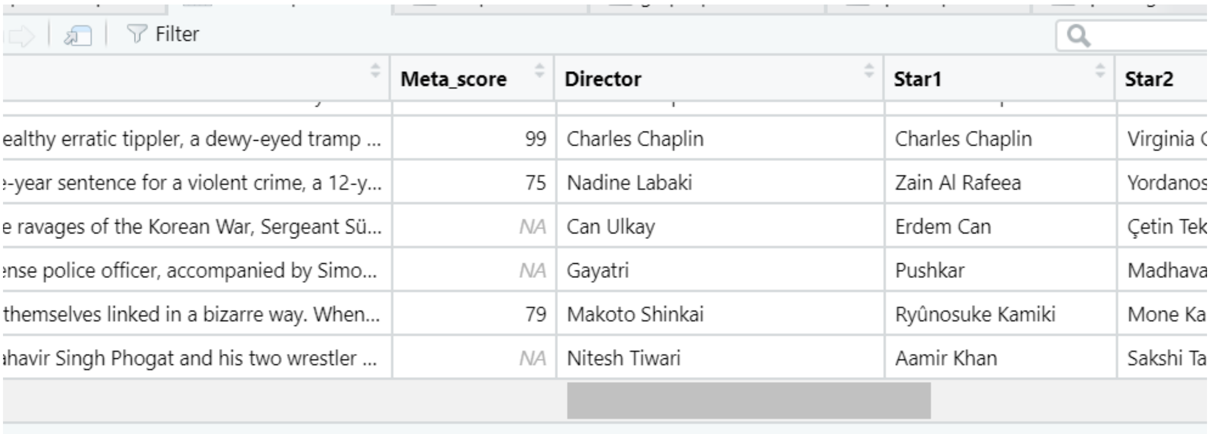
4. Data Visualization – VG question:

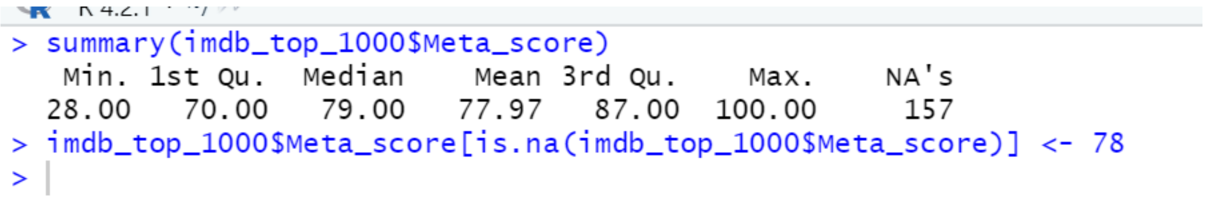
I spent so long trying to decide upon a dataset and the graphs to go with it for question 3 that I had less time than I would have liked for question 4. After university I went to film school and have written screenplays and advertisements on and off since then. It seemed natural for me to search for a dataset within the field. I went with the top 1000 films on imdb, mainly because the length was sufficient for the brief, though I have not exactly been wowed by the data on offer. Given my self-enforced time restrictions I have had to make do.

The dataset was not 'plot' ready, I found a number of rows with 'NA's'.



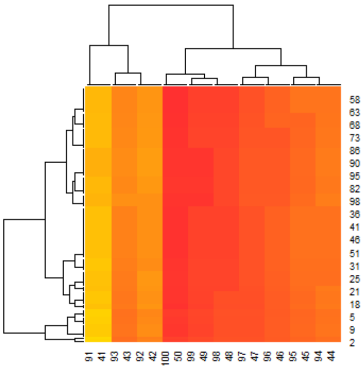
	Meta_score	Director	Star1	Star2
ealthy erratic tippler, a dewy-eyed tramp ...	99	Charles Chaplin	Charles Chaplin	Virginia (
;-year sentence for a violent crime, a 12-y...	75	Nadine Labaki	Zain Al Rafeea	Yordanos
e ravages of the Korean War, Sergeant Sü...	NA	Can Ulkay	Erdem Can	Çetin Tek
ːnse police officer, accompanied by Simo...	NA	Gayatri	Pushkar	Madhava
themselves linked in a bizarre way. When...	79	Makoto Shinkai	Ryûnosuke Kamiki	Mone Ka
ıhavir Singh Phogat and his two wrestler ...	NA	Nitesh Tiwari	Aamir Khan	Sakshi Ta

I used the same approach as in question 2, I got a mean for the row and replaced the missing values.



