#puxa somente os 10 primeiros generos
dezgenerosmaisfilmes <- head(estatisticas_fgeneros,10)
```

Podemos obeservar no gráfico abaixo que o gênero **Drama** possui mais de 5000 títulos, seguidos por **Comedy** (mais de 3000) e **Thriller** (mais de 2000) especificamente.

```
#Ordenar no gráfico os generos
dezgenerosmaisfilmes3 <- dezgenerosmaisfilmes2 <- data.frame(car = rownames(dezgenerosmaisfilmes),
                                                dezgenerosmaisfilmes, row.names = NULL)

dezgenerosmaisfilmes3$genres <- factor(dezgenerosmaisfilmes3$genres,
              levels = dezgenerosmaisfilmes2[order(dezgenerosmaisfilmes$qtd), "genres"])

#Exibir o gráfico

p1<- ggplot(dezgenerosmaisfilmes3, aes(x = genres, y = qtd, fill=genres)) +
  geom_bar(stat = "identity") +
  xlab("Gêneros") +
  ylab("Total de Filmes") +
  coord_flip() +
  theme(legend.position = "none")

grid.arrange(arrangeGrob(p1))
```



Uma vez descoberto os 10 gêneros com mais filmes, vamos separar os dados em dataset separados para facilitar a estimativa da variância.

```
genero1_ratings <- filter(movies.ratings.soutliers, grepl("Drama",genres))
genero1_ratings <- select(genero1_ratings, title,rating,genres)

genero2_ratings <- filter(movies.ratings.soutliers, grepl("Comedy",genres))
genero2_ratings <- select(genero2_ratings, title,rating,genres)

genero3_ratings <- filter(movies.ratings.soutliers, grepl("Thriller",genres))
genero3_ratings <- select(genero3_ratings, title,rating,genres)


genero4_ratings <- filter(movies.ratings.soutliers, grepl("Romance",genres))
genero4_ratings <- select(genero4_ratings, title,rating,genres)

genero5_ratings <- filter(movies.ratings.soutliers, grepl("Action",genres))
genero5_ratings <- select(genero5_ratings, title,rating,genres)

genero6_ratings <- filter(movies.ratings.soutliers, grepl("Crime",genres))
genero6_ratings <- select(genero6_ratings, title,rating,genres)

genero7_ratings <- filter(movies.ratings.soutliers, grepl("Adventure",genres))
genero7_ratings <- select(genero7_ratings, title,rating,genres)

genero8_ratings <- filter(movies.ratings.soutliers, grepl("Horror",genres))
```

```
genero8_ratings <- select(genero8_ratings, title,rating,genres)

genero9_ratings <- filter(movies.ratings.soutliers, grepl("Sci-Fi",genres))
genero9_ratings <- select(genero9_ratings, title,rating,genres)

genero10_ratings <- filter(movies.ratings.soutliers, grepl("Mystery",genres))
genero10_ratings <- select(genero10_ratings, title,rating,genres)
```

Uma função foi definida com o intuito de facilitar o cálculo da variância das amostras, em um número de repetições definidas previamente. Abaixo segue a descrição da função:

```
bsciv<-function(x,B,N,m){
    bstrap <- data.frame(upper = c(), variance = c(), lower = c())
    for (i in 1:B){
        bsample <- sample(x,N,replace=T)
        interval <- CI.percentile(bootstrap(bsample, var, R = B))
        bstrap <-rbind(bstrap, data.frame(variance = var(bsample),
                                          lower = interval[1],
                                          upper = interval[2]))
    }
    bstrap <- bstrap %>%
        mutate(contem_var = (upper >= m & lower <= m))

    return(bstrap)
}
```

Com a função definida, calculamos para cada gênero selecionado o IC da variância. Como parâmetros, definimos o tamanho da amostra como N=100 e a quantidade de repetições como R=1000.

```
  saida_g1_var <- bsciv(genero1_ratings$rating,1000,100,var(genero1_ratings$rating))
  saida_g2_var <- bsciv(genero2_ratings$rating,1000,100,var(genero2_ratings$rating))
  saida_g3_var <- bsciv(genero3_ratings$rating,1000,100,var(genero3_ratings$rating))
  saida_g4_var <- bsciv(genero4_ratings$rating,1000,100,var(genero4_ratings$rating))
  saida_g5_var <- bsciv(genero5_ratings$rating,1000,100,var(genero5_ratings$rating))
  saida_g6_var <- bsciv(genero6_ratings$rating,1000,100,var(genero6_ratings$rating))
  saida_g7_var <- bsciv(genero7_ratings$rating,1000,100,var(genero7_ratings$rating))
  saida_g8_var <- bsciv(genero8_ratings$rating,1000,100,var(genero8_ratings$rating))
  saida_g9_var <- bsciv(genero9_ratings$rating,1000,100,var(genero9_ratings$rating))
  saida_g10_var <- bsciv(genero10_ratings$rating,1000,100,var(genero10_ratings$rating))
```

Com as variâncias calculadas para todos as amostras de todos os gêneros selecionados, vamos definir uma função para calcular o IC da média da variância de todas as amostras, com confiança de 95%. A função é detalhada a seguir:

