# Module 3 Solutions

Make sure the dplyr and ggplot2 packages are installed and then loaded correctly!

Hopefully you have saved the data and RMD file into the same folder, and created a project file for that folder. If so then the imdb dataset can be read in without modifying the code below.

Since some of the outputs can be pretty long, I would advice not to knit this file and only explore it in RStudio, or to modify it so that the outputs get shorter, using the function slice() for example.

library(dplyr)

##   
## Attaching package: 'dplyr'

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

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

library(ggplot2)  
imdb <- readRDS("imdb2020.RDS")

## Exercises

**Exercise 1. rank the directors from youngest to oldest**

## Answer 1

A simple answer without using pipe would be

arrange(imdb, desc(birthYear)) %>%  
 slice(1:20)

## title type year length numVotes averageRating  
## 1 The Last Guest movie 2017 90 735 6.9  
## 2 Bruder vor Luder movie 2015 88 788 1.7  
## 3 Who I Am video 2014 3 1529 8.1  
## 4 Destination: Dewsbury movie 2018 86 622 5.8  
## 5 Square One: Michael Jackson movie 2019 83 1223 8.1  
## 6 Blame movie 2017 100 2428 5.8  
## 7 Farmlands movie 2018 73 917 8.6  
## 8 White Chamber movie 2018 89 1278 4.4  
## 9 Honeyland movie 2019 89 12277 8.0  
## 10 Mickey and the Bear movie 2019 88 884 6.6  
## 11 After Everything movie 2018 95 1045 6.3  
## 12 Singularity movie 2017 92 12748 3.8  
## 13 The Visitor short 2012 4 947 6.4  
## 14 Homeward movie 2019 96 732 7.3  
## 15 Conscientia short 2011 6 896 6.2  
## 16 The Slender Case short 2012 15 1105 5.8  
## 17 Starters short 2012 6 1061 6.6  
## 18 Hate Crime movie 2012 73 813 4.6  
## 19 Population Zero short 2013 12 1336 6.6  
## 20 Coin Heist movie 2017 97 2368 4.8  
## director birthYear animation action adventure comedy  
## 1 Matt Horton 2000 TRUE TRUE TRUE FALSE  
## 2 Roman Lochmann 1999 FALSE FALSE FALSE TRUE  
## 3 Joshua Widgren 1998 FALSE FALSE FALSE FALSE  
## 4 Jack Spring 1996 FALSE FALSE FALSE TRUE  
## 5 Danny Wu 1996 FALSE FALSE FALSE FALSE  
## 6 Quinn Shephard 1995 FALSE FALSE FALSE FALSE  
## 7 Lauren Southern 1995 FALSE FALSE FALSE FALSE  
## 8 Paul Raschid 1993 FALSE FALSE FALSE FALSE  
## 9 Tamara Kotevska 1993 FALSE FALSE FALSE FALSE  
## 10 Annabelle Attanasio 1993 FALSE FALSE FALSE FALSE  
## 11 Hannah Marks 1993 FALSE FALSE FALSE TRUE  
## 12 Robert Kouba 1992 FALSE TRUE TRUE FALSE  
## 13 Robert Kouba 1992 FALSE FALSE FALSE FALSE  
## 14 Nariman Aliev 1992 FALSE FALSE FALSE FALSE  
## 15 Robert Kouba 1992 FALSE FALSE FALSE FALSE  
## 16 Robert Kouba 1992 FALSE FALSE FALSE FALSE  
## 17 Robert Kouba 1992 FALSE FALSE FALSE FALSE  
## 18 James Cullen Bressack 1992 FALSE TRUE FALSE FALSE  
## 19 Robert Kouba 1992 FALSE TRUE FALSE FALSE  
## 20 Emily Hagins 1992 FALSE FALSE FALSE FALSE  
## documentary fantasy romance sci\_fi thriller  
## 1 FALSE FALSE FALSE FALSE FALSE  
## 2 FALSE FALSE TRUE FALSE FALSE  
## 3 FALSE FALSE FALSE FALSE FALSE  
## 4 FALSE FALSE FALSE FALSE FALSE  
## 5 TRUE FALSE FALSE FALSE FALSE  
## 6 FALSE FALSE FALSE FALSE FALSE  
## 7 TRUE FALSE FALSE FALSE FALSE  
## 8 FALSE FALSE FALSE TRUE FALSE  
## 9 TRUE FALSE FALSE FALSE FALSE  
## 10 FALSE FALSE FALSE FALSE FALSE  
## 11 FALSE FALSE TRUE FALSE FALSE  
## 12 FALSE FALSE FALSE FALSE FALSE  
## 13 FALSE FALSE FALSE TRUE FALSE  
## 14 FALSE FALSE FALSE FALSE FALSE  
## 15 FALSE FALSE FALSE TRUE FALSE  
## 16 FALSE FALSE FALSE FALSE FALSE  
## 17 FALSE FALSE TRUE TRUE FALSE  
## 18 FALSE FALSE FALSE FALSE TRUE  
## 19 FALSE FALSE FALSE TRUE FALSE  
## 20 FALSE FALSE TRUE FALSE FALSE

