Construction of an Age Certification Predictor for Movies and Shows Streamed on *Netflix*

Shiyuan Zhou - 1346384
Wen Harng Justin Goh – 1268654
Jasir Syed – 1272460
Shuxuan Wang- 1407511

COMP20008
Elements of Data Processing
Assignment 2
Submitted on: 9th October

Contents

1 Aim

In this report, we determined if the age-certification of various movies and TV shows can effectively be classified using a dataset that summarises key characteristics of streamed titles on *Netflix*. While the dataset is provided by the aforementioned streaming platform, the attributes are more general and applicable to most TV shows and movies (and therefore other potential dataset). Therefore, such a model has the potential to streamline the certification process for the agencies responsible for the task such as the Australian Classification Board (ACB). Alternatively, the model can be consumer-facing; it could act as a validation method for provided certifications that would allow families to exercise better parental controls and reduce the risk of children being exposed to explicit content.

2 Dataset Description

There are two datasets provided: credits.csv and titles.csv. The titles.csv contains over 5,850 rows of TV shows and movies. It has a total of 15 features including **5** numerical features. There are **8** categorical features, providing comprehensive information about the nature, origin, and classification of each title in the dataset. In addition, there are **2** textual features, including title and description.

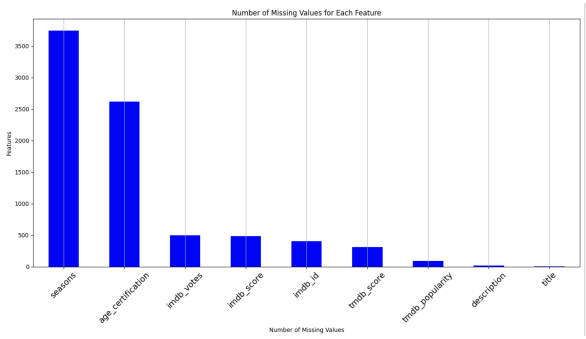


Figure 1: Distribution of missing values by features

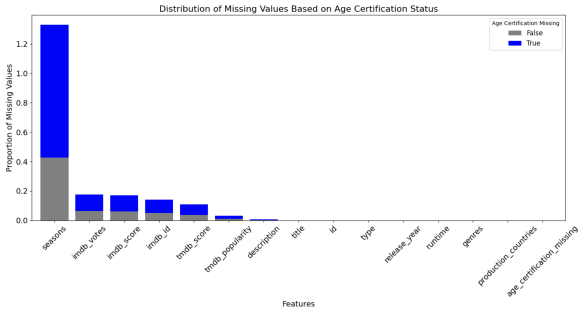


Figure 2: Distribution of Missing Values Based on Age Certification Status

Upon examining the dataset, distinct patterns emerge in the missing age certification entries. Movies have fewer certifications compared to shows, and titles that have more recent release year have slightly more missing certifications. Most notably, titles with longer runtime more often have more missing age certifications. These patterns indicate that the absence of data is not entirely random. Instead, it is influenced by features such as the type and runtime. This observation aligns with the "Missing at Random" (MAR) classification.