```
varg95 <- function(x,d,R,idg){
  se <- sd(d$var)/sqrt(R)
  vlower <-mean(d$variance) -1.96*se
  vupper <-mean(d$variance) +1.96*se
  return(rbind(x, data.frame(genero=idg,variance = mean(d$var),
                                        lower = vlower,
                                        upper = vupper)))
```

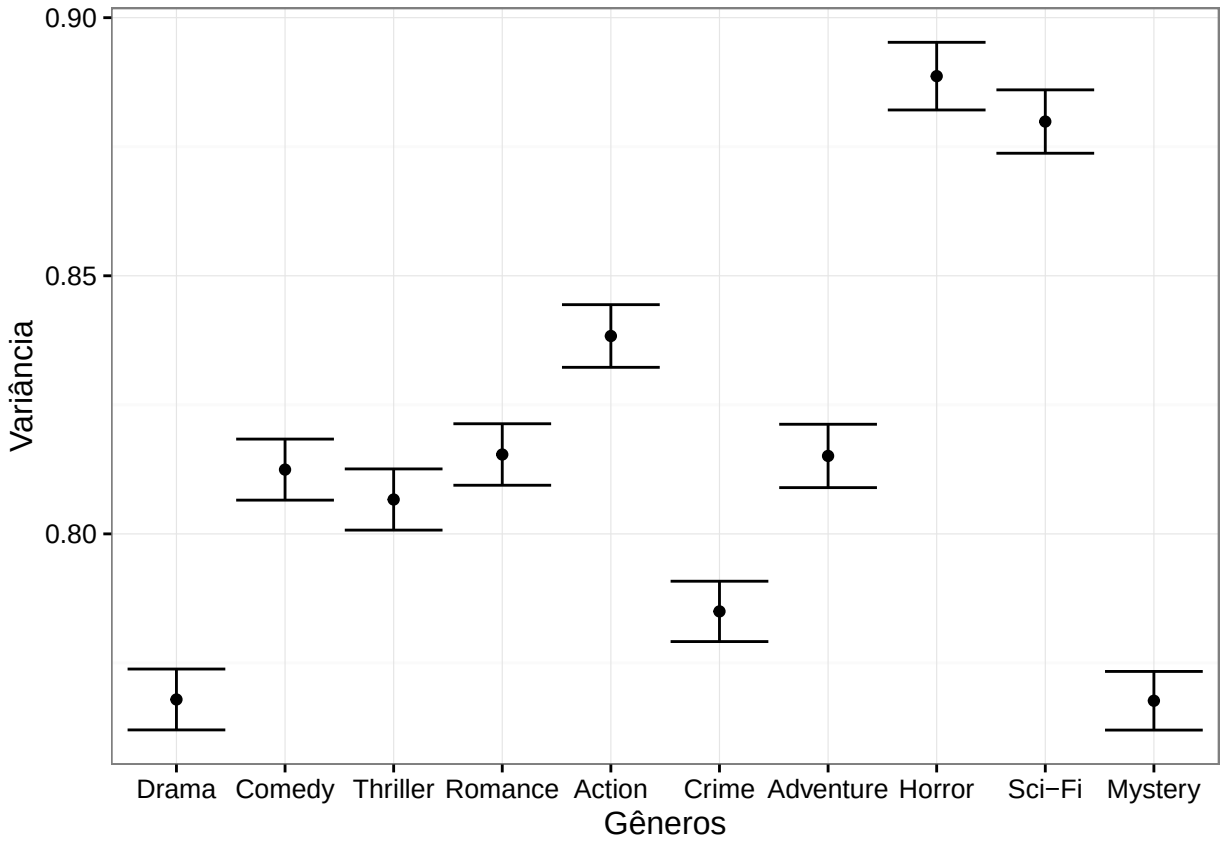
```
}
```

Agora vamos agrupar os resultados em um único dataset.

```
medias.generos <- data.frame(genero=c(),upper = c(), mean = c(), lower = c())

var.generos <- data.frame(genero=c(),upper = c(), variance= c(), lower = c())

var.generos <- varg95(var.generos,saida_g1_var,1000,"Drama")
var.generos <- varg95(var.generos,saida_g2_var,1000,"Comedy")
var.generos <- varg95(var.generos,saida_g3_var,1000,"Thriller")
var.generos <- varg95(var.generos,saida_g4_var,1000,"Romance")
var.generos <- varg95(var.generos,saida_g5_var,1000,"Action")
var.generos <- varg95(var.generos,saida_g6_var,1000,"Crime")
var.generos <- varg95(var.generos,saida_g7_var,1000,"Adventure")
var.generos <- varg95(var.generos,saida_g8_var,1000,"Horror")
var.generos <- varg95(var.generos,saida_g9_var,1000,"Sci-Fi")
var.generos <- varg95(var.generos,saida_g10_var,1000,"Mystery")
```

O gráfico a seguir ilustra a estimativa de variância para cada gênero, com uma confiança de 95%. Podemos notar que os gêneros **Horror** e **Sci-Fi** apresentam as maiores variâncias dentre os 10 gêneros com mais filmes.

```
var.generos %>%
  ggplot(aes(x = genero, y = variance, )) +
  geom_point() +
  geom_errorbar(aes(ymin = lower, ymax = upper)) +
  labs(x="Gêneros",y="Variância")
```


0.90
0.85
0.80
Variância
Drama
Comedy
Thriller
Romance
Action
Crime
Adventure
Horror
Sci−Fi
Mystery
Gêneros

“‘