It gives us duplicated rows though, as many directors have produced several entries. So a better answer would be to us the function distinct() after arrange. And even best, to add the argument .keep\_all=TRUE to keep all the columns, and then to select only the relevant columns.

imdb %>%   
 arrange(desc(birthYear)) %>%  
 distinct(director, .keep\_all = TRUE) %>%  
 select(director, birthYear) %>%  
 slice(1:20)

## director birthYear  
## 1 Matt Horton 2000  
## 2 Roman Lochmann 1999  
## 3 Joshua Widgren 1998  
## 4 Jack Spring 1996  
## 5 Danny Wu 1996  
## 6 Quinn Shephard 1995  
## 7 Lauren Southern 1995  
## 8 Paul Raschid 1993  
## 9 Tamara Kotevska 1993  
## 10 Annabelle Attanasio 1993  
## 11 Hannah Marks 1993  
## 12 Robert Kouba 1992  
## 13 Nariman Aliev 1992  
## 14 James Cullen Bressack 1992  
## 15 Emily Hagins 1992  
## 16 Aleksander Pietrzak 1992  
## 17 Lucien Flores 1992  
## 18 Helene Hegemann 1992  
## 19 Castille Landon 1991  
## 20 Alex R. Wagner 1991

You were not expected to give this full answer though

**Exercise 2. Identify and correct the four mistakes that I made in the command below, to obtain the median duration of all the movies released after the year 2000**

imdb %>% filter(imdb, type=“movie” & year>2000) %>% sumarize(medianDuration = median(length)

## Answer 2

1. either imdb should be removed from filter, or the first line should be deleted
2. the condition of the filter function needs a double equals
3. the function summarise was misspelled
4. the parenthesis of the function summarise was not closed

Here is the corrected command:

imdb %>%  
 filter(type=="movie" & year>2000) %>%  
 summarise(medianDuration = median(length))

## medianDuration  
## 1 100

**Exercise 3. Which are the earliest released titles for each type of entry**

## Answer 3

Using pipes, we groupe the data by type and then apply filter()

imdb %>%   
 group\_by(type) %>%  
 filter(year==min(year))

## # A tibble: 9 x 17  
## # Groups: type [9]  
## title type year length numVotes averageRating director birthYear animation  
## <chr> <fct> <dbl> <dbl> <int> <dbl> <chr> <dbl> <lgl>   
## 1 Quee~ video 1975 6 1722 8.8 Bruce G~ 1940 FALSE   
## 2 The ~ movie 1906 70 591 6.1 Charles~ 1868 FALSE   
## 3 The ~ tvSp~ 1967 82 692 7.7 Philip ~ 1928 FALSE   
## 4 Cras~ tvMo~ 1954 78 759 3.1 Holling~ 1910 FALSE   
## 5 Gold~ vide~ 1997 130 5001 9.1 Martin ~ 1971 FALSE   
## 6 Gadk~ tvSh~ 1956 19 744 7.6 Vladimi~ 1916 TRUE   
## 7 Quat~ tvMi~ 1958 30 866 8 Rudolph~ 1904 FALSE   
## 8 Tom ~ tvSe~ 1940 8 900 9.2 Chuck J~ 1912 TRUE   
## 9 Pass~ short 1874 1 1209 6.9 P.J.C. ~ 1824 FALSE   
## # ... with 8 more variables: action <lgl>, adventure <lgl>, comedy <lgl>,  
## # documentary <lgl>, fantasy <lgl>, romance <lgl>, sci\_fi <lgl>,  
## # thriller <lgl>

**Exercise 4. Produce a list of Thriller TV Series who received more than 10000, ordered from best to worst average ratings**

## Answer 4

We filter the data to only retrieve the TV Series of genre thrille having received more than 10000 votes. Then we order the rows using arrange().

imdb %>%  
 filter(thriller==TRUE & type=="tvSeries" & numVotes>10000) %>%  
 arrange(desc(averageRating)) %>%  
 slice(1:20)