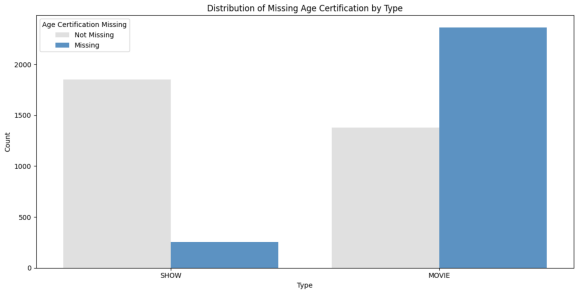


Figure 3: Distribution of Missing Age Certifications by Type

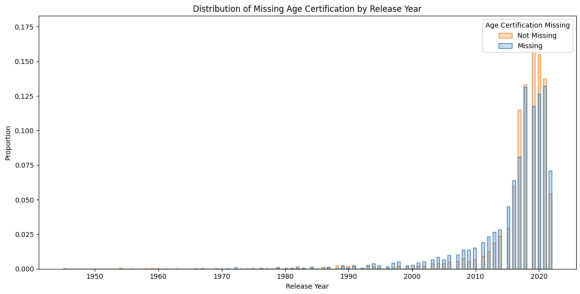


Figure 4: Distribution of Missing Age Certification by Runtime

3 Pre-processing

3.1 Preliminary Steps

Imputation is a common technique for handling missing data, but was inappropriate for our situation. Using median or average values for missing IMDB or TMDB scores may introduce bias if the data is not missing at random (MNAR). Furthermore, median imputation and most other techniques are not applicable to the description column. Finally, imputation was not conducted on the age-classification column as correctness in our model's target variable is essential for reliable accuracy measurements. The dataset has a large bias for R-rated movies and TV-MA shows. From domain knowledge, we know that these are not the most frequent certification and in fact are a minority [2]. In summary, it can be extrapolated that the subset of the data that is not missing age certifications is also not representative of the distribution we would expect in said feature. Therefore, mode imputation could potentially help our model work with this particular dataset, but it would likely make it inaccurate in its general purpose of classifying movies and TV shows when working with others.

The data-set was split into two distinct sections: movies and TV shows. Noting that the difference in entertainment medium may result in unique data patterns and trends, this was deemed an optimal compartmentalization step. This makes the *'seasons'* column in the movies sub-dataset redundant and it was consequently dropped.

It was noted that in the case of, both, *'production_countries'* and *'genres'*, string representations of lists were used, so these were converted into processible, Python lists.

Both IMDB and TMDB ID columns were dropped as these are useful for lookup purposes, but provide us with no utilizable information.

3.2 Description

Standard techniques for pre-processing text were employed to handle the *'descriptions'* column. Case-folding (lowercase), removal of non-alphabetic characters, tokenization, removal of stop words and, finally, lemmatization were employed.

3.3 Encoding

In order to conduct feature selection for our model, correlation identification methods such as mutual information (MI) calculations or Pearson coefficient analysis will be required. While the former is designed to be applicable to categorical data, most library functions and procedures that employ it still require numerical data. One-hot encoding is an effective technique for handling this issue and was applied to the categorical columns that we were interested in.

3.4 Outliers

Before any filtering for outliers, it was important to consider what could effectively be considered an outlier in the context of our dataset. An initial analysis would suggest that instances with disproportionately high popularity and scores and votes on IMDB and TMDB should be omitted. However, further consideration of domain knowledge would suggest keeping such scores is recommended - they represent instances of great critical success and being able to handle such cases would make our model industry-applicable. On the other end, low popularity and IMDB scores are worth keeping as they may allow for identification of trends relating age certification to commercial viability.

This leaves only one category for consideration: low IMDB votes. Similar to the TMDB popularity distribution graph below, IMDB votes were found to have a strong positive skew. This meant that without log transformation (which was another pre-processing step undertaken), methods like the IQR technique did not actually identify any low-scoring outliers. Consequently, removing outliers was deemed an unnecessary step, but further exploration of the impact of this omission could be a potential next step.

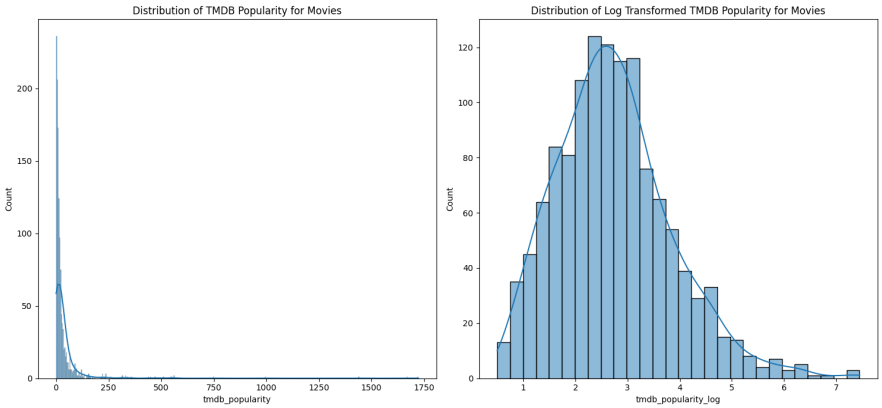


Figure 5: Example of log transformation

3.5 Discretization

Mutual information was selected as the correlation identification technique (see section 4 for justification), but for it to work, all our features need to be made categorical. In the case of continuous, numerical data such as IMDB scores, this can be done by discretization.

The discretization method of choice depends on various factors. In the case of, both, IMDB and TMDB scores, equal-width binning was selected. Equal-width naturally leads to a total of ten bins in both cases for each integer value from one to ten - scores that are easy to interpret for the experimenter compared to the random intervals that would be generated as a result of equal-frequency binning. Equal width was also utilized for the other TMDB/IMDB features using similar arguments.

