Data Understanding

The dataset that will support data mining to address our business problem comes from the database of IMDB website. We obtained it in two ways. First, the majority of the data came from its official API. In addition, we used several web scrapping techniques to collect some complementary data from other websites counting statistical records of movies, such as 'The Numbers', and 'Box Office Mojo'. In total, we gathered more than 15500 movie records. With such large number of data, we hope to build a model to predict the box office for incoming movies in which we can make businesses based on this powerful model.

Data Cleaning

Introduction

This memo details the conventions used for the cleaning and checking of potentially suspicious or out-of-range values on variables of this dataset. The cleaning took place in 2 stages: a preliminary stage when raw data were obtained from the API and then a final cleaning stage. This memo outlines the procedures used at both stages of data cleaning.

Preliminary Cleaning

After we obtaining the data by requesting the IMDB API, a set of procedures was followed to complete a preliminary cleaning of the file.

First, we recode all numeric variables, such as `Runtime` , `Budgets` , and `BoxOffice` , from string to int or float. For the target variable `BoxOffice` specifically, we need to remove the dollar sign and commas before casting it to float. For variables `imdbRating` , `Internet_Movie_Database` , `Rotten_Tomatoes` , `Metacritic` involving ratings but using different scales, we need to transform them to a scale of 1 to 10.

In addition, we truncate several variables to make them simpler. For variable `Actor` , we select the first four actors who are the most famous ones to represent each movie. For movies with less than four actors, we use `nan` to fill the blank. Similarly, for variables `Director` and `Writer` , we only choose its top value for each movie.

Last, we apply the method of feature engineering by adding and removing variables. Based on our domain knowledge and common sense, we believe the number of premiere countries should have a great impact on the target variable. Therefore, we create a new variable `Country_count` by counting the number of premiere countries of each movie. Also, we replacing the variable `Genre` with 24 dummy variables representing different types of movie. Since we are only interested in movies, instances such as TV series, documentary, and other categories should be removed from the data set.

The final results are in Table 1.

Table 1

Variable	Type	Description
BoxOffice	float	Box office of the movie
NLP_Score	float	Self-assigned score of the movie
Director	string	Name of the director
Runtime	int	Runtime of the movie
Title	string	Name of the movie
Writer	string	Name of the write
Year	int	Released year
imdbVotes	int	Number of votes
imdbRating	float	Rating from IMDB
Internet_Movie_Database	float	Rating from Internet Movie Database
Rotten_Tomatoes	float	Rating from Rotten Tomatoes
Metacritic	float	Rating from Metacritic
Actor	int	Name of actors
Country	string	Names of premiere countries
Country_count	int	Number of premiere countries
Adjusted_Budgets	float	Movie budget
Genre	factor	Type of the movie (Action,Adventure,Animation, Biography,Comedy,Crime,Documentary,Drama,Family, Fantasy,FilmNoir,History,Horror,Music,Musical, Mystery,News,Romance,Sci-Fi,Short,Sport,Thriller, War,Western)

Final Cleaning and Verification in Detail

While the Preliminary Cleaning stage caught many cleaning-related issues, there were still issues that either were not caught by the cleaning programs or were somehow missed. Thus, we reexamined each variable once we got the dataset. The following changes were given particular attention: 1) Variables with missing values; 2) Transforming several variables; Each of these issues is addressed in more detail below.

1. Handling Missing Values

   For the whole dataset, there are only three variables with missing values, which are `BoxOffice` , `Budget` , and `Rating` . We applied two methods to fill up the missingness. For the target variable `BoxOffice` and variable `Budget` , we use another dataset obtained from the IMDB Pro. On the other hand, we impute the mean to the missing values of `Rating`,

2. Transforming variables

   It is clear that several variables are greatly influenced by year. For instance, the inflation of each year should have a great impact on variables such as `BoxOffice` and `Budget`. To make them more comparable, we used the inflation rate of each year to adjust these two variables for each movie. After adjusting, we divided them by 1,000,000 and then applied the log transformations to them to make the relationship more clear. Also, the variable `imdbVotes` should be adjusted based on year, since the number of Internet users has a strong positive relationship with year. To make it more accurate, we fit a simple linear regression model to adjust the vote for each movie.