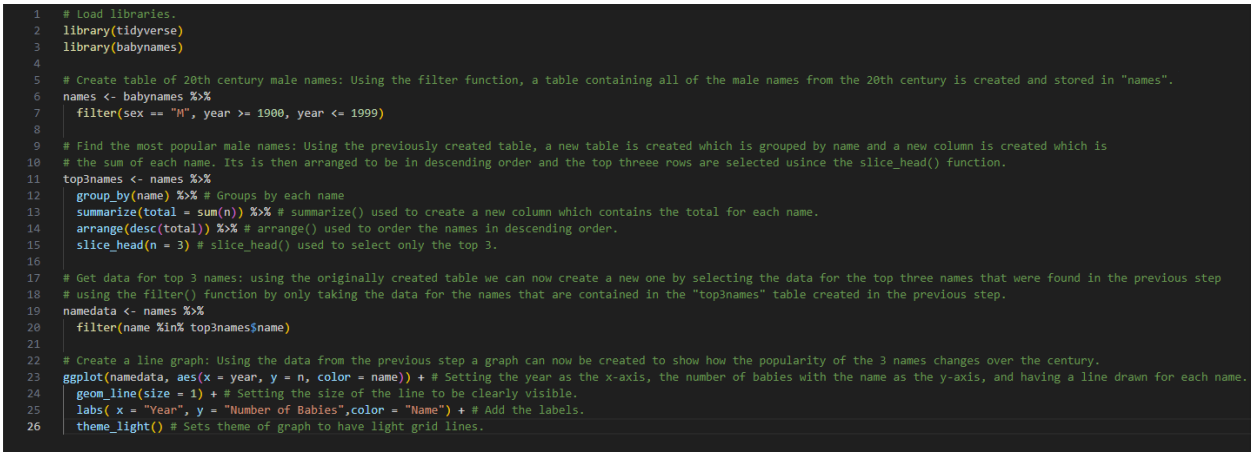
R Assignment: John Harrow

Code and files can be found on github, didn't know how to submit multiple files on turnitin.