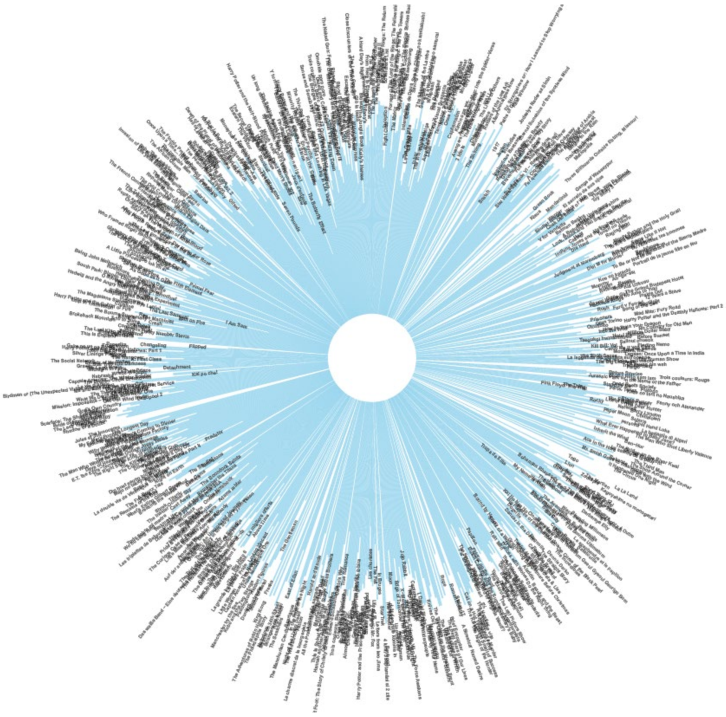
```
> summary(imdb_top_1000$Meta_score)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
  28.00   70.00   79.00   77.97   87.00  100.00     157 
> imdb_top_1000$Meta_score[is.na(imdb_top_1000$Meta_score)] <- 78
> 
```

With a thousand rows of data I considered it wise to start with a landscape view, in order to get an idea of what I was dealing with. Here are a couple of valiant but failed attempts to discover insights from a mess of mass:



