Group 23: Immanuel Ponminissery, Khoi Tran, Hyeon Gu Kim & Shruti Kapur

Movie Recommender for Children

Introduction

As video streaming platforms have accelerated the trend of cord-cutting via today's fast-evolving technology, recommendation systems have become very significant in video streaming services such as Netflix. However, there is a high risk that failure of said systems could lead to children being easily exposed to explicit video content. Acknowledging the importance of differentiating explicit content and family-friendly content in the recommendation system, our group decided *to analyze video contents from Netflix and classify said content as either child safe or not*.

Data Collection and Cleaning

Rather than analyzing just one data set, we decided to take Netflix dataset as our primary dataset and filled some details about movies through 4 IMDb datasets from Kaggle. Since there were 61 columns for both of the datasets, we dropped 25 columns that we considered unnecessary in recommending movies to the users (such as biographical information of actors, duration of movies, and specific jobs done by cast members, etc…). Figure 1 shows the columns of our main data frame.

Exploratory Data Analysis

For this project, we limit our dataset to movies on Netflix and define the category of 'children' as those aged 10 and below. In a bid to explore our data better, the team set out to answer the following questions:

1. What percentage of all movies on Netflix are not children friendly?
2. Which genres have the most non children friendly movies?
3. Can we predict if a movie falls in a non children friendly genre?
4. Is the movie child friendly?
5. Which movies are similar to this movie?

What percentage of all movies on Netflix are not children friendly?

To find the percentage of Netflix movies which were not children friendly, we first decided to label the content ratings as 'adult' or not. Please refer to figure 2. We see that around 84% of movies on Netflix are not children friendly. That's a huge number, thus giving some validation to our course topic!

Which genres have the most non children friendly movies?

To find the genre with the most non children friendly movies, we then created a data subset using a mask. From this data subset, we created a list and took value_counts() across the list of genres to find the highest count of genres with non-children friendly movies. Note that each movie here could be classified across multiple genres. Thus, the total count of genres might not equal the total count of movies. Please refer to figure 3.

Thus, we see that 'Drama' tops the list of genres among the non children friendly movies. This makes sense as children below the age of 10 are usually not mature enough to handle the subject matter of serious drama movies and drama genre forms a large bulk of all movies released. Note that this value would have been normalized by total counts across genres if the question had been which genre has the highest probability of a movie being non children friendly. However, since we are looking for the genre with most non children friendly movies, value_counts() works well.

Can we predict if a movie falls in a non children friendly genre?
In order to predict whether a movie is a children friendly genre or not, our team decided to utilize logistic regression. We set 'Drama' as our threshold meter and made it a target where contents that contain 'Drama' under their genre columns are classified as positive classes (or 1) and otherwise negative classes (or 0). To make our model run properly, we dropped some predictors that have correlation between each other or contain too many categorical values. As a result, our group narrowed the predictors down to 'weighted_average_vote', 'total_votes', 'top1000_voters_rating', and 'type'. Figure 4 shows distribution of 'Drama' across content ratings.

The accuracy score for our training data sets was 0.6415 and the accuracy score for our test data sets was 0.6557. Figure 5 shows the weights of each feature. While the 'weighted_average_vote' is the heaviest (0.861988), 'total_votes', 'top1000_voters_rating' and dummy variable 'type[TV Show]' are low negative weights, indicating that the increase in these features leads to higher likelihood that a movie is classified as 'safe'. Figure 6 is the confusion matrix of our logistic regression. The confusion matrix gave us an overall view of how well our regression model is working. The sensitivity is 0.7355 and the specificity is 0.5508. Utilizing our logistic regression model with 'A Cinderella Story' as an input, the model predicted it as 'Not Drama' which is most likely 'safe' to children.

Is the movie child friendly?
At first, the team decided that the use of a combination of categorical and numerical data points to understand which movies were child friendly would yield accurate results. However, a kNN model does not work with categorical variables and modifications had to be made to ensure that the kNN model was dealing with binary variables. As a result, a new column, 'safe', was created in the data frame. The 'safe' column had a value of 1 if the movies could be viewed by children who were around the age of 10. Based on this condition, a mask was created with the help of the content ratings of the movies and values were assigned to 'safe'.

