Kelly Garrett & Lichen Zhen
INF 385T: Data Wrangling
May 3, 2017

Project Workflow & Report

Disney Analysis

Introduction

This project seeks to find the relationship between box office gross and MPAA ratings in Disney movies. The common assumption is that G-rated movies generate the most revenue because the largest portion of viewers are allowed admittance to these movies, children and adults alike. Does this assumption hold true?

Link to the workflow:

Workflow Path:
~lzhen/project/report/

Database Path:

lzhen/lzhen_disney/

Data Sources

Our project includes five CSVs from four different sources, all of which we found in the form of HTML tables.

- Sugarcane, "Walt Disney Animation Studios Films"
    - The link provides a list of Disney animated movies and the hero/villain character names in each movie.
- The Numbers, "Movies Released by Walt Disney"
    - It is a chart and provides a list of Disney movies, and their genre, gross, and MPAA ratings.
- Wikipedia, "List of Disney animated universe characters"
    - The link provides a complete list of Disney characters and their voice actors.
- Wikipedia, "List of Walt Disney Animation Studios films"
    - The link provides a list of Disney animated movies and the director of each movie.
- Wikipedia, "Annual gross revenues of The Walt Disney Company"
    - This is a Disney financial data chart which contains annual gross revenues by sections (includes studio entertainment, parks and resorts, etc.) from 1991-2016. The data are collected from the Disney annual report.

We used import.io to scrape the data and convert it to CSV. We learned of this powerful tool from the workshop “Scraping Twitter with Import.io” hosted by UT Libraries in February. Below, we have included screenshots of the first 10 lines of each CSV:

Disney Movies Total Gross

```
lzhen@holden ~/assign/project $ head disney_movies_total_gross.csv
"movie_title","release_date","genre","MPAA_rating","total_gross","inflation_adju
sted_gross"
"Snow White and the Seven Dwarfs","Dec 21, 1937","Musical","G","$184,925,485","$
5,228,953,251"
"Pinocchio","Feb 9, 1940","Adventure","G","$84,300,000","$2,188,229,052"
"Fantasia","Nov 13, 1940","Musical","G","$83,320,000","$2,187,090,808"
"Song of the South","Nov 12, 1946","Adventure","G","$65,000,000","$1,078,510,579
"
"Cinderella","Feb 15, 1950","Drama","G","$85,000,000","$920,608,730"
"20,000 Leagues Under the Sea","Dec 23, 1954","Adventure","","$28,200,000","$528
,279,994"
"Lady and the Tramp","Jun 22, 1955","Drama","G","$93,600,000","$1,236,035,515"
"Sleeping Beauty","Jan 29, 1959","Drama","","$9,464,608","$21,505,832"
"101 Dalmatians","Jan 25, 1961","Comedy","G","$153,000,000","$1,362,870,985"
lzhen@holden ~/assign/project $
```

Disney Revenues

```
lzhen@holden ~/assign/project $ head disney_revenue_1991-2016.csv
Year,Studio Entertainment[NI 1],Disney Consumer Products[NI 2],Disney Interactive[NI
3][Rev 1],Walt Disney Parks and Resorts,Disney Media Networks,Total
1991,2593,724,,2794,,6111
1992,3115,1081,,3306,,7502
1993,3673.4,1415.1,,3440.7,,8529
1994,4793,1798.2,,3463.6,359,10414
1995,6001.5,2150,,3959.8,414,12525
1996,,,,4502,"4,142",18739
1997,6981,3782,174,5014,6522,22473
1998,6849,3193,260,5532,7142,22976
1999,6548,3030,206,6106,7512,23402
lzhen@holden ~/assign/project $
```

Disney Characters

```
lzhen@holden ~/assign/project $ head disney-characters.csv
movie_title,release_date,hero,villian,song
"
Snow White and the Seven Dwarfs","December 21, 1937",Snow White,Evil Queen,Some Day M
y Prince Will Come
"
Pinocchio","February 7, 1940",Pinocchio,Stromboli,When You Wish upon a Star
"
Fantasia","November 13, 1940",,Chernabog,
Dumbo,"October 23, 1941",Dumbo,Ringmaster,Baby Mine
"
Bambi","August 13, 1942",Bambi,Hunter,Love Is a Song
lzhen@holden ~/assign/project $
```