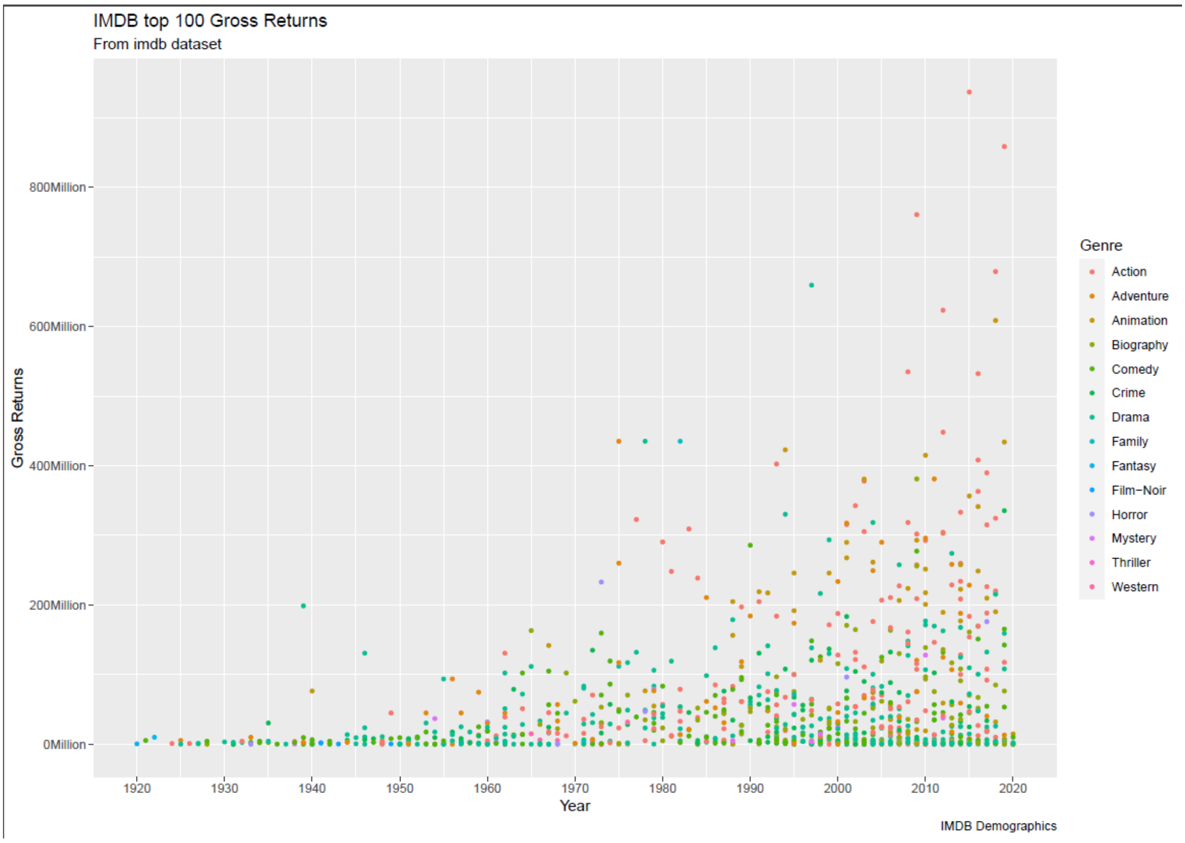
I started with a heatmap, but even with a thousand elements in the dataset it doesn't equate to much range when you consider the tens of thousands of rated films in the database.

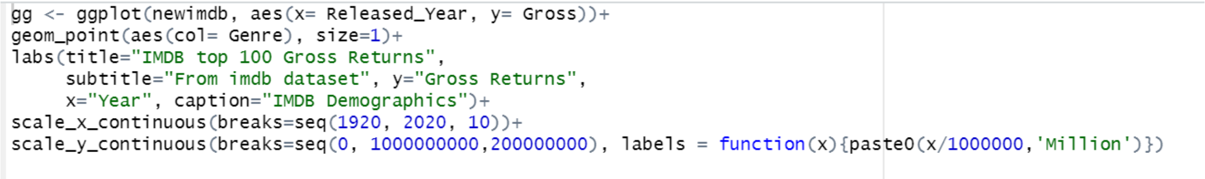
Attempted a circular plot that came out eye strainingly mental:



I dare you to search for movie titles. Unfortunately, in the confusion of trying to figure out what I had created and the need to create something that made sense I deleted the code that formed this aberration.