The team decided that it would be worth investigating the predictive abilities of different demographic variables on the ratings of different movies. Since the team had data regarding the ratings provided by people from a wide range of age groups, a model that can predict if a movie was safe for children based on said ratings could theoretically be created. A model with 3 fold cross validation was created and it was noticed that the value for accuracy remains constant for a higher number of neighbors. This could be due to the fact that the model is simply predicting the most common value of the 'safe' variable. Since it was brought to attention during class that higher dimensions usually cause kNN models to fail, a limited number of variables had to be chosen. It was conjectured that more mature content could possibly have been rated more favorably by younger adults, between the ages of 18 and 30, as opposed to older individuals above the age of 45 who might have children and whose tastes may have changed after becoming parents. For analysis purposes, ratings for males were closely inspected and a plot that shows the accuracy of a 3 fold cross validation model against the number of nearest neighbors is shown in figure 7. It is worth noting that 'distance' was assigned to the 'weights' parameter to capture the effect of the small groups of 'safe' movies. As figure 7 shows, there is some demarcation, however, the plot in figure 8 shows that even if there is a slight increase in accuracy for 15 nearest neighbors, it is not substantially different from a case when the model is making the prediction based on the most common value in 'safe'. Using the kNN model and inputs based on the movie 'A Cinderella Story', the kNN predicted that it was a 'safe' movie.

Which movies are similar to this movie?

Our group further decided to take an attempt at building a recommender system for movies that were safe for children. Recommender systems can be either content based, or collaborative, or hybrid. Content based systems build on discrete characteristics of an item to recommend similar items, whereas collaborative filtering utilizes the user's past behaviour. Since we did not have user data such as timestamps of content watched, we decided to go ahead with 'Content-based recommendation'.

Going by the same logic that we used for kNN, we had to keep our predictor variables in the category of continuous variables, because k-means utilizes least-squares to define distances. Such distances can be defined inherently in numeric data, but for categorical data, the concept of 'distance' can not be defined directly. In this case, we have used the *same predictors* as in kNN.

Determining the optimal number of clusters is one of the key points of the K-Means algorithm. This can be done using the 'Knee' method (or the 'Elbow' method). This method involves plotting the intra-cluster variation as a function of the number of clusters, and picking the 'elbow' of the curve as

the optimal number of clusters. We ran if on our dataset of movies, plotting n between 5 to 500, with a leap of 5, for ease of computation. The graph with two 'elbows' marked at k = 20 and k = 200, can be found in figure 9. The lack of a clear 'elbow' in the graph could point to the fact that clustering might not be the best algorithm here. However, for sake of academic exploration, we decided to go ahead with the results and compared cluster sizes across k = 20 and k = 200. Please refer to figure 10 and figure 11.

Since we find that both values of K yield unbalanced clusters, we decided to go with k=20 for sake of simplicity. With the optimal K as 20, we decided to build our recommender system as the group of common movies for our specified movie, 'A Cinderella Story'. Thus we got the following list of movies similar to our movie, and thus safe to watch for children. Please refer to figure 12 for the final result. The content and rating of the movies do validate the similarity in the case of our example.

Future Scope

1. Some possible shortcomings for our project included that the inputs for both kNN and K-means utilized only the continuous variables, and thus limited the scope of predictive power for our models. For instance, for the final recommendation, we do see movies across languages cropping up. This could be due to the fact that 'language' was not included in the predictors, due to it being categorical. Such issues can always be addressed by 'dummy' variables, but doing so here across all our categories was beyond the scope of our project.
2. The logistic regression has potential limitations. It is possible to have omitted variable bias as we dropped most of our features for a simple yet effective regression model. Such shortcoming leads to another limitation: the lack of relevant data for determining whether a content is 'Drama' or not. Since the original data sets were widely used for movie recommendation systems, the data sets were not perfectly suitable for our logistic regression model.
3. Our team also decided to explore the TF-IDF and Cosine functions. For the purpose of our project, we decided to use the inbuilt functions on scikit learn and use it to examine descriptions of movies. An example of the output to the input movie name 'A Cinderella Story' is shown in figure 13 and table 1 highlights similar words used to make the prediction in figure 13.
4. We also considered the possibility of adding data on streaming platforms for each movie title, so that this could become a recommender system for where to find certain content. However, due to lack of data, this went beyond the scope of our current project but continues to serve as a potential future project idea.