Disney Directors

```
lzhen@holden ~/assign/project $ head disney-director.csv
name,director
Snow White and the Seven Dwarfs,David Hand
Pinocchio,Ben Sharpsteen
Fantasia,full credits
Dumbo,Ben Sharpsteen
Bambi,David Hand
Saludos Amigos,Jack Kinney
The Three Caballeros,Norman Ferguson
Make Mine Music,Jack Kinney
Fun and Fancy Free,Jack Kinney
lzhen@holden ~/assign/project $
```

Disney Voice Actors

```
lzhen@holden ~/assign/project $ head disney-voice-actors.csv
character,voice-actor,movie
Abby Mallard,Joan Cusack,Chicken Little
Abigail Gabble,Monica Evans,The Aristocats
Abis Mal,Jason Alexander,The Return of Jafar
Abu,Frank Welker,Aladdin
Achilles,None,The Hunchback of Notre Dame
Adella,Sherry Lynn,The Little Mermaid
Adorabeezle Winterpop,None,Wreck-It Ralph
The Agent,Greg Germann,Bolt
Agent Wendy Pleakley,Kevin McDonald,Lilo & Stitch
lzhen@holden ~/assign/project $
```

Workflow

After finding our data and using import.io, we modeled the data with an ER diagram, relational vocabulary, and sample tables. In the end, we ended up with eight tables: characters, characters_movies, movies, movies_studios, studios, studio_revenues, roles, and people. Then we created the structure of the database in phpMyAdmin and imported the CSV data into the database using Python. In the Python script we also transformed and cleaned up the data. For example, we transformed our dates using strptime.

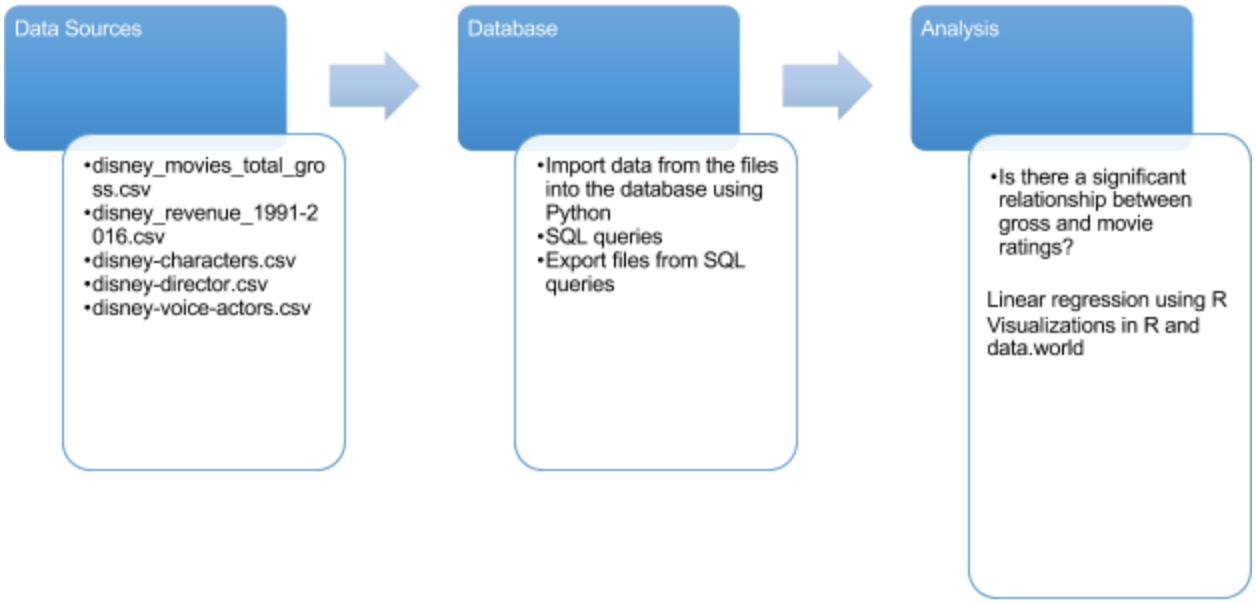
```
row["release_date"] = datetime.datetime.strptime(row["release_date"], "%B %d, %Y")
```