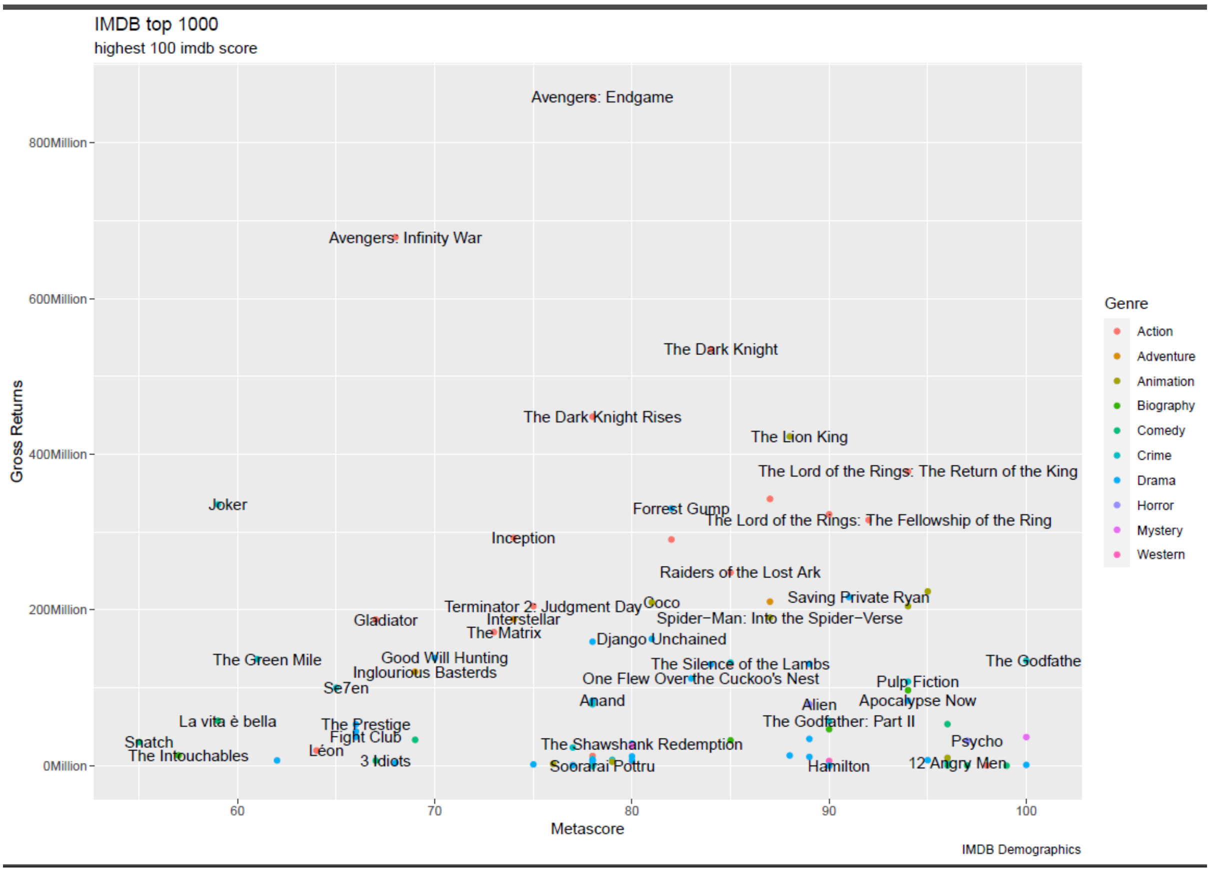
So I turned to the tried and tested, and created a scatterplot when compared 'Gross profit' to 'Release Year' and using genre as the key. The idea of using actual film names for this graph was a non-starter. The problem with the result is that the profits of older films have not been adjusted according to inflation, so the results are skewed heavily in favour of films released in the last twenty years or so.





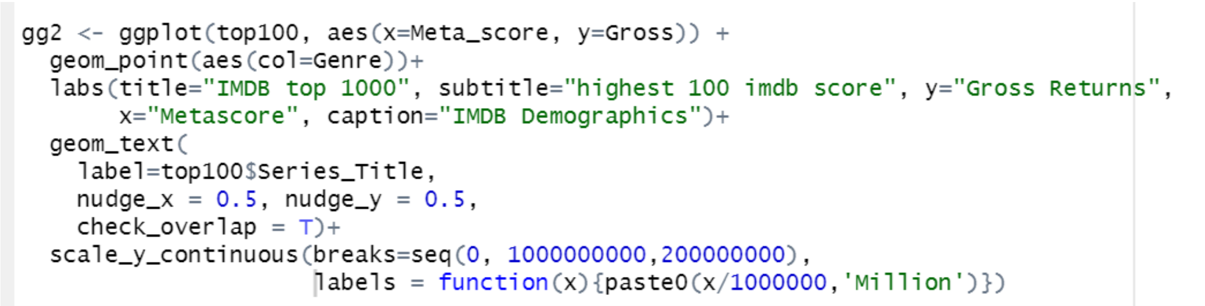
```
gg <- ggplot(newimdb, aes(x= Released_Year, y= Gross))+
geom_point(aes(col= Genre), size=1)+
labs(title="IMDB top 100 Gross Returns",
     subtitle="From imdb dataset", y="Gross Returns",
     x="Year", caption="IMDB Demographics")+
scale_x_continuous(breaks=seq(1920, 2020, 10))+
scale_y_continuous(breaks=seq(0, 1000000000,200000000), labels = function(x){paste0(x/1000000,'Million')})
```