Plots and Figures

```
movies_df.columns
```

```
Index(['imdb_title_id', 'imdb_name_id', 'characters', 'title', 'year', 'genre',
       'country_x', 'language', 'director_x', 'name', 'weighted_average_vote',
       'total_votes', 'mean_vote', 'votes_10', 'votes_9', 'votes_8', 'votes_7',
       'votes_6', 'votes_5', 'votes_4', 'votes_3', 'votes_2', 'votes_1',
       'males_allages_avg_vote', 'males_0age_avg_vote', 'males_18age_avg_vote',
       'males_30age_avg_vote', 'males_45age_avg_vote',
       'females_allages_avg_vote', 'females_0age_avg_vote',
       'females_18age_avg_vote', 'females_30age_avg_vote',
       'females_45age_avg_vote', 'top1000_voters_rating', 'show_id', 'type',
       'director_y', 'cast', 'country_y', 'date_added', 'release_year',
       'rating', 'listed_in', 'description'],
      dtype='object')
```

Figure 1

```
# Create list of non - children friendly movies
adult_ratings = ['TV-MA', 'TV-14', 'R', 'PG-13', 'NR', 'UR']
mask = (ex11['rating'].isin(adult_ratings))

ex11 = ex11[mask]
ex11['genre']
```

```
1                       [Drama]
2                       [Drama]
3      [Drama, Musical, Romance]
4                       [Drama]
5                      [Action]
                ...
906              [Drama, Family]
907                     [Drama]
908                     [Drama]
909                     [Drama]
910           [Horror, Thriller]
Name: genre, Length: 771, dtype: object
```

```
percentage_unsafe_movies = len(ex11)/len(ex1)
percentage_unsafe_movies
```

```
0.8463227222832053
```

Figure 2

```
# Count total number of genres among movies unsafe for children
genres = []
for i in ex11['genre']:
    genres.extend(i)
unique_genres = list(set(genres))
temp = pd.Series(genres)
temp.value_counts()
```

```
Drama          447
Comedy         253
Action         170
Thriller       166
Romance        145
Crime          133
Horror          95
Mystery         68
Adventure       48
Sci-Fi          32
Biography       32
Fantasy         23
Family          20
History         18
Music           13
War             13
Sport           11
Musical          8
Western          7
Animation        4
dtype: int64
```

Figure 3

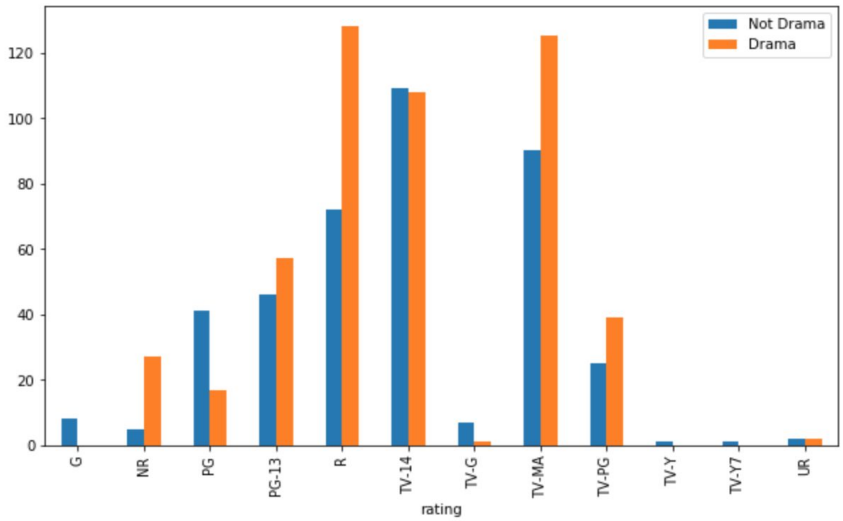

Figure 4

```
total_votes                -0.316091
top1000_voters_rating      -0.036548
type[TV Show]              -0.006226
type[Movie]                 0.006226
weighted_average_vote       0.861988
dtype: float64
```