Deleted the dollar sign:

```
row["total_gross"] = re.sub('[\$,]','',row["total_gross"])
row["inflation_adjusted_gross"] = re.sub('[\$,]','',row["inflation_adjusted_gross"])
```

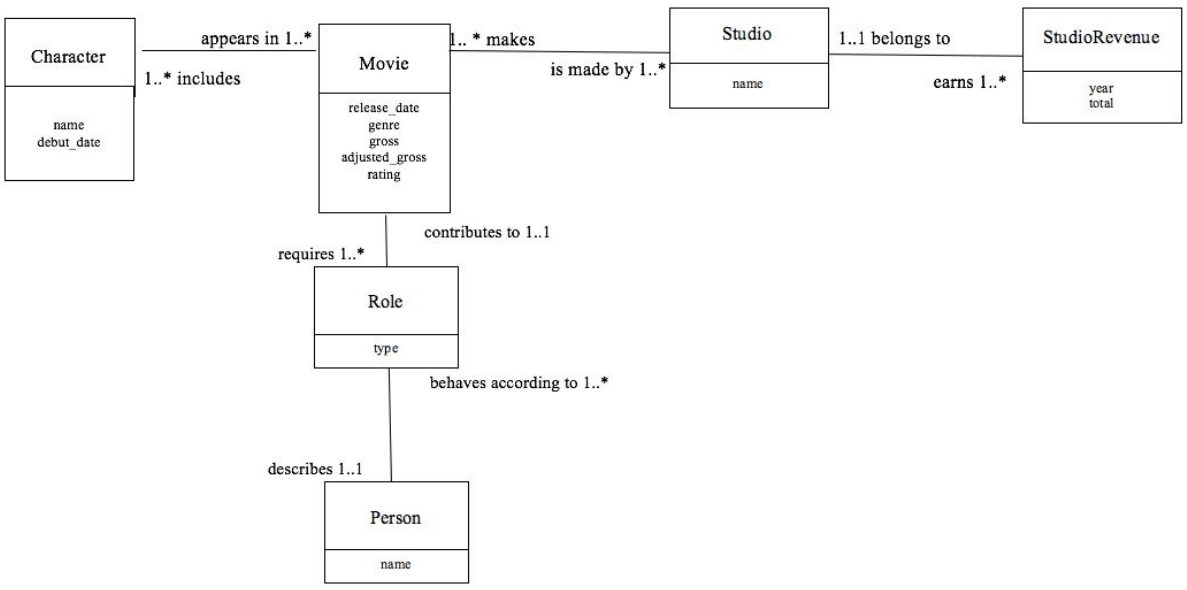
Deleted the “new row”:

```
row["movie_title"] = re.sub('[\n]','',row["movie_title"])
```



Data Model

ER Diagram



Relational Vocab

Movie **has_and_belongs_to_many** Character
Character **has_and_belongs_to_many** Movie

Movie **has_and_belongs_to_many** Studio
Studio **has_and_belongs_to_many** Movie

Studio **has_many** StudioRevenue
StudioRevenue **belongs_to** Studio

Movie **has_many** Person **through** Role
Person **has_many** Movie **through** Role
Role **belongs_to** Movie
Role **belongs_to** Person

Sample Tables

characters			
id	name	debut_date	type
10	Moana	2016-03-04	hero
20	Judy Hopps	2016-11-23	hero

movies						
id	name	release_date	genre	gross	adjusted_gross	rating
1	Moana	23/11/2016	Adventure	246082029.00	246082029.00	PG
2	Zootopia	04/03/2016	Adventure	341268248.00	341268248.00	PG

characters_movies	
character_id	movie_id
10	1

20	2

studios	
id	name
34	Disney

movies_studios	
movie_id	studio_id
1	34
2	34

studio_revenues			
id	year	total	studio_id
82	2015	52465	34
83	2016	55632	34

people	
id	name
300	Byron Howard
301	Ron Clements

roles			
id	type	movie_id	person_id
76	director	1	300
77	director	2	301

Analysis

For our analysis we will answer the following research questions:

- Is there a significant relationship between gross and movie ratings? How has this changed through the decades? In recent years, PG movies have become the top grossers? Is this solely because there are more PG than G movies released by Disney now than in the past?