I did take the top hundred grossing films and plotted their gross profit to their ‘Metascores’ (a much more accurate measure of their critical worth than the ‘IMDB rating’, in my opinion). In this graph I could include the names of the films and again used ‘genre’ as a legend. I see it as the most natural and significant way to divide large groups of movies.



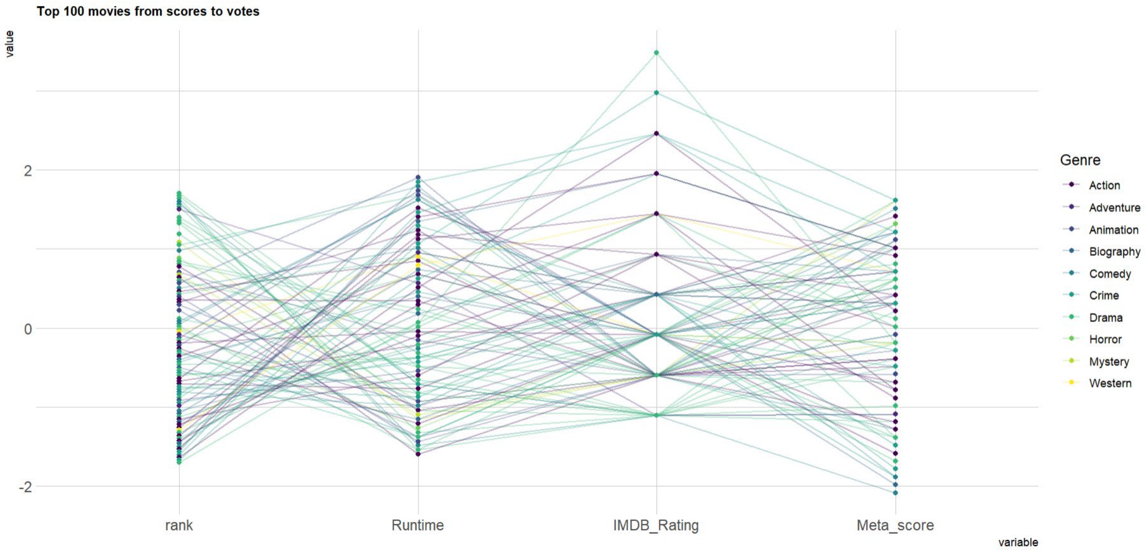
IMDB Demographics

The graph clearly shows the dominance of action movies at the top of the box office, and within the genre how indomitable the comic book has been over the past decades.