Figure 5

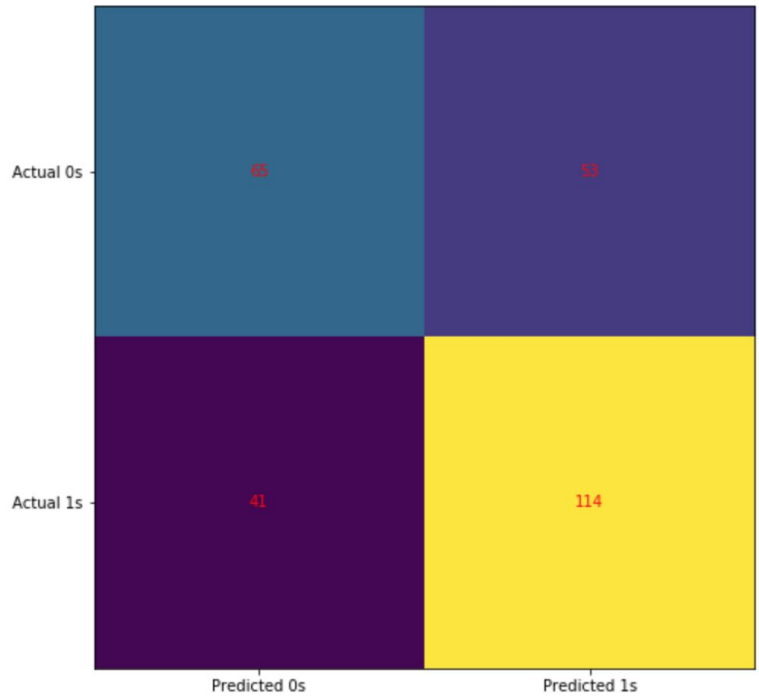

Figure 6

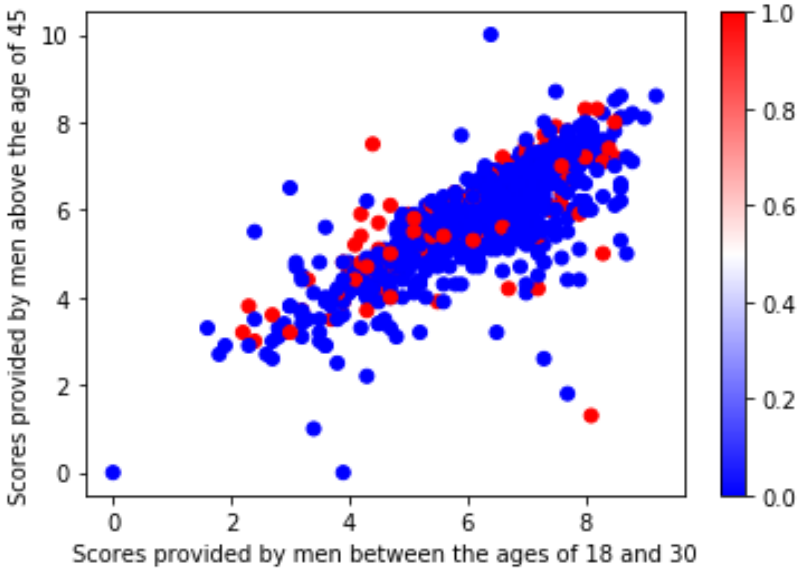

Figure 7 : Scatterplot of scores based on age categories for males

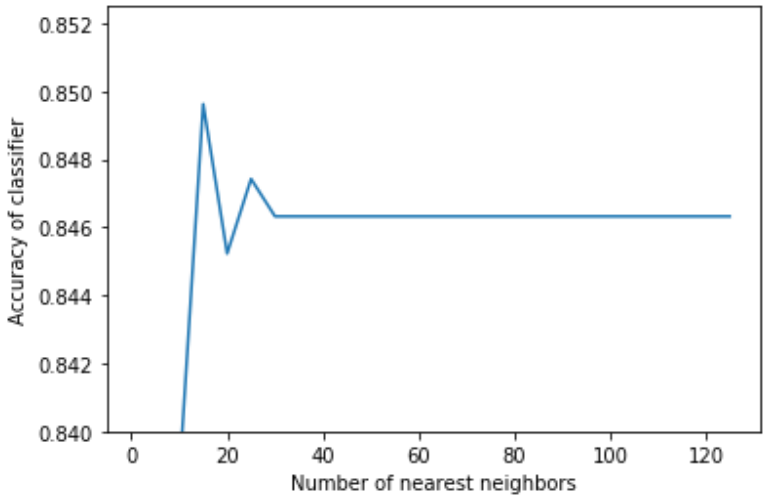

Figure 8 : Accuracy as a function of the number of nearest neighbors

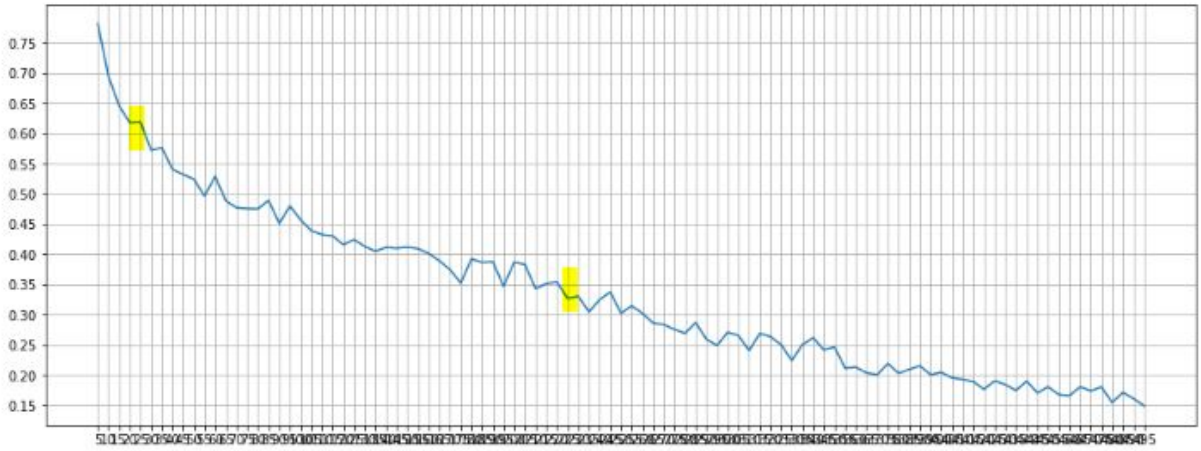

Figure 9 : Sum of squared distances v/s values of k

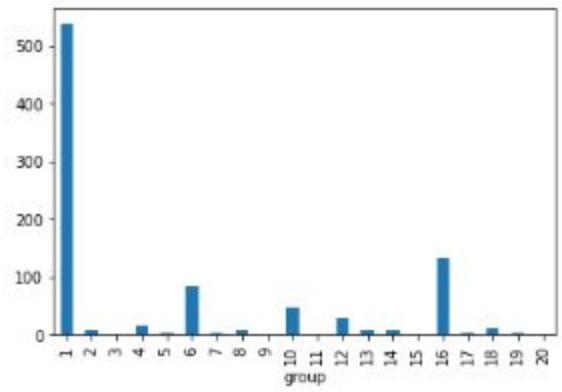

Figure 10 : k = 20

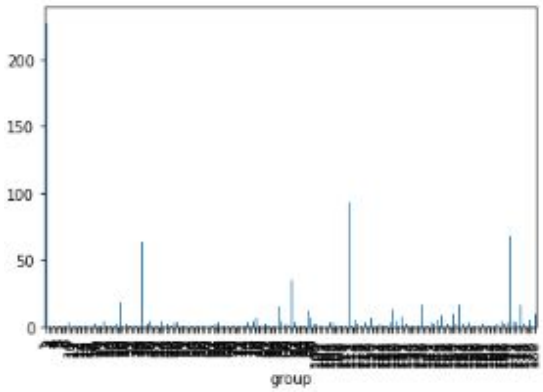

Figure 11 : k = 200

11	20	Daddy Day Care	5.2
40	20	The Tuxedo	5.1
38	20	The Pink Panther	5.0
13	20	Dil Chahta Hai	5.0
26	20	Nacho Libre	5.0
14	20	Eat Pray Love	4.9
28	20	Silent Hill: Revelation	4.7
0	20	A Cinderella Story	4.7
35	20	The Dukes of Hazzard	4.4

Figure 12 : List of movies similar to ' A Cinderella Story'

Udaan
Happy New Year
Center Stage
Manglehorn
Tanu Weds Manu

Figure 13 : List of movies similar to ' A Cinderella Story' according to tf-idf

Table 1 : tf-idf analysis with cosine similarity of ' A Cinderella Story'

Movie	Description
A Cinderella Story	Teen Sam meets the boy of her dreams at a dance before returning to toil in her stepmother's diner. Can her lost cell phone bring them together?
Udaan	Upon returning to his industrial hometown, a young man must decide whether to follow his own dreams or acquiesce to his father's plans for his future.
Happy New Year	A revenge-seeking diamond thief gathers a ragtag crew to infiltrate a Dubai hotel hosting a dance contest. But first they have to learn how to dance.
Center Stage	Vying for a spot in the American Ballet Company, 12 dance students are ready to push their bodies and minds to the limit to realize their dreams.
Manglehorn	A reclusive small-town locksmith who can't stop writing letters to a lost love meets a kindly bank teller who challenges him to look to the future.
Tanu weds Manu	When London-based doctor Manu reluctantly returns to India to find a bride, he meets the girl of his dreams only to discover she loves another man.

Sources

https://scikit-learn.org/stable/modules/generated/sklearn.metrics.accuracy_score.html

https://github.com/vishank94/Movie-Recommendation-System/blob/master/Scripts/recommendationSystem.ipynb

https://www.analyticsvidhya.com/blog/2019/04/predicting-movie-genres-nlp-multi-label-classification/

https://towardsdatascience.com/unsupervised-classification-project-building-a-movie-recommender-with-clustering-analysis-and-4bab0738efe6

https://chrisalbon.com/machine_learning/trees_and_forests/random_forest_classifier_example/