After creating our database, we used SQL to query the database and generate CSVs. After each query we exported the CSV to import into R to do linear regression, correlation and confidence intervals, and basic summary stats. We begin with our SQL queries, followed by code in R, and then a summary of our analysis with graphs created in data.world.

SQL

Here are the SQL queries we executed in phpMyAdmin, along with links to the corresponding output files and basic visualizations:

1.How many percentage does box office account for revenue each year?

```
SELECT studios_revenues.year, SUM(movies.gross) AS TOTAL_BOXOFFICE,
studios_revenues.total * 1000000 AS ANNUAL_REVENUE,
SUM(movies.gross)/(studios_revenues.total * 1000000) AS Percentage
FROM movies
JOIN movies_studios
  ON movies.id = movies_studios.movie_id
JOIN studios
  ON movies_studios.studio_id = studios.id
JOIN studios_revenues
  ON YEAR(movies.release_date) = studios_revenues.year
GROUP BY YEAR(movies.release_date)
ORDER BY Percentage DESC
```

https://docs.google.com/a/utexas.edu/spreadsheets/d/1BaaAegHcYtCqY_RuwMWAxgx4JWMMbE-O0bZb5iyygVU/edit?usp=sharing

2. Which hero characters generate the highest box office revenues?

```
SELECT characters.name, movies.gross, YEAR(characters.debut_date) AS debut_year
FROM characters
JOIN characters_movies
  ON characters.id = characters_movies.character_id
```

```
JOIN movies
  ON characters_movies.movie_id = movies.id
WHERE characters.type = "hero"
ORDER BY movies.gross DESC
```

https://docs.google.com/a/utexas.edu/spreadsheets/d/1bSgS0e17S3YR4gzT3ljHmTanDQHWVYDPypRcX3NRxSo/edit?usp=sharing

3. Who is the most reoccurring Disney character?

```
SELECT characters.name, COUNT(*) AS characters_popularity
FROM characters
JOIN characters_movies
  ON characters.id = characters_movies.character_id
JOIN movies
  ON characters_movies.movie_id = movies.id
GROUP BY characters_movies.character_id
ORDER BY characters_popularity DESC
```

https://docs.google.com/a/utexas.edu/spreadsheets/d/1GkCYR8rKKhkpsCUtl-hsRc-3Q7jUvT40cuiTjl1q_3w/edit?usp=sharing

4. Who is the most reoccurring movie person?

```
SELECT people.name, COUNT(*) AS people_work_most
FROM people
JOIN roles
  ON people.id = roles.person_id
JOIN movies
  ON roles.movie_id = movies.id
GROUP BY roles.person_id
HAVING people.name NOT LIKE 'None'
ORDER BY people_work_most DESC
```

https://docs.google.com/a/utexas.edu/spreadsheets/d/1sCpCN5U91px10yOXvxyr_3wYFG80p1gvVk-G0-cHQM8/edit?usp=sharing

R

How We Learned About R: We chose to utilize R for our analysis tool because we both have experience using statistical packages. This semester Kelly learned the basics of R in the LBJ course “Statistical Analysis and Learning.” Lichen also has classroom experience with

proprietary software SAS. For reference, we used open source textbook “Introduction to Statistical Analysis” by G. James, D. Witten, T. Hastie, and R. Tibshirani.

Using the GUI RStudio, we executed these basic commands in R:

- summary()
- contrasts()
- confint()
- plot()

Using CSV from this SQL query

```
SELECT *
FROM movies
```

we were able to compute basic summary statistics, correlation, and confidence intervals.

Here we are interested in running linear regression on “adjusted_gross” and the categorical variable “rating” to see whether MPAA rating have a significant effect on the gross of movies. In order to run this analysis we need to remove the null values. We tried again and again to get this working in R, but we were unsuccessful, so we ended up having to filter the rating column in Excel just to get the non-null rows. See our attempts in R below:

```
Movies=read.csv("movies.csv",header=T,na.strings = "?")
fix(Movies)
dim(Movies)
Movies=na.omit(Movies)
dim(Movies)
names(Movies)
plot(Movies$rating, Movies$adjusted_gross)
library(tidyr)
Movies %>% drop_na(rating)
dim(Movies)
lm.fit=lm(gross~rating, data = Movies)
Movies$rating <- as.numeric(as.character(Movies$rating))
Movies[ -grep("?", Movies$rating, invert = TRUE) , ]
```

The working code is quite easy after we got the nulls removed, however. R automatically creates five dummy variables for rating which correspond to each level: not rated, G, PG, PG-13, and R.

