Data Mining 1 Project

Giulia Fabiani   Camilla Maffeis

1 DATA UNDERSTANDING AND PREPARATION

The goal of the project is to analyze the IMDb Dataset, which contains data about movies, TV shows, and other forms of visual entertainment, along with their ratings. Each record includes key information about the title, as well as insights into critical aspects such as awards and reviews, as well as statistical ratings and other metadata. The dataset is updated as of September 1, 2024.

1.1 Data Semantics and First Global Exploration of the Dataset

The IMDb Dataset contains 16431 records and 23 attributes, of which 10 are categorical and 13 numeric. They are listed in Tables 1 and 2. Among the **numerical attributes**, most of them are discrete and ratio-scaled, as they represent counts, with *startYear* and *endYear* being interval values. Only *runtimeMinutes* can be classified as a continuous attribute, as it represents a time duration, even though it only assumes integer values in the dataset. Among the **categorical attributes**, *canHaveEpisodes*, *isRatable* and *isAdult* are **binary**, *rating* , *worstRating* and *bestRating* are **ordinal**, while the rest are all **nominal**.

Looking at the records' data types, we can observe some syntactic inaccuracies: *isAdult* should be converted from integer to boolean, whereas *rating* assumes ten categorical values of the form *'(0, 1]', '(1, 2]', ..., '(9, 10]'*. We can assume they represent the binning of a continuous rating, therefore we map them into an integer space [1,10] to ease later analysis. Just by looking at the head of the dataframe, we can also tell that *countryOfOrigin* and *genres* contain collections of categorical values; therefore, we transform them into lists to allow for easier handling. Some discrete numerical attributes (*endYear*, *awardWins*) are stored as float rather than boolean because `int64` does not support NaN values; however, we do not deem it necessary to convert them.

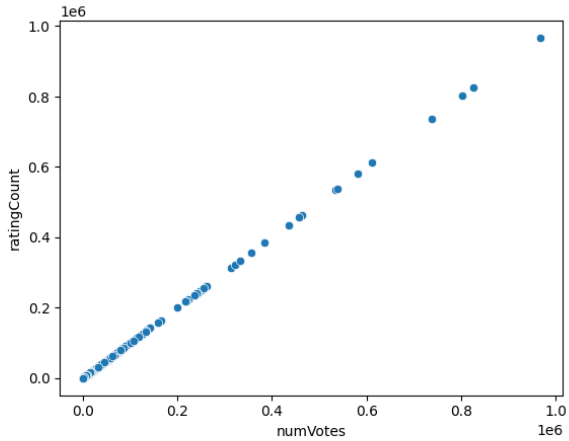

Figure 1: Scatterplot ratingCount~numVotes.

A first global analysis of the dataframe's summary and descriptive statistics already gives us valuable insight into which variables have missing values[1], as reported in Table 3, and into which features are uninformative: *bestRating*, *worstRating*, and *isRatable* can be safely dropped, as they only assume one value for all records (10, 1, and 1, respectively). We can also see that *numVotes* and *ratingCount* share very similar, when not identical, statistics. The scatterplot in Figure 1 proves their correlation close to 1, which leads us to drop *ratingCount* and only keep *numVotes*.

1.2 Distribution of categorical variables and statistics

In an effort to analyze the univariate distribution of categorical variables, we produced barplots for *titleType* (Figure 2), *countryOfOrigin* (Figure 3), and *genres* (Figure 4). All distributions are imbalanced, being dominated by few very common classes: *movie* and *tvepisode*, the *US*, *drama* and *comedy*, respectively. A barplot for *rating*[2], in Figure 5,

1 As the missing values were recorded inconsistently in the dataset, we used the `read_csv` function, passing a list of strings to recognize as missing values to the parameter `na_values`.

2 In this data exploration phase, we analyzed this ordinal variable both from a categorical and a numeric point of view, as to maximize insights.