## title type year length numVotes averageRating  
## 1 Koombiyo tvSeries 2017 20 16098 9.6  
## 2 Breaking Bad tvSeries 2008 49 1391850 9.5  
## 3 The Wire tvSeries 2002 59 274896 9.3  
## 4 Fargo tvSeries 2014 53 309806 8.9  
## 5 Black Mirror tvSeries 2011 60 418123 8.8  
## 6 Oz tvSeries 1997 55 86865 8.7  
## 7 Gomorrah tvSeries 2014 55 27555 8.7  
## 8 The Shield tvSeries 2002 47 70271 8.7  
## 9 Special OPS tvSeries 2020 49 13144 8.6  
## 10 Sons of Anarchy tvSeries 2008 45 250272 8.6  
## 11 Mindhunter tvSeries 2017 60 199564 8.6  
## 12 Bron/Broen tvSeries 2011 60 55899 8.6  
## 13 Ezel tvSeries 2009 90 25219 8.6  
## 14 Mr. Robot tvSeries 2015 49 321246 8.5  
## 15 The Handmaid's Tale tvSeries 2017 60 160517 8.5  
## 16 Asur: Welcome to Your Dark Side tvSeries 2020 43 20696 8.4  
## 17 Top Boy tvSeries 2011 60 11855 8.4  
## 18 Taboo tvSeries 2017 59 112091 8.4  
## 19 Babylon Berlin tvSeries 2017 45 16804 8.4  
## 20 Ozark tvSeries 2017 60 170435 8.4  
## director birthYear animation action adventure comedy documentary  
## 1 Lakmal Darmarathna NA FALSE FALSE FALSE FALSE FALSE  
## 2 Thomas Schnauz NA FALSE FALSE FALSE FALSE FALSE  
## 3 Joe Chappelle NA FALSE FALSE FALSE FALSE FALSE  
## 4 Dearbhla Walsh NA FALSE FALSE FALSE FALSE FALSE  
## 5 Toby Haynes NA FALSE FALSE FALSE FALSE FALSE  
## 6 Nick Gomez 1963 FALSE FALSE FALSE FALSE FALSE  
## 7 Francesca Comencini 1961 FALSE FALSE FALSE FALSE FALSE  
## 8 Guy Ferland 1966 FALSE FALSE FALSE FALSE FALSE  
## 9 Shivam Nair NA FALSE TRUE FALSE FALSE FALSE  
## 10 Billy Gierhart NA FALSE FALSE FALSE FALSE FALSE  
## 11 David Fincher 1962 FALSE FALSE FALSE FALSE FALSE  
## 12 Lisa Siwe 1968 FALSE FALSE FALSE FALSE FALSE  
## 13 Uluç Bayraktar NA FALSE FALSE FALSE FALSE FALSE  
## 14 Christoph Schrewe 1964 FALSE FALSE FALSE FALSE FALSE  
## 15 Mike Barker 1965 FALSE FALSE FALSE FALSE FALSE  
## 16 Oni Sen NA FALSE FALSE FALSE FALSE FALSE  
## 17 Yann Demange 1977 FALSE FALSE FALSE FALSE FALSE  
## 18 Anders Engström 1963 FALSE FALSE FALSE FALSE FALSE  
## 19 Henk Handloegten 1968 FALSE FALSE FALSE FALSE FALSE  
## 20 Daniel Sackheim NA FALSE FALSE FALSE FALSE FALSE  
## fantasy romance sci\_fi thriller  
## 1 FALSE FALSE FALSE TRUE  
## 2 FALSE FALSE FALSE TRUE  
## 3 FALSE FALSE FALSE TRUE  
## 4 FALSE FALSE FALSE TRUE  
## 5 FALSE FALSE TRUE TRUE  
## 6 FALSE FALSE FALSE TRUE  
## 7 FALSE FALSE FALSE TRUE  
## 8 FALSE FALSE FALSE TRUE  
## 9 FALSE FALSE FALSE TRUE  
## 10 FALSE FALSE FALSE TRUE  
## 11 FALSE FALSE FALSE TRUE  
## 12 FALSE FALSE FALSE TRUE  
## 13 FALSE FALSE FALSE TRUE  
## 14 FALSE FALSE FALSE TRUE  
## 15 FALSE FALSE TRUE TRUE  
## 16 FALSE FALSE FALSE TRUE  
## 17 FALSE FALSE FALSE TRUE  
## 18 FALSE FALSE FALSE TRUE  
## 19 FALSE FALSE FALSE TRUE  
## 20 FALSE FALSE FALSE TRUE

**Exercise 5. What are the minimum, average and maximum age of a movie director releasing a movie in the imdb dataset? (you will need to add na.rm=T in your summary functions to deal with the entries where the year of birth of the director is missing)**