Here is the code:

```
Movies=read.csv("movies2.csv",header=T)
```

```
View(Movies)
dim(Movies)
fix(Movies)
lm.fit=lm(adjusted_gross~rating,data=Movies)
lm.fit
summary(lm.fit)
```

Here is the output:

```
Call:
lm(formula = adjusted_gross ~ rating, data = Movies)

Coefficients:
    (Intercept)  ratingNot Rated         ratingPG      ratingPG-13
      291260995          8612418       -189719561       -188312411
        ratingR
     -235955189


 Call:
 lm(formula = adjusted_gross ~ rating, data = Movies)

 Residuals:
        Min          1Q      Median          3Q         Max
 -299447167  -77872124  -40731719   15814059 4937692256

 Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
 (Intercept)      291260995   31264487   9.316  < 2e-16 ***
 ratingNot Rated    8612418  170288642   0.051     0.96
 ratingPG        -189719561   37775640  -5.022 7.04e-07 ***
 ratingPG-13     -188312411   39461473  -4.772 2.38e-06 ***
 ratingR         -235955190   42445352  -5.559 4.35e-08 ***
 ---
 Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

 Residual standard error: 289900000 on 518 degrees of freedom
 Multiple R-squared:  0.06722,   Adjusted R-squared:  0.06001
 F-statistic: 9.332 on 4 and 518 DF,  p-value: 2.743e-07
```

All the variables are extremely statistically significant except Not Rated movies. We believe this can be attributed to the relatively small number of movies coded as Not Rated.

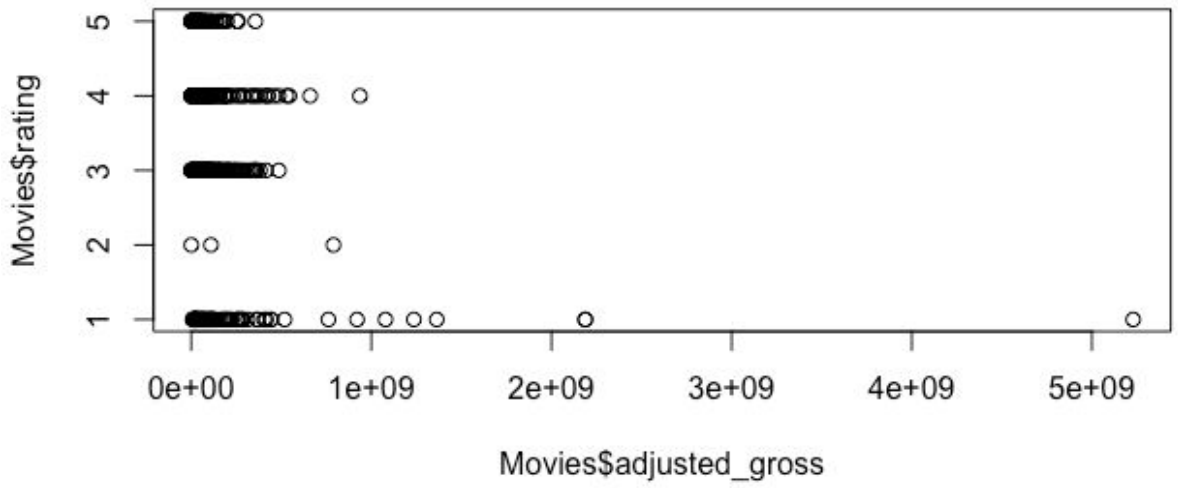
```
## this function returns coding that R used for the dummy variables
contrasts(Movies$rating)
```

```
          Not Rated PG PG-13 R
G                 0  0     0 0
Not Rated         1  0     0 0
PG                0  1     0 0
PG-13             0  0     1 0
R                 0  0     0 1
```

confidence intervals
confint(lm.fit)

```
                    2.5 %     97.5 %
(Intercept)      229840216  352681774
ratingNot Rated -325928849  343153684
ratingPG        -263931853 -115507270
ratingPG-13     -265836614 -110788208
ratingR         -319341385 -152568994
```