To discretize runtimes, the distribution within the dataset along with consideration of domain knowledge was used. For TV shows, this lead to bins which roughly corresponded to short-form shows, standard 30-minute episodes, mainstream (45min) shows and long-form TV [1]. Similarly, for movie the bins roughly corresponded to short films, short feature films, mainstream feature films, longer feature films and finally epics [2][3].

The last variable to discretize was the release year. Unlike run-time, it is reasonable to assume a similar distribution for both movies and TV shows and so they were considered concurrently. Domain knowledge was not utilized extensively, with the data distribution being focused on instead.

In summary, the discretization technique chosen varied between different attributes, with the chief goal of creating reasonable and qualitatively meaningful bins for any potential future analysis.

4 Feature Selection

4.1 Methodology: Mutual Information and Word Clouds

Mutual Information calculations was the selected technique for identifying correlations. Since the target feature of our model is categorical (making this a classification problem) as are several of the other attributes, Pearson correlation was not appropriate. While it is true that one-hot encoding has been used to turn these features into numerical values, said values are not expected to have a linear relationship with age-classification. Thus, MI remains the clear choice.

Note that as part of the encoding process, the dimensions of our data set was increased several-fold, adding columns for vector representation of each potential genre and production country. Consequently, to calculate the MI value between the aforementioned two features and our target attribute, the MI value between each vector and said target must be computed.

We hypothesized that the description feature would prove to have minimal correlation with the age-classification. Vectorizing the feature would increase our already-wide dataset by several dozen dimensions and would require PCA in order to keep the shape of our dataset appropriate. This level of effort for an attribute not predicted to be a focal component of our model was deemed unproductive. To validate our decision, we instead generated word clouds representing the most frequently occurring words for each age classification. These clouds were then compared by vectorizing the words found within and using cosine similarity (visualized by a confusion matrix).

4.2 Correlation Analysis

- **Word Cloud:** Two clear features can be identified in the confusion matrix. Firstly, the similarities are considerably high - the lowest score is 0.49 in the case of. Secondly, the distribution of scores is pretty small (bar the one outlier of 0.8). From this we can effectively conclude that there is significant overlap as well as consistent similarity across the age certifications. It is, therefore, unlikely that the description feature will prove useful for our model as it lacks uniqueness when compartmentalized by our target feature.

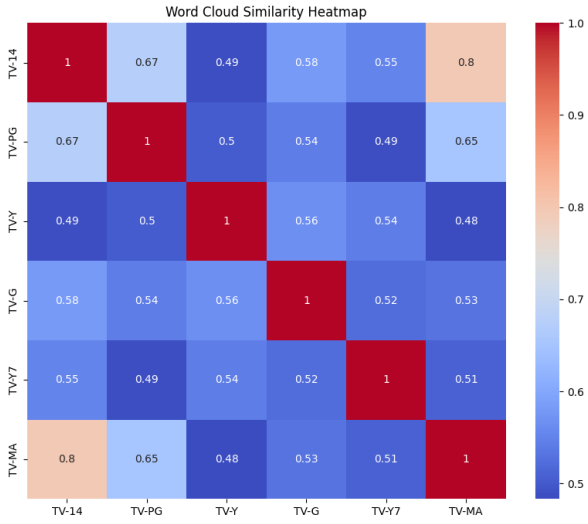
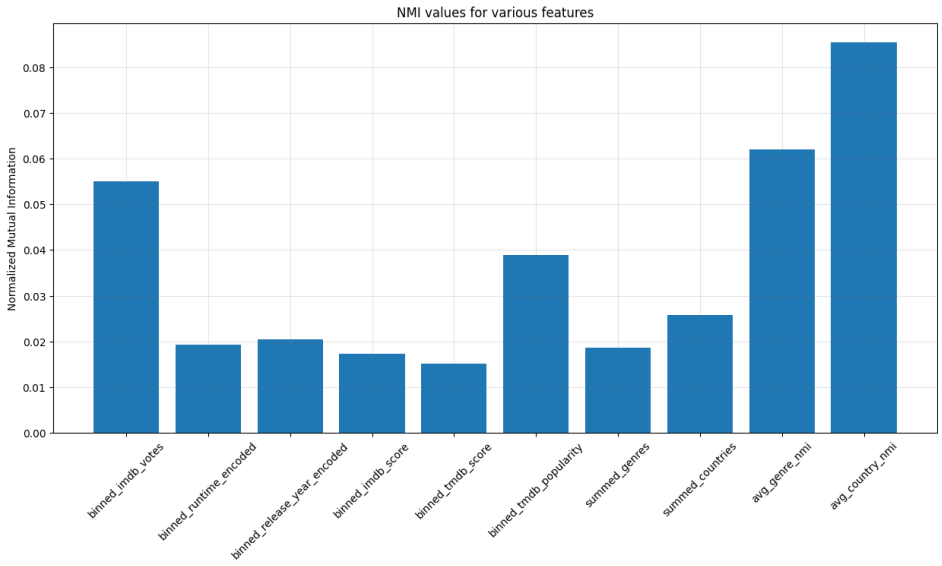