Feature	Description	Type	Pandas dtype
runtimeMinutes	Primary runtime of the title, in minutes.	Continuous. Ratio-scaled.	float64
startYear	Represents the release year of a title. In the case of TV Series, it is the series' start year.	Discrete. Interval.	int64
endYear	TV Series end year.	Discrete. Interval.	float64
ratingCount	The total number of user ratings submitted for the title.	Discrete. Ratio-scaled.	int64
numVotes	Number of votes the title has received.	Discrete. Ratio-scaled.	int64
numRegions	The regions number for this version of the title.	Discrete. Ratio-scaled.	int64
totalImages	Total Number of Images for the title within the IMDb title page.	Discrete. Ratio-scaled.	int64
totalVideos	Total Number of Videos for the title within the IMDb title page.	Discrete. Ratio-scaled.	int64
totalCredits	Total Number of Credits for the title.	Discrete. Ratio-scaled.	int64
criticsReviewTotal	Total Number of Critic Reviews.	Discrete. Ratio-scaled.	int64
awardWins	Number of awards the title won.	Discrete. Ratio-scaled.	float64
awardNominationExclWins	Number of award nominations excluding wins.	Discrete. Ratio-scaled.	int64
userReviewsTotal	Total Number of Users Reviews.	Discrete. Ratio-scaled.	int64

Table 1: Numeric attributes.

Feature	Description	Type	Pandas dtype
originalTitle	Original title, in the original language.	Categorical; mostly unique classes (i.e., names).	object
titleType	The type/format of the title (e.g., movie, short, episode, video, etc.).	Categorical, 10 classes.	object
countryOfOrigin	The country where the title was primarily produced. Some titles can belong to more than one class.	Categorical, 153 classes. A record can belong to more than one class.	object
genres	The genre(s) associated with the title (e.g., drama, comedy, action). Some titles can belong to more than one class.	Categorical, 29 classes. A record can belong to more than one class.	object
canHaveEpisodes	Whether or not the title can have episodes.	Asymmetric binary.	bool
isRatable	Whether or not the title can be rated by users.	Asymmetric binary.	bool
isAdult	Whether or not the title is for adults.	Asymmetric binary.	int64
rating	IMDB title rating class.	Ordinal, 10 values.	object
worstRating	Worst title rating.	Ordinal, supposedly [1,10].	int64
bestRating	Best title rating.	Ordinal, supposedly [1,10].	int64

Table 2: Categorical, ordinal, and binary attributes.

Table 3: Features with missing values.

Feature	Non-Null Records
endYear	814
runtimeMinutes	11579
awardWins	13813
genres	16049

shows a left-skewed distribution, with the mode being the [7,8) category.

By looking at the categories for the different variables, we question the informativeness of *canHaveEpisodes* and *isAdult*. The first assumes positive values only for the title types *tvSeries* and *tvMiniSeries*: we keep it, as an explicit feature of serialized content. *isAdult* proves to be redun-

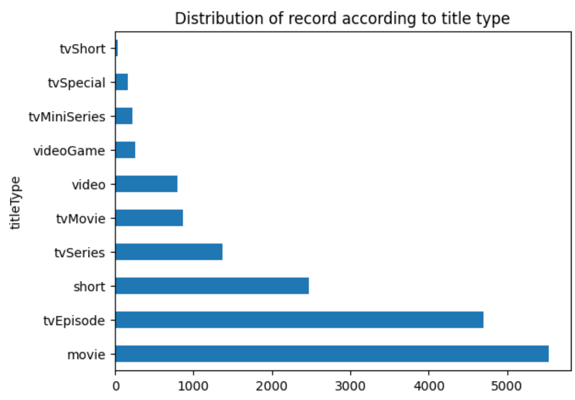

Figure 2: Barplot for *titleType*.

dant as its information is conveyed by the *adult* category in *genres*, therefore we drop it.