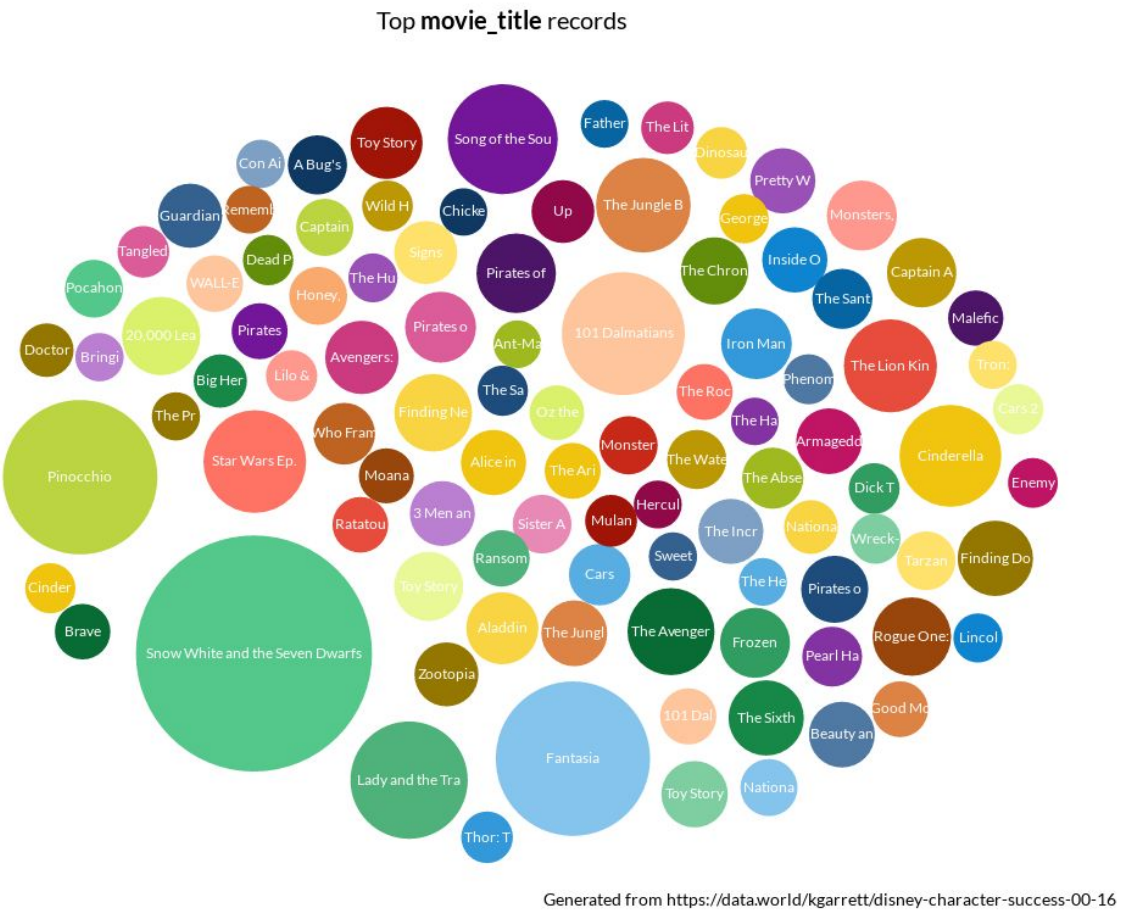
plot(Movies$adjusted_gross,Movies$rating)



In level 1, G-rated movies, “Snow White and the Seven Dwarfs” appears to be an outlier. As you can see there are very few observations in level 2, Not Rated, which may account for why it was not statistically significant.

