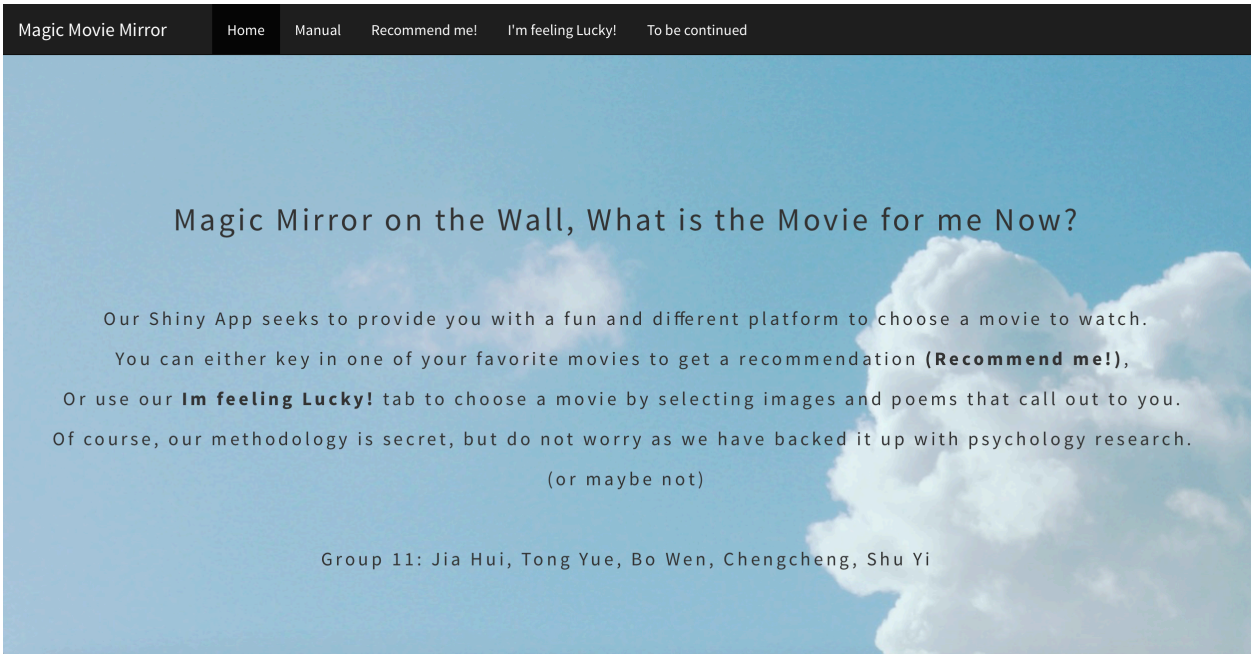
ADS project 5: Movie Magic Mirror



1. Project summary

In this project, we seek to build a shiny app that would recommend movies to users based on two different methods:

a. More direct method: Use a KDTree classifier that we built using the data and additional features, we will recommend 4 movies based on one particular favorite movie of the user.
b. More indirect method: Using “psychological analysis”, we will recommend a few movies based on what picture, what color, and what poetic line the user chooses.

We applied data science and natural language processing tools such as tree classification, LDA, sentimental analysis and word cloud visualization to implement the recommendation system and visualize the results in a more efficient, convenient, and most importantly fun manner.

2. Data source

The main dataset we used is the IMDB 5000 Movie Dataset from Kaggle which contains data on top 5000 movies from 1927 to 2016. Variables in the dataset include movie title, director name, genres, gross, plot keywords, actor names, Facebook likes, imdb score and so on. After cleaning our dataset and removing data points with NA values, we are left with 3609 observations.

On top of this, we wanted to add on new interesting dimensions to our analysis, on top of just doing recommendations based on "traditional" variables like score and gross movie earnings. Hence, we decided to scrap movie reviews for each our data points from IMDB using Python with the "BeautifulSoup" package (please refer to codes for this in lib). We find this a very interesting source to work with as it could be more reflective of the true sentiments of the audience to the movie (of course there would be nonsensical ones), and there is a larger potential of applying various natural language processing tools to this dataset, which we would explain further in the report.

3. Our Shiny App "Magic Movie Mirror" and How it Works

There are 2 main features of our shiny app:
(1) The panel "Recommend me!" requires the user to key in his/her favorite movie and we would recommend four movies using a KD Classifier tree.
(2) The panel "I'm feeling lucky!" requires the user to choose a picture, a color, and a poetic line that calls out to him respectively and we will use these to filter 4 movies for him.
(3) Word Cloud generation on movie reviews