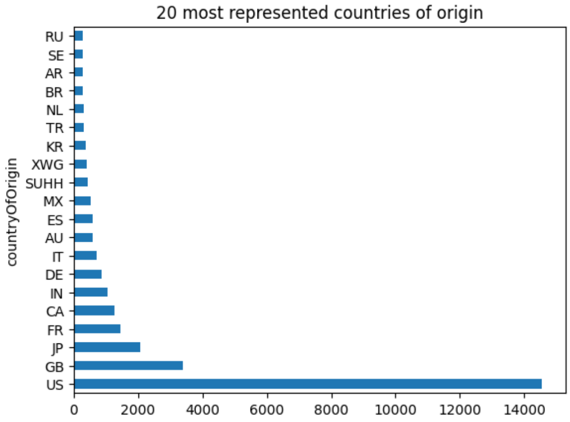

Figure 3: Barplot for *countryOfOrigin* (top 20).

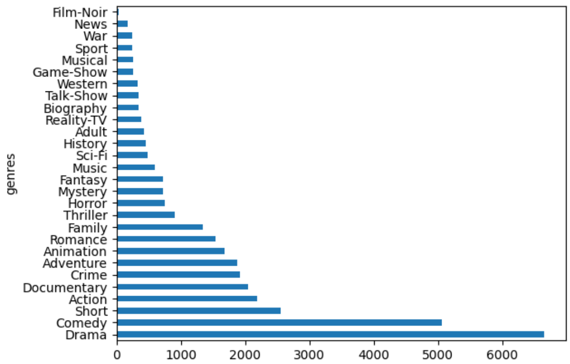

Figure 4: Barplot for *genres*.

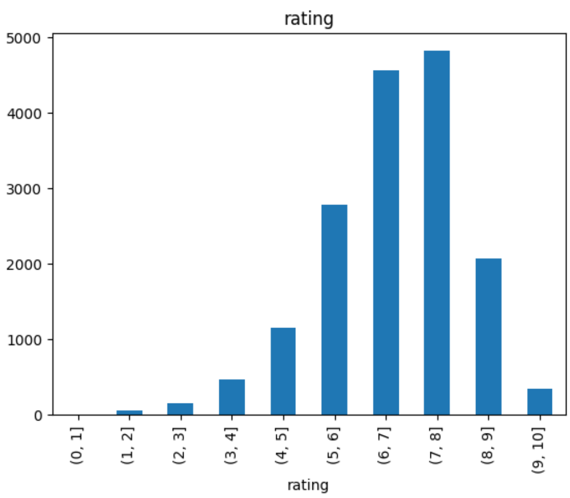

Figure 5: Barplot for *rating*.

1.3 Distribution of numeric variables and statistics; transformation of variables

Before analyzing numeric variables, we add the following features:

- *moreCountriesOfOrigin*: the number of countries listed for the record for the variable *countryOfOrigin*. For the training set, the values range from 1 to 10.
- *numGenres*: the number of genres listed in *genres*. For the training set, the values range from 1 to 3.
- *reviewsTotal*: the sum of *criticReviewsTotal* and *userReviewsTotal*.
- *criticReviewsRatio*: the proportion of review from critics (*criticsReviewTotal*) on the total number of reviews of a record (*reviewsTotal*). For unreviewed records, it is set to zero.
- *awardsAndNominations*: the sum of *awardWins* and *awardNominationsExcludeWins*.

Moreover, we use the available interval features, *startYear* and *endYear* to generate the timeline graph in Figure 6. We can observe how the number of titles being released each year has consistently grown until the end of the 2010s, with two noticeable dips in this trend: around 2001 (which could be linked to the aftermath of 9/11) and after 2020 (probably due to the Covid-19 pandemic). The time series for *endYear* follows the same trend on a smaller scale: we suspect a positive correlation between the two variables. There are no end dates available before the 1940s, which is understandable, as serialized content was not produced until around the 1930s.