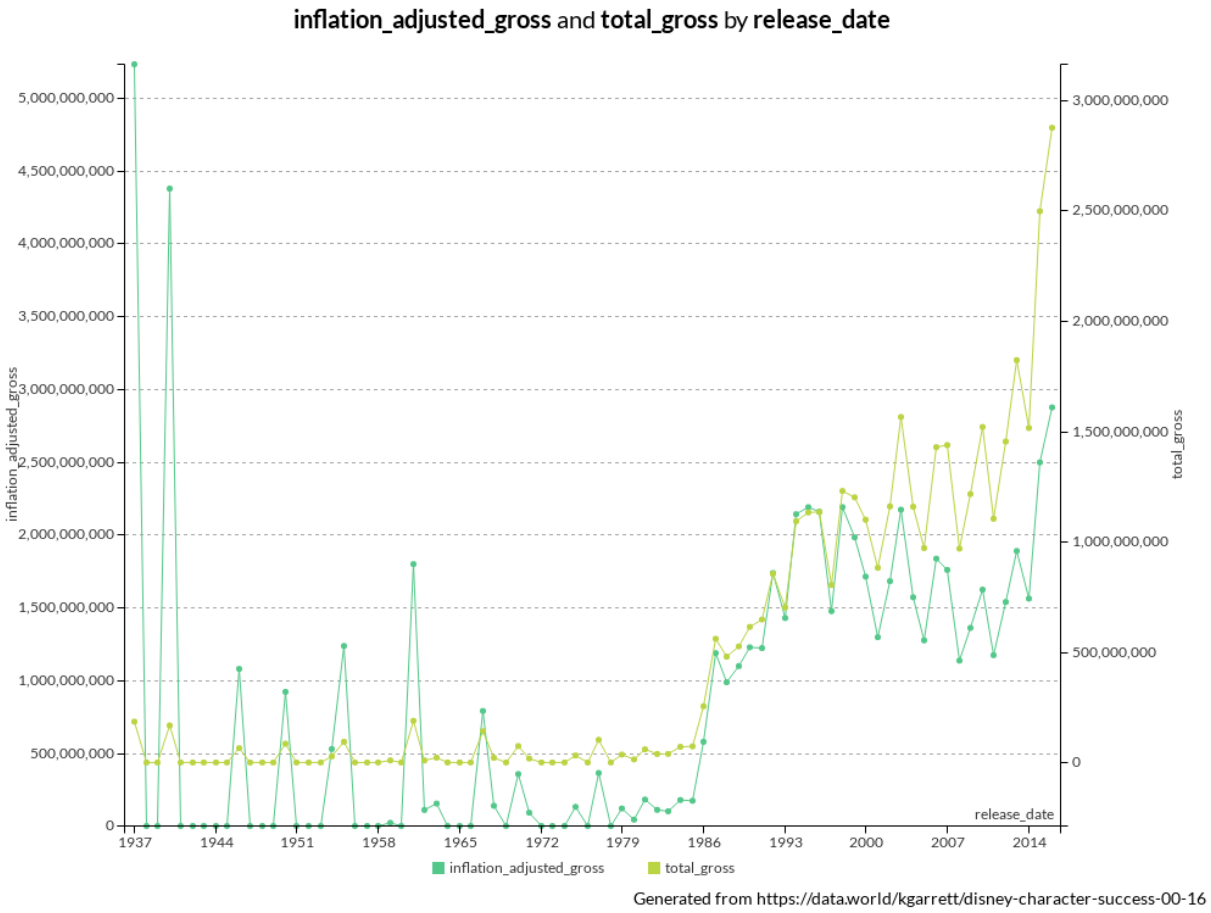
Putting It All Together: Visualizations & Summary

Movies by Adjusted Gross



Frozen was a box office hit in 2014, and we assumed that it would be one of the highest grossers. As you can see from the visualizations, while Frozen performed well compared to movies of that time, when you adjust for inflation, many of the older movies have done much better, such as Snow White and the Seven Dwarfs and Pinocchio.