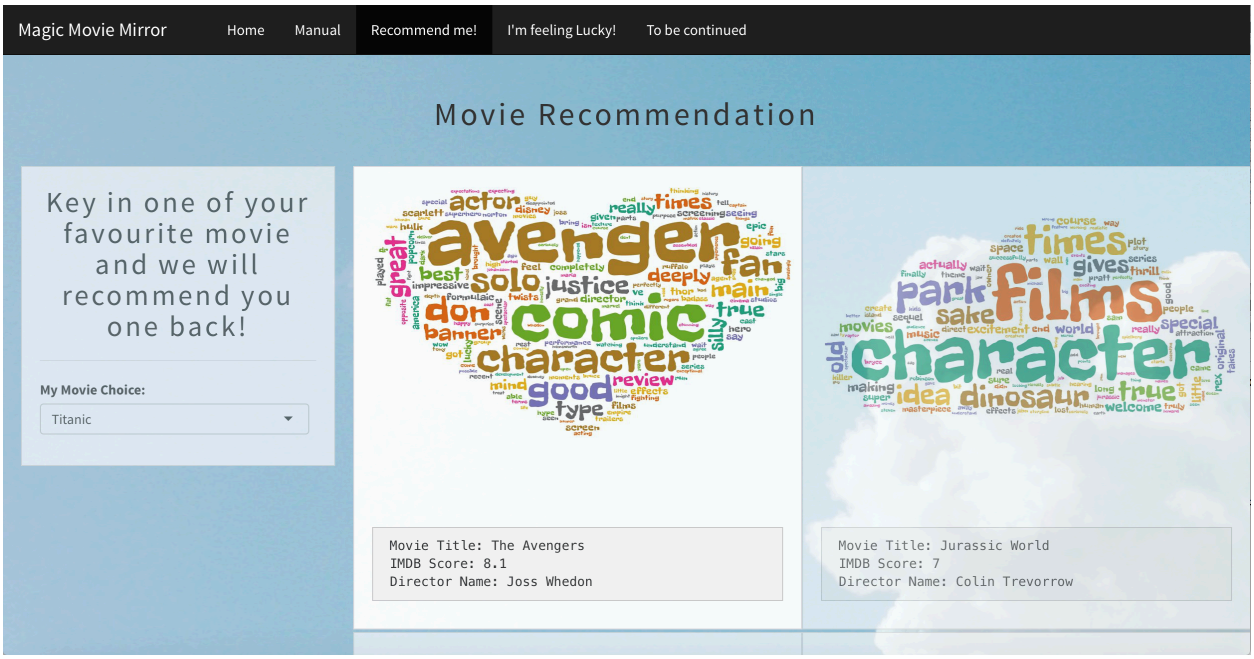
Part 1: "Recommend me!" panel

(1) Objective:
The main objective of this panel is to recommend four movies according to the favorite movie that the user inputs.

(2) Methodology:
We use the KD-Tree classifier "nn2" in RANN, which uses a KD-Tree to find a p number of near neighbors for each point in an input/output dataset. The advantage of the KD-Tree is that it runs in O (M log M) time where M is the number of data points. For more information on how the KD classifier works, please refer to (https://en.wikipedia.org/wiki/K-d_tree).

On top of using the variables in the IMDB dataset such as gross, genre (we converted each genre's appearance for each movie into a binary variable), number of Facebook likes, imdb score etc, we also applied NLP Sentiment Analysis on the variable "plot keywords". Using "get_nrc_sentiment", we obtained scores for 6 different sentiments: anger, anticipation, disgust, fear, joy, sadness and added this as plus features on our dataset, which we believe would be very useful in providing information on how similar certain movies are to each other using on the sentiment feelings derived from the plot description.

(3) Interface and output:
The output of this part is displayed in the first tab of app.



After the input of a favorite movie, 4 movie recommendations will be presented to the user in the form of a word cloud and some basic information. Each word cloud is constructed from the respective movie reviews on Rotten Tomatoes that we scraped.

The reason why we chose to display a word cloud instead of a more intuitive picture (like the movie poster) is that we feel that this is an interesting feature/information to display to the user. We are often bombarded by the magnitude and multitude of movie reviews, and the inertia is often too high to be able to read through these reviews properly and extract clear information from them. Using a word cloud, this would provide the user with a quick method of extracting the overall idea that the movie review conveys and their key words.

Basic information such as Movie Title, IMDB Score, and Director Name will be provided.

Part 2: "I'm feeling lucky!" panel

(1) Objective:

In this part, we also seek to recommend movies to our users. However, we seek to do this with a different twist!
Instead of inputting a favorite movie, we would like you to answer three fun relaxed questions that seek to provide as little idea to the user as to what information we are trying to extract so that it is more casual and informal.

We will then, just like before, recommend four movies together with their reviews word clouds and their related information.