In order to have a first global view of both the univariate distribution of the numeric variables singularly, and of possible interesting pairwise correlations, we build a pairplot with a KDE graph for each variable in the diagonal. Our previous intuition about a positive correlation between *startYear* and *endYear* is proved by the scatterplot in Figure 7, therefore we drop *endYear*, which is more problematic due to the large number of missing values.

Most numeric features (*awardWins*, *numVotes*, *totalImages*, *totalVideos*, *totalCredits*, *criticReviewsTotal*, *awardNominationsExcludeWins*, *awardsAndNominations*, *numRegions*, *userReviewsTotal*, *num-*

CountriesOfOrigin, *reviewsTotal*) have heavily right-skewed distributions. It makes sense that the distribution of these types of features would approximate a power law. We can address this problem using a log transformation of these features.

On the other hand, *runtimeMinutes* also has a right-skewed distribution, but it really doesn't make sense for it to approximate a power law. When excluding outliers (values higher than 3rd quartile + 1.5 IQR), we can see that, for the most part, runtimes are dependent on the title type and that the distribution of the runtime for each title type approximates a bell curve (Figure 8); therefore, we prefer not to log transform this variable and be mindful of the presence of outliers. Examining the titles with a longer runtime reveals that part of the outliers is due to the inconsistent strategy used for the computation of the runtime for serialized titles (series and miniseries): in some cases, the episode runtime is provided, in others, it reports the runtime for the entire series, i.e. the sum of all its episodes' runtimes (some examples are reported in Table 4).

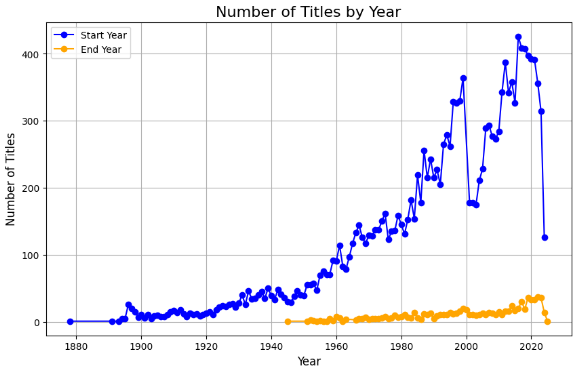

Figure 6: Time series: number of titles being released and ended by year.

originalTitle	runtimeMinutes	titleType
Alim Day1	3000	tvSeries
Jerry Lewis MDA Labor Day Telethon	1290	tvSeries
Voice of the Planet	600	tvMiniSeries
Orbius	570	movie
Heritage: Civilization and the Jews	540	tvSeries

Table 4: Top 5 titles with the longest runtime.

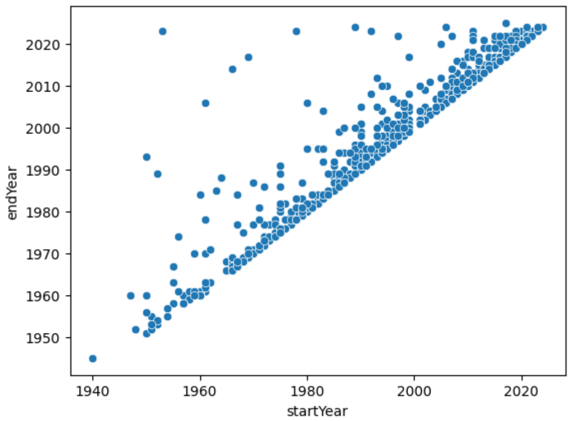

Figure 7: Scatterplot endYear~startYear.

1.4 Handling Missing Values

As explained in Subsection 1.1 and summarized in Table 3, four variables in the dataset have been found to contain missing values: *endYear*, *runtimeMinutes*, *awardWins*, *genres*.