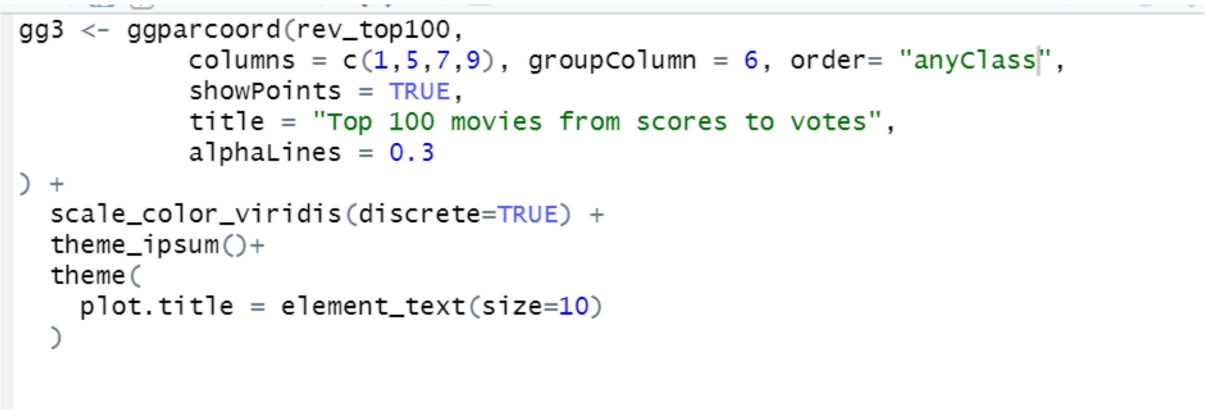


```
gg2 <- ggplot(top100, aes(x=Meta_score, y=Gross)) +
  geom_point(aes(col=Genre))+
  labs(title="IMDB top 1000", subtitle="highest 100 imdb score", y="Gross Returns",
       x="Metascore", caption="IMDB Demographics")+
  geom_text(
    label=top100$Series_Title,
    nudge_x = 0.5, nudge_y = 0.5,
    check_overlap = T)+
  scale_y_continuous(breaks=seq(0, 1000000000,200000000),
                     labels = function(x){paste0(x/1000000,'Million')})
```

I wanted to find a way to show how people voting compared to critic’s votes, which is the essential difference with the IMDB and metascore rating system. This was the graph that I tried out to show the difference. I also included a row for film length to see if that reflected any opinions on the films themselves. Genre again indexes, but the results are far from clear. What I think can be seen quite clearly is how critics tend to judge films more harshly than the average cinema goer.

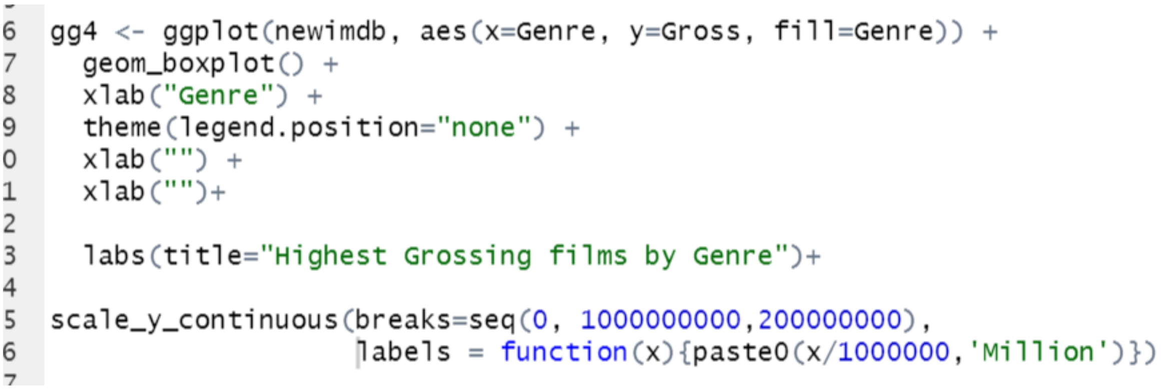


The third row is the public voting, the last row the critic's.



```r
gg3 <- ggparcoord(rev_top100,
           columns = c(1,5,7,9), groupColumn = 6, order= "anyClass",
           showPoints = TRUE,
           title = "Top 100 movies from scores to votes",
           alphaLines = 0.3
) +
  scale_color_viridis(discrete=TRUE) +
  theme_ipsum()+
  theme(
    plot.title = element_text(size=10)
  )
```