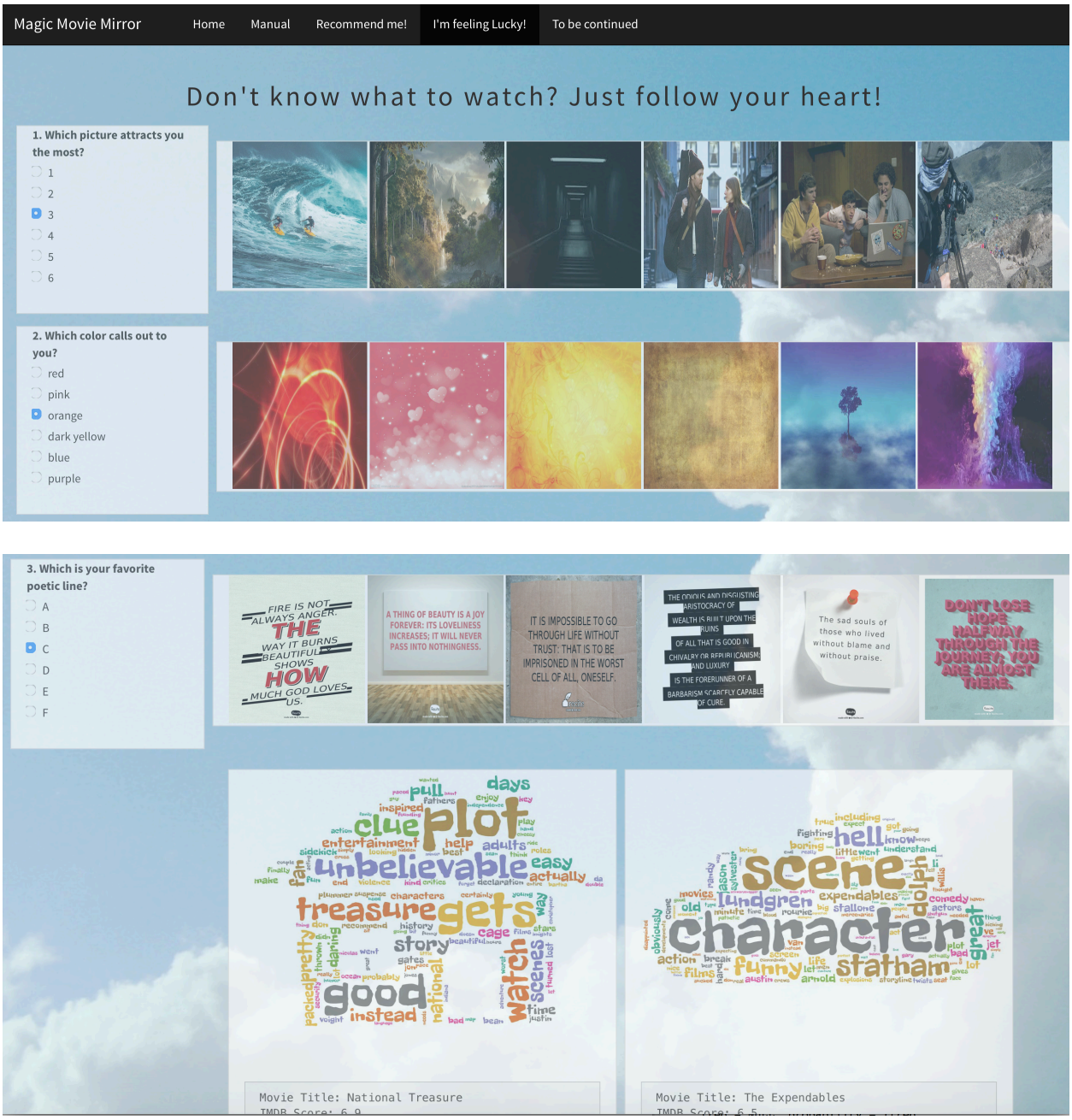
(2) Methodology:

The user is encouraged to choose a picture, color, and poetic line respectively that call out to him the most. We did some research on psychological explanation of links between these and sentiments/genre, and we applied this to our analysis.

For the first question, we used 6 pictures each corresponding to a genre (Action, Mystery, ….). However, we chose pictures that we hope are not as obvious as to which genre they are describing. We then filter the dataset according to the selected genre variable (i.e. corresponding picture) in the dataset.

For the second question, we used the sentiment analysis variable done on the plot keywords to filter the dataset. 6 colors (Red….) are chosen to represent one or several of the sentiments anger, anticipation, disgust, fear, joy, sadness, surprise and trust. . This was based on some research done that show certain colors reflect certain moods/mood preferences of the user (i.e. Pink refers to joy). Finally, for the last question, we extracted 6 lines from 6 poems that we hope would convey a sentiment each.

(3) Interface:
The output of this part is displayed in the second tab of app



Part 3: Generate Word Cloud

(1) Objective:

The objective of generating the word cloud is to give users a quick big picture about what are the main keywords of the review of the movie.

(2) Methodology:

Here we used Python to make the Word Cloud fancier by changing the font, shape, color of the Word Cloud. While generating the Word Cloud, we used TF-IDF and LDA to ensure the accuracy

of importance of each word, so that the output Word Cloud pictures will have a better representation.

(3) Interface:
The output of this part is displayed in the first and second tab of app.

Part 4: Exploratory data analysis

(1) Objective:
In this part, we hope to provide some additional information that we hope would be useful in some way in helping the user decide on what kind of movie to watch (i.e. which genre, and which country). In view of this, we created two plots.

(2) Methodology:
We use R package plotly to show some two visualization results.

First, we plot a 3D graph to show the relationship between country, gross and IMDB score. This allows the user to explore the distribution of IMDB score and gross for movies of various countries.

The second graph is a box plot of IMDB score across different genres. This is to give a preliminary visualization of which genre seem to have a higher IMDB mean /spread size.

(3) Interface:

The output of this part is displayed in the third tab of app.