In Subsection 1.3, we opted to drop *endYear* because its value was missing for most records. This decision is corroborated by the fact that, on one side, it heavily correlates with *startYear*, and on the other, the only records that do have an end year are of the type *tvSeries* and *tvMiniSeries*, and the information about the serial nature of media is already carried by the feature *canHaveEpisodes*.

awardWins has almost three thousand missing values. Given its highly unbalanced distribution, with the large majority of the records having no awards, it makes sense to fill them with the most frequent value, i.e. zero.

The stratified barplot in Figure 9 shows that there is some variation in the frequency of differ-

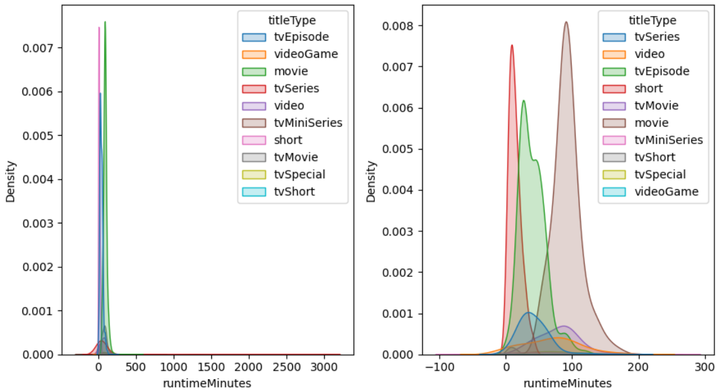

Figure 8: KDE for runtimeMinutes per title type, including and excluding outliers.

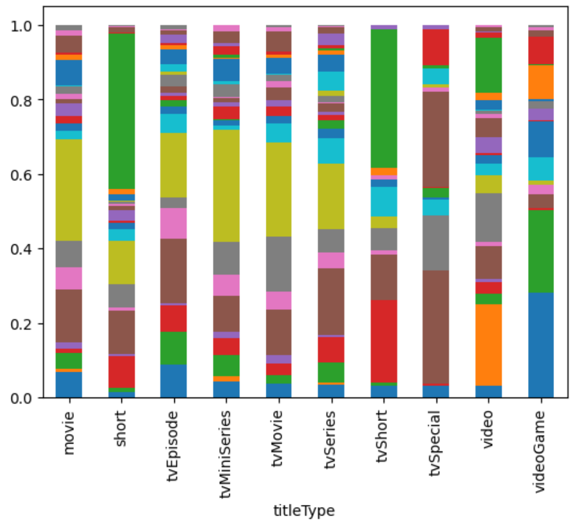

Figure 9: Title type frequency grouped by genre.

ent genres with respect to *titleType*. We decided to fill the 382 missing values for *genres* with the most frequent list of genres given the title type of the record, as shown in Table 5.

titleType	top	# NaN
movie	[Drama]	230
tvSeries	[Comedy]	53
tvMovie	[Drama]	25
video	[Adult]	23
tvSpecial	[Comedy]	18
tvMiniSeries	[Drama]	15
videoGame	[Action, Adventure, Fantasy]	12
tvEpisode	[Comedy]	6

Table 5: Top genre for each title type with missing values for *genres*.

As previously mentioned in Subsection 1.3, *runtimeMinutes* is dependent on the *titleType* of a record (cf. Figure 8): movies are generally longer than TV episodes, and shorts are typically shorter than both. Since we have some missing values for *runtimeMinutes*, we can use the median value for the respective type to fill them (Table 6).

titleType	median runtime
movie	90
short	12
tvEpisode	40
tvMiniSeries	60
tvMovie	86
tvSeries	31
tvShort	10
tvSpecial	60
video	76
videoGame	28

Table 6: Median runtime (in minutes) by title type.