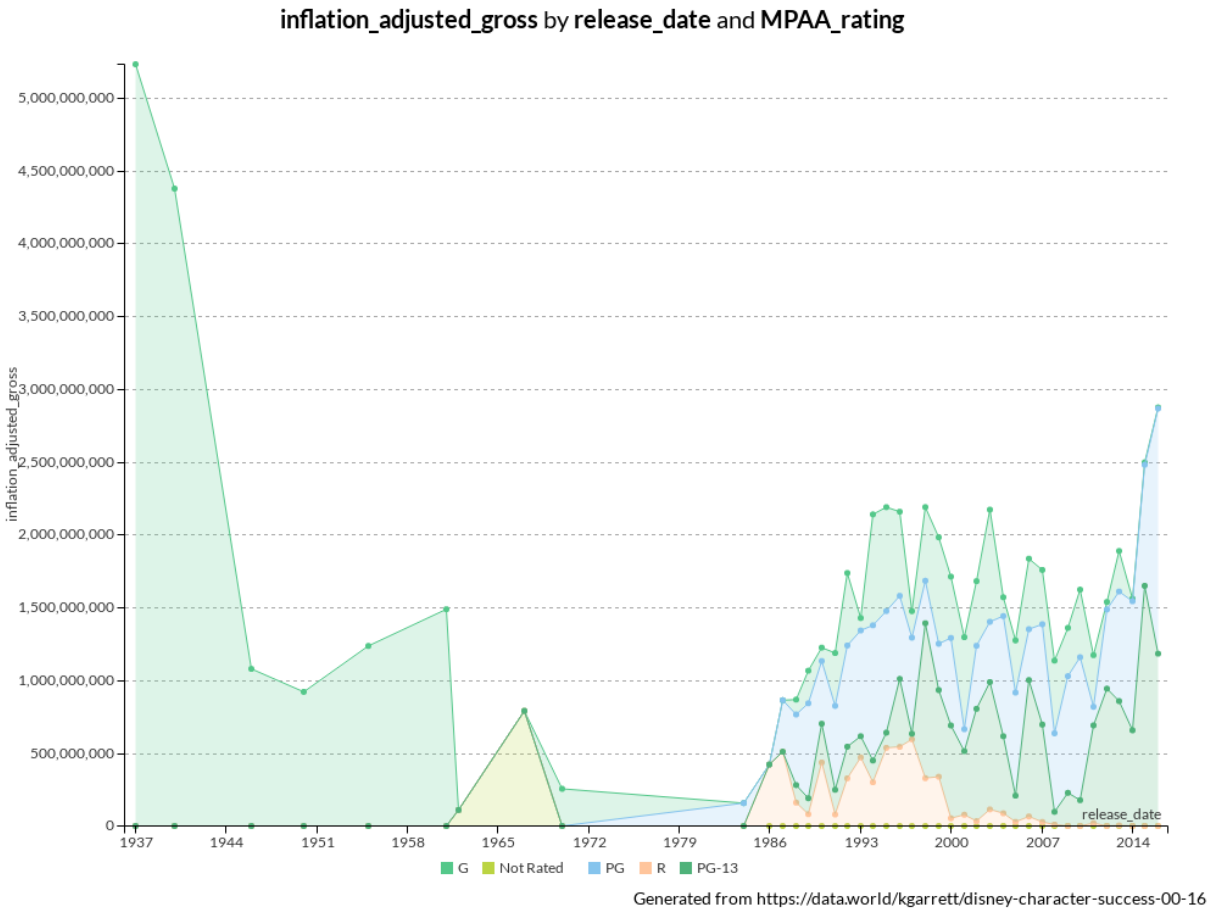
Movie Gross and Adjusted Gross by Year



Another interesting graph shows the total gross and inflation adjusted gross across years. There is still a very sharp increase in gross around the year 2014, but Disney is not making nearly as many movies as they were in the early 1990s. 2014 seems to be the year of the comic book/superhero movie craze, so there might be something there….

Disney's purchase of the Star Wars Franchise which brought in an additional ~$940 million in 2015 and another ~$530 million in 2016. This would have been unrealized revenue without the acquisition.

Movie Gross by MPAA Rating and Release Year



Just as the R code does, this graph excludes movies that include null or blank values for rating. We can see that all Disney movies have been coded as G prior to the mid-1950s. By the late 1980s Disney begins to earn revenue from all the category of ratings. Present day Disney does not appear to be releasing many R movies since revenue for this category has been zero or nearly zero since 2006.

Challenges

Throughout this project we have had several challenges that we were able to resolve. We are most proud of the following:

1. We've designed a database that has seven entities and modeled them using proper relational vocabulary.
2. The movie title data come from 4 different CSVs, and the characters' names come from 2 CSVs. There were repeated and new values in each CSV. To solve this, we created a list for movie titles and a list for characters, then we used "if" condition to check whether the title/name was repeated when reading a different CSV.

3. There were several issues with data formats when reading CSVs. For instance, the date format was different from database's, the movie title had a "/n" in front of each value, gross had a $ in each value. By using the syntax we've learned, we have solved the issues successfully.
4. Retrieving foreign keys from two tables and then insert these keys to one has_and_belongs_to_many table.