## Answer 5

For this exercise, we need to create a new variable that is the age of the director at the time of release. We do that with mutate. We then filter the data to only keep the entries of type movie (we could have placed this step first though). The order did not matter here. Finally we use summarise to calculate the requested summary statistics. Since there are missing values in the column birthYear, there are also missing values in the new column age, and so we add the argument na.rm=TRUE to get rid of these missing values.

imdb %>%  
 mutate(age=year-birthYear) %>%  
 filter(type=="movie") %>%  
 summarise(minAge=min(age, na.rm=TRUE), meanAge=mean(age, na.rm=TRUE), maxAge=max(age, na.rm=TRUE))

## minAge meanAge maxAge  
## 1 16 45.28538 104

So there was a director who directed a movie at the age of 16, and one who directed a movie at age 104! Let’s find who they are. To do so, I’m just copying the above command, but changing the summarise into a filter. I use the logical operator OR | to keep both, the director who directed a movie at the youngest and at the oldest age. At the end, I use select to only show the columns I’m interested in

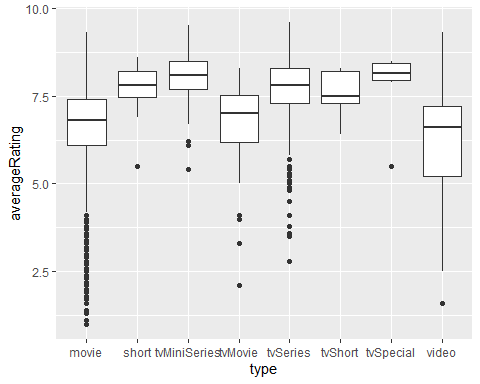
imdb %>%  
 mutate(age=year-birthYear) %>%  
 filter(type=="movie") %>%  
 filter(age==min(age, na.rm=TRUE) | age==max(age, na.rm=TRUE))%>%  
 select(director, age, title, numVotes, year)

## director age title numVotes year  
## 1 Roman Lochmann 16 Bruder vor Luder 788 2015  
## 2 Manoel de Oliveira 104 Gebo et l'ombre 563 2012

**Exercise 6. Generate a boxplot of average rating by type of entry having received more than 10000 votes**

We filter our rows, and make our boxplot!

imdb %>%  
 filter(numVotes>10000)%>%  
 ggplot(aes(x=type, y=averageRating))+  
 geom\_boxplot()



**Exercise 7. In three parts:**

1. **Find who is the worst director of romantic comedy movies, only counting directors who made at least 5 romantic comedies that received at least 5000 votes.**
2. **Find the worst rated movie (of any genre) that this director has released.**
3. **Watch this movie.**

For the first part, we now that we will need to group our data by director. So I start with that. Then we only want to count romantic comedy movies that received at least 5000 votes, so we filter the data accordingly. Here,the order of the two first commands don’t matter. Then I wan to keep only the directors who have made at least 5 of these movies. So I need to count the number of entries for each director, using summarise() and the function n(). Then I apply a filter to filter out the directors who directed less than 5 such movies. Finally I need to find the worst of these directors. Well assuming that worst mean that the mean of average rating is the worst, Ineed to calculate this average. We can add this calculation is the summarise command. Finally let’s see who is the worst by applying a last filter to our data.

imdb %>%  
 group\_by(director) %>%  
 filter(romance==TRUE & comedy==TRUE & type=="movie" & numVotes>5000) %>%  
 summarize(n=n(), meanRating=mean(averageRating))%>%  
 filter(n>=5)%>%  
 filter(meanRating==min(meanRating))

## `summarise()` ungrouping output (override with `.groups` argument)

## # A tibble: 1 x 3  
## director n meanRating  
## <chr> <int> <dbl>  
## 1 Tyler Perry 6 4.92

For the second part of the question, we want to only look at the entries of type movies by this director. So we can use filter. Then we could use filter again to keep the movie with worst average, but I prefer using arrange here to get an idea of his other entries.

imdb %>%  
 filter(director=="Tyler Perry" & type=="movie") %>%  
 arrange(averageRating) %>%   
 slice(1:20)