https://github.com/JohnHarrow/AC50002-AssignmentR

Part 1



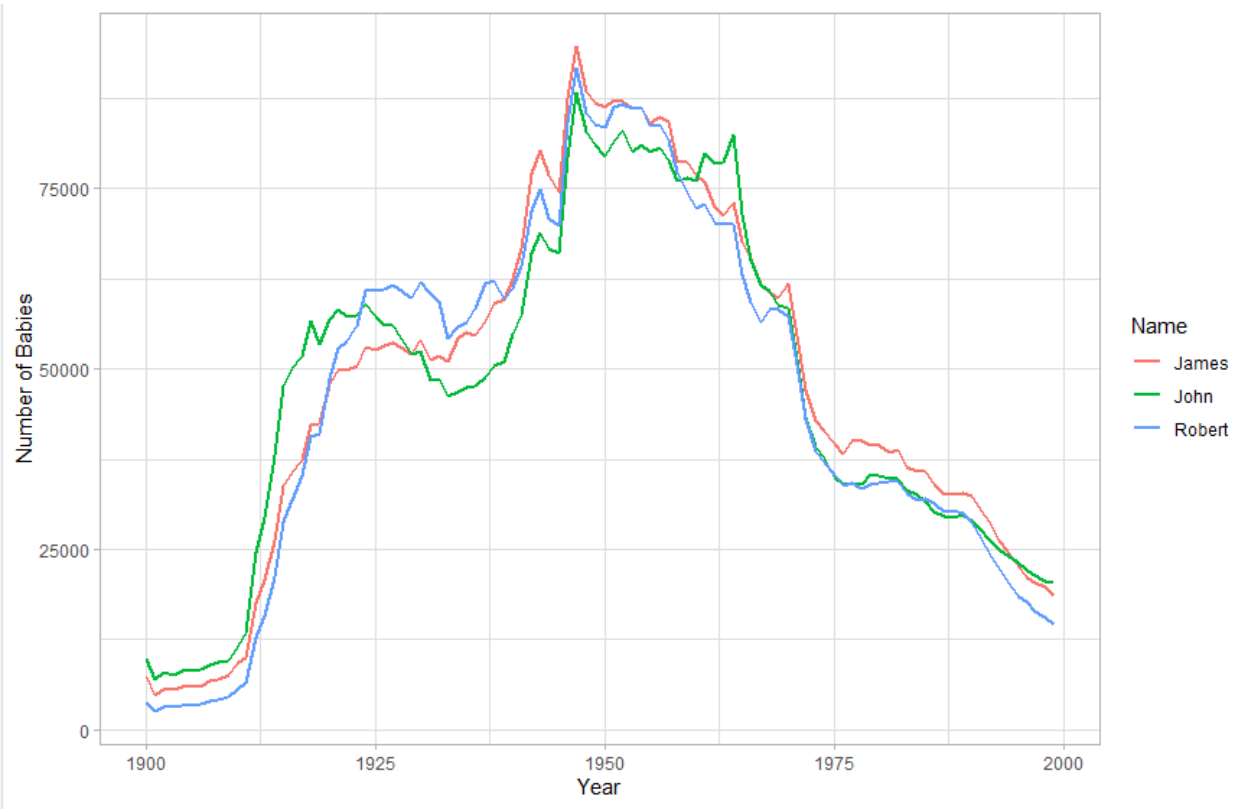
```r
# Load libraries.
library(tidyverse)
library(babynames)

# Create table of 20th century male names: Using the filter function, a table containing all of the male names from the 20th century is created and stored in "names".
names <- babynames %>%
  filter(sex == "M", year >= 1900, year <= 1999)

# Find the most popular male names: Using the previously created table, a new table is created which is grouped by name and a new column is created which is
# the sum of each name. Its is then arranged to be in descending order and the top threee rows are selected usince the slice_head() function.
top3names <- names %>%
  group_by(name) %>% # Groups by each name
  summarize(total = sum(n)) %>% # summarize() used to create a new column which contains the total for each name.
  arrange(desc(total)) %>% # arrange() used to order the names in descending order.
  slice_head(n = 3) # slice_head() used to select only the top 3.

# Get data for top 3 names: using the originally created table we can now create a new one by selecting the data for the top three names that were found in the previous step
# using the filter() function by only taking the data for the names that are contained in the "top3names" table created in the previous step.
namedata <- names %>%
  filter(name %in% top3names$name)

# Create a line graph: Using the data from the previous step a graph can now be created to show how the popularity of the 3 names changes over the century.
ggplot(namedata, aes(x = year, y = n, color = name)) + # Setting the year as the x-axis, the number of babies with the name as the y-axis, and having a line drawn for each name.
  geom_line(size = 1) + # Setting the size of the line to be clearly visible.
  labs( x = "Year", y = "Number of Babies",color = "Name") + # Add the labels.
  theme_light() # Sets theme of graph to have light grid lines.
```



Part 2

The dataset that I found that I thought would be interesting to explore is the MovieLens dataset. This dataset consists of information about movies and how people have rated the movies. The dataset was created by the GroupLens Research team at the University of Minnesota. The data was collected to be used for things like training recommendation algorithms.

The dataset was initially two files, movies.csv which contained information about the movies, and ratings.csv which contained user ratings for movies. Both sets of data contained the 'movieId' column so they were joined to create one table that contained the ratings and the information about the movies. The code used to join the files can be found at [1].