1.5 Pairwise correlations and eventual elimination of variables

We observed pairwise linear correlation among numerical variables (after operating the log transformation detailed in Subsection 1.3) by computing their correlation heatmap, displaying Pearson's correlation coefficient for each pair of features (Figure 10).

numVotes, *userReviewsTotal*, *criticReviewsTotal* and *reviewsTotal* are all highly positively correlated with each other (with a correlation coefficient greater than 70). We only keep *numVotes*, which has the least skewed distribution.

Since the distribution of awards and nominations is not simply skewed, but the majority of records received neither, we binarize *awardsAndNominations* (Table 7) and drop both *awardWins* and *awardNominationsExcludeWins*.

Again, the majority of records do not have videos: we drop *totalVideos* and substitute it with a boolean feature, *hasVideos* (Table 8).

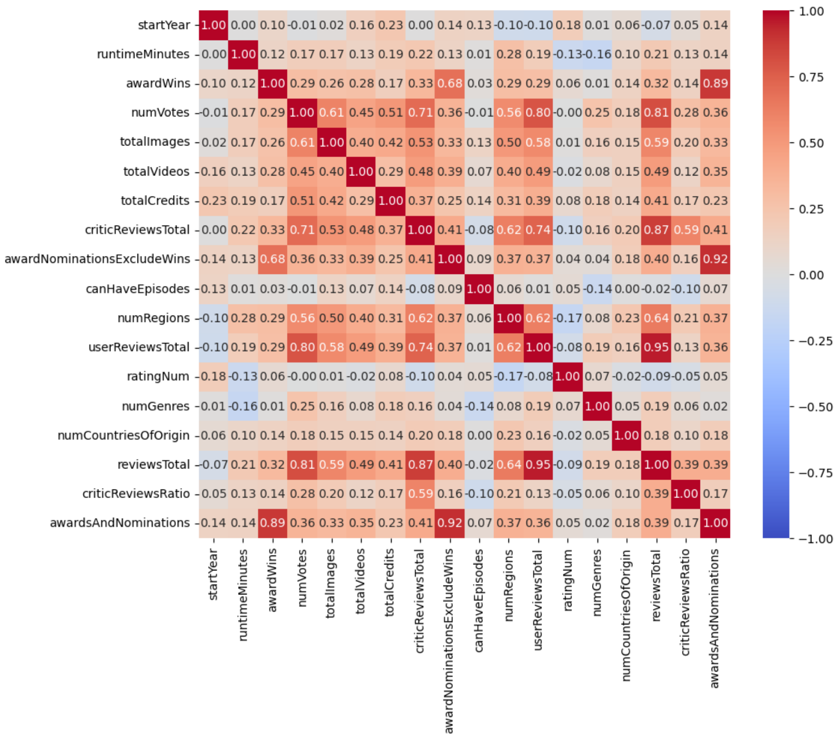

Figure 10: Correlation heatmap before attribute removal.

awardsAndNominations	
False	13692
True	2739

Table 7: *awardsAndNominations* boolean values counts.

hasVideos	
False	14821
True	1610

Table 8: *hasVideos* boolean values counts.

Similarly, most records only have one country of origin, thus we substitute *numCountriesOfOrigin* with the boolean *moreCountriesOfOrigin* (Table 9).

moreCountriesOfOrigin	
False	15285
True	1146

Table 9: *moreCountriesOfOrigin* boolean values counts.

The overall number of features was reduced from 23 to 18; they are listed in Tables 10 and 11. Among the remaining numeric features, we observe in the heatmap in Figure 11 that *numVotes* positively correlates to *totalImages* (0.61, moderate to high correlation), to *numRegions* (0.56, moderate correlation), and to *totalCredits* (0.51, low-moderate correlation).