## title type year length numVotes  
## 1 Boo 2! A Madea Halloween movie 2017 101 3959  
## 2 Madea's Neighbors from Hell movie 2014 138 571  
## 3 Meet the Browns movie 2008 100 6472  
## 4 Madea Gets a Job movie 2013 145 822  
## 5 Tyler Perry's a Madea Family Funeral movie 2019 109 3605  
## 6 Madea Goes to Jail movie 2009 103 11244  
## 7 I Can Do Bad All by Myself movie 2009 113 9453  
## 8 Tyler Perry's Boo! A Madea Halloween movie 2016 103 7504  
## 9 Why Did I Get Married Too? movie 2010 121 8965  
## 10 Tyler Perry's Madea's Big Happy Family movie 2011 106 7369  
## 11 Nobody's Fool movie 2018 110 6087  
## 12 Madea's Witness Protection movie 2012 114 9955  
## 13 Tyler Perry's A Madea Christmas movie 2013 100 5739  
## 14 Temptation: Confessions of a Marriage Counselor movie 2013 111 11343  
## 15 Madea's Family Reunion movie 2006 107 10467  
## 16 The Single Moms Club movie 2014 111 5779  
## 17 Tyler Perry's Good Deeds movie 2012 129 7355  
## 18 Daddy's Little Girls movie 2007 100 6931  
## 19 The Family That Preys movie 2008 111 6717  
## 20 Tyler Perry's Acrimony movie 2018 120 7655  
## averageRating director birthYear animation action adventure comedy  
## 1 3.8 Tyler Perry 1969 FALSE FALSE FALSE TRUE  
## 2 4.2 Tyler Perry 1969 FALSE FALSE FALSE TRUE  
## 3 4.4 Tyler Perry 1969 FALSE FALSE FALSE TRUE  
## 4 4.4 Tyler Perry 1969 FALSE FALSE FALSE TRUE  
## 5 4.4 Tyler Perry 1969 FALSE FALSE FALSE TRUE  
## 6 4.5 Tyler Perry 1969 FALSE FALSE FALSE TRUE  
## 7 4.5 Tyler Perry 1969 FALSE FALSE FALSE TRUE  
## 8 4.6 Tyler Perry 1969 FALSE FALSE FALSE TRUE  
## 9 4.7 Tyler Perry 1969 FALSE FALSE FALSE TRUE  
## 10 4.7 Tyler Perry 1969 FALSE FALSE FALSE TRUE  
## 11 4.8 Tyler Perry 1969 FALSE FALSE FALSE TRUE  
## 12 4.9 Tyler Perry 1969 FALSE FALSE FALSE TRUE  
## 13 4.9 Tyler Perry 1969 FALSE FALSE FALSE TRUE  
## 14 5.0 Tyler Perry 1969 FALSE FALSE FALSE FALSE  
## 15 5.2 Tyler Perry 1969 FALSE FALSE FALSE TRUE  
## 16 5.3 Tyler Perry 1969 FALSE FALSE FALSE TRUE  
## 17 5.5 Tyler Perry 1969 FALSE FALSE FALSE TRUE  
## 18 5.7 Tyler Perry 1969 FALSE FALSE FALSE FALSE  
## 19 5.7 Tyler Perry 1969 FALSE FALSE FALSE FALSE  
## 20 5.8 Tyler Perry 1969 FALSE FALSE FALSE FALSE  
## documentary fantasy romance sci\_fi thriller  
## 1 FALSE FALSE FALSE FALSE FALSE  
## 2 FALSE FALSE FALSE FALSE FALSE  
## 3 FALSE FALSE TRUE FALSE FALSE  
## 4 FALSE FALSE FALSE FALSE FALSE  
## 5 FALSE FALSE FALSE FALSE FALSE  
## 6 FALSE FALSE FALSE FALSE FALSE  
## 7 FALSE FALSE FALSE FALSE FALSE  
## 8 FALSE FALSE FALSE FALSE FALSE  
## 9 FALSE FALSE TRUE FALSE FALSE  
## 10 FALSE FALSE FALSE FALSE FALSE  
## 11 FALSE FALSE TRUE FALSE FALSE  
## 12 FALSE FALSE FALSE FALSE FALSE  
## 13 FALSE FALSE TRUE FALSE FALSE  
## 14 FALSE FALSE FALSE FALSE TRUE  
## 15 FALSE FALSE TRUE FALSE FALSE  
## 16 FALSE FALSE FALSE FALSE FALSE  
## 17 FALSE FALSE TRUE FALSE FALSE  
## 18 FALSE FALSE TRUE FALSE FALSE  
## 19 FALSE FALSE FALSE FALSE FALSE  
## 20 FALSE FALSE TRUE FALSE TRUE

For the third part, if you really want to answer this question completely, I’ve found a streaming link that may not be totally legal: <https://ww2.123movieshub.dev/movies/show/31864/tyler-perrys-boo-2-a-madea-halloween-2017> But if I was you, I would skip this last part. Remember, there is no grade at the end.