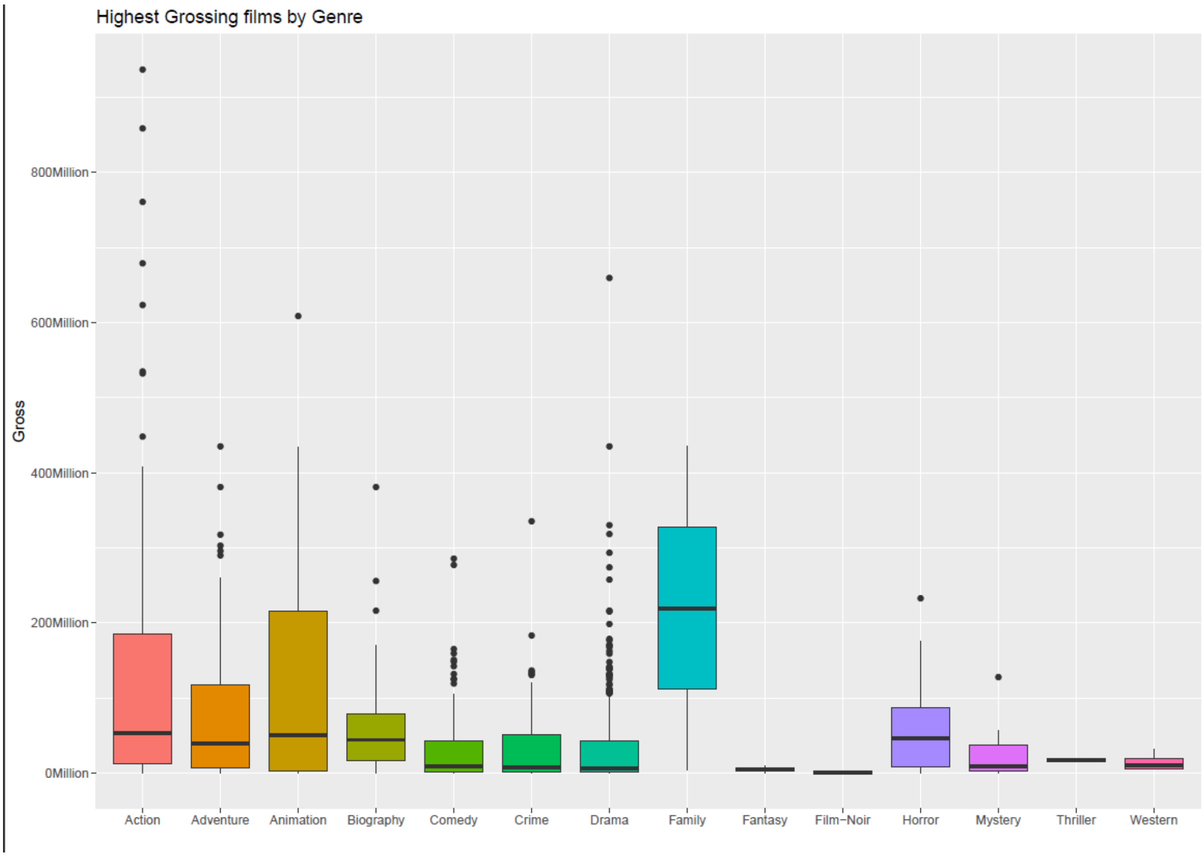
'I will end my graphing of the thousand highest rated movies on IMDB with a boxplot, which shows much more clearly than the scatterplot how successful different genres have been over the decades. What stands out here is that although actions movies have outliers that are more successful on orders of magnitude from most other films, it is the family film which is the safest bet from an investment perspective, as they're most likely to break even or make a profit.



```r
gg4 <- ggplot(newimdb, aes(x=Genre, y=Gross, fill=Genre)) +
  geom_boxplot() +
  xlab("Genre") +
  theme(legend.position="none") +
  xlab("") +
  xlab("")+

  labs(title="Highest Grossing films by Genre")+

scale_y_continuous(breaks=seq(0, 1000000000,200000000),
                   labels = function(x){paste0(x/1000000,'Million')})
```



I hope this has been an enlightening look into how successful different genres of film have been throughout the world. It was incredibly interesting trying out different techniques and plotting styles. I just wish I'd had more time and was able to experiment more. This was a fun start though.