2 CLUSTERING

3 CLASSIFICATION

4 REGRESSION

5 PATTERN MINING

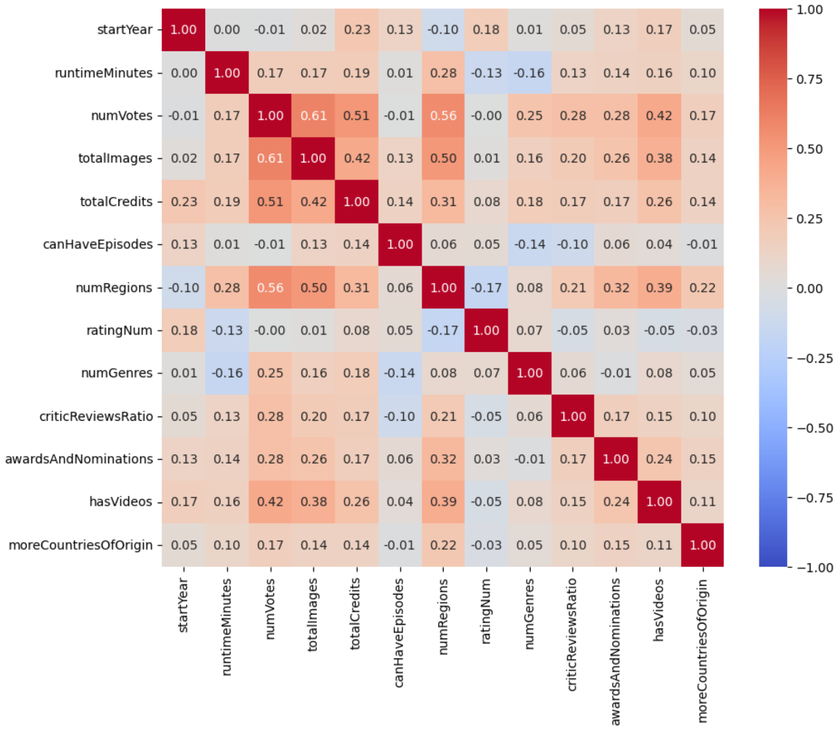

Figure 11: Correlation heatmap after attribute removal.

Feature	Description	Type
runtimeMinutes	Primary runtime of the title, in minutes.	Continuous. Ratio-scaled.
startYear	Represents the release year of a title. In the case of TV Series, it is the series' start year.	Discrete. Interval.
numVotes	Number of votes the title has received.	Discrete. Ratio-scaled. Log-transformed.
numRegions	The regions number for this version of the title.	Discrete. Ratio-scaled. Log-transformed.
totalImages	Total number of images for the title within the IMDb title page.	Discrete. Ratio-scaled. Log-transformed.
totalCredits	Total number of credits for the title.	Discrete. Ratio-scaled. Log-transformed.
criticReviewsRatio	The proportion of reviews from critics on the total number of reviews of a record. For unreviewed records, it is set to 0.	Continuous. Ratio-scaled.
awardsAndNominations	Total number of awards and nominations.	Discrete. Ratio-scaled. Log-transformed.
numGenres	The number of genres listed for the variable *genres*.	Discrete. Ratio-scaled.

Table 10: Final numeric attributes.

Feature	Description	Type
originalTitle	Original title, in the original language.	Categorical; mostly unique classes (i.e. names).
titleType	The type/format of the title (e.g. movie, short, tvseries, tvepisode, video, etc.).	Categorical, 10 classes.
countryOfOrigin	The country where the title was primarily produced.	Categorical, 153 classes. Some titles can belong to more than 1 class.
genres	The genre(s) associated with the title (e.g., drama, comedy, action).	Categorical, 29 classes. Some titles can belong to more than 1 class.
rating	IMDB title rating class.	Ordinal, 10 values.
ratingNum	Mapping of *rating* to the integers [1,10].	Ordinal, 10 values.
canHaveEpisodes	Whether or not the title can have episodes.	Asymmetric binary.
hasVideos	Whether or not the title IMDb title page has at least one video.	Asymmetric binary.
moreCountriesOfOrigin	Whether or not the title was produced in more than one country.	Binary.

Table 11: Final categorical, ordinal, and binary attributes.