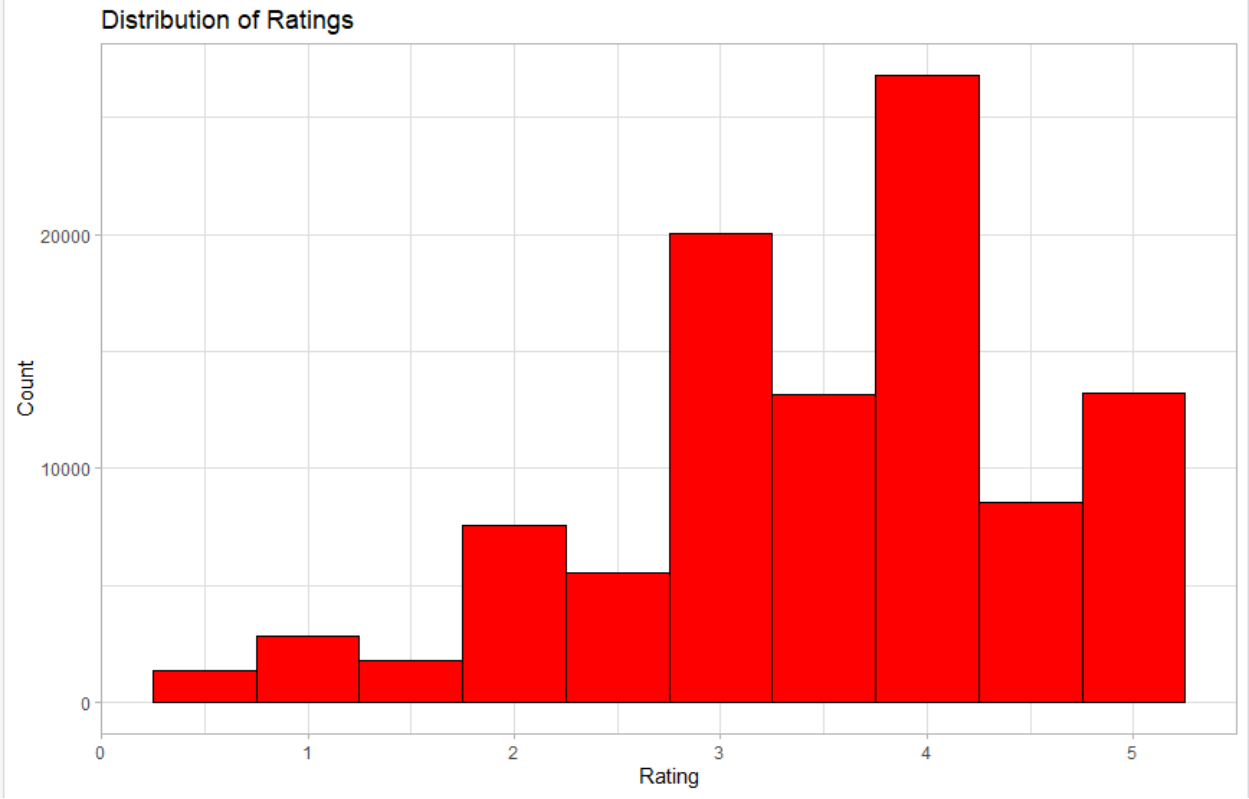
userId	movieId	rating	timestamp	title	genres
1	1	4	964982703	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy
1	3	4	964981247	Grumpier Old Men (1995)	Comedy|Romance
1	6	4	964982224	Heat (1995)	Action|Crime|Thriller

The dataset consists of a ‘userId’ which is the user who made the review, a ‘movieId’ which is the unique ID of the movie that has been reviewed, a ‘rating’ which is a score from 1 to 5, a ‘timestamp’ which is a unix timestamp of when the review was created, a ‘title’ which is just the name of the movie with the release year, and ‘genres’ which are all the genres that the movie fits into. The genres are split by a pipe so they can be easily separated to do things like list all reviews for movies in specific genres.

The first idea for what to use this dataset for was to find out the top rated movies. This was done by grouping by the movie name and finding the average rating then ordering the results. The code can be found at [2] and the output can be seen below.

	title	average_rating	count
	<chr>	<dbl>	<int>
1	Shawshank Redemption, The (1994)	4.43	317
2	Godfather, The (1972)	4.29	192
3	Fight Club (1999)	4.27	218
4	Godfather: Part II, The (1974)	4.26	129
5	Departed, The (2006)	4.25	107
6	Goodfellas (1990)	4.25	126
7	Casablanca (1942)	4.24	100
8	Dark Knight, The (2008)	4.24	149
9	Usual Suspects, The (1995)	4.24	204
10	Princess Bride, The (1987)	4.23	142

The next idea was to try and see the distribution of all the ratings as it could be helpful for analysing rating behaviour to see if the results are skewed in any way. The code can be found at [3] and can be seen below.



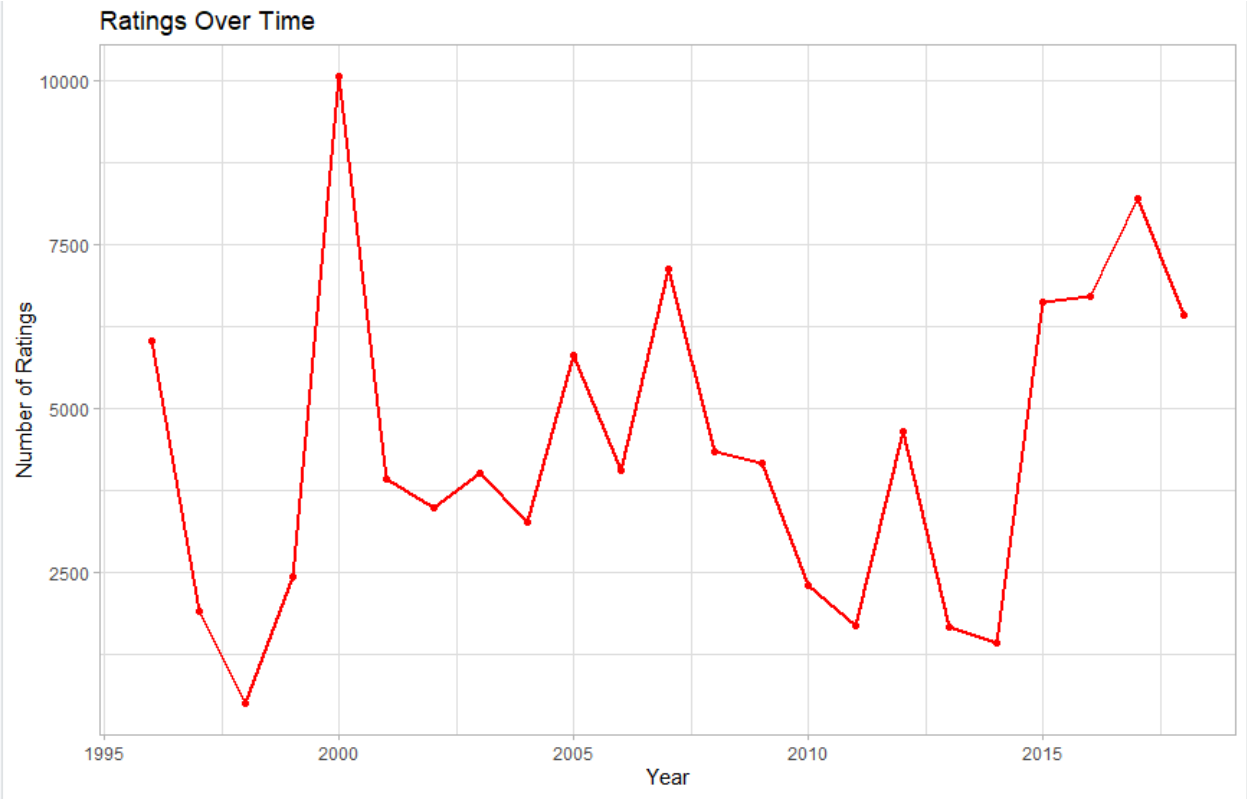
From this graph you can see that 4 is the most given rating closely followed by 3 which could suggest that people are hesitant to give movies they like a rating of 5 even if they enjoyed them a lot.

The next idea was to find out how many movies are in each genre. This was slightly more difficult as the genres are all contained in a single column but are separated by "|" which makes it possible to split the column into separate rows. The code can be found at [4].

genres	count
Drama	41928
Comedy	39053
Action	30635
Thriller	26452
Adventure	24161

The next idea was to see how the number of ratings changed over time as this could be useful to see how the activity of user ratings changes. This was slightly more difficult due to the timestamp having to be changed into date format so the

year could be extracted. The code for this can be found at [5] and the graph created below.



Overall working with this dataset gave me a better idea of how things like recommendation algorithms could be created and how user reviews and activity can be used to not only recommend other movies to that user but to other users with similar activity to them. It links well to what we are learning about in the dev-ops module about how services like Netflix are created which is one of the reasons the dataset stood out.