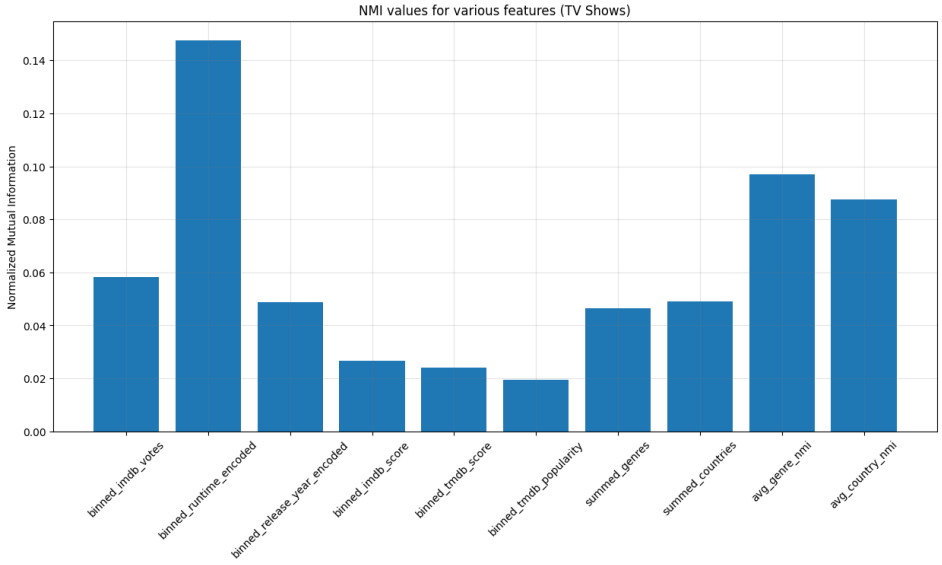
Figure 6: Cosine similarity confusion matrix for TV certifications

- **Mutual Information Scores:** The results of the normalized mutual information scores (NMI) proved quite disappointing, with no score even reaching 0.1 for movies and only runtime passing that threshold for TV shows. However, the presence of features with relatively higher MI scores suggest that it possible the low scores might be a byproduct of our data filtering steps (see discussion on limitations). Consequently, the five features with the highest scores were still selected and the model was proceeded with. Hypotheses were presented for each 'highly-correlated' feature identified and utilized in our model.
  - **Production countries:** certain countries are known for having trends in their content maturity preferences. For instance, French cinema is somewhat infamous for being more relaxed in producing erotica.
  - **Genres:** different genres are expected to correlate to different age certifications. For instance, a thriller is far more likely to be rateed R than PG.
  - **IMDB votes and TMDB popularity:** lower maturity levels of content might be more accessible to consumers, leading to higher mainstream engagement.
  - **Runtimes (for TV only):** channels such as *Nickelodeon* and *Cartoon Network* are known for using shorter run times and this trend holds generally true for shows targeting younder audiences (i.e. shows with lower maturity ratings/age certifications).

The above identified features are the ones inputted into one of our models (not both for reasons discussed in the next section).



(a) NMI - Movies



(b) NMI - TV shows

Figure 7: NMI values

4.3 Train-test split

An 80-20 train-test split was applied to the data post-processing steps. This is standard procedure in creating an ML model: training set allows for the tuning of our model, while the test set provides an opportunity to assess performance with unseen data. Making sure there is no overlap also helps prevent any data leakage.

4.4 Zero-R

In order to have a baseline measurement to interpret the accuracies of our models, a Zero-R model was constructed that simply predicts the most frequently occurring age certification for all instances. Despite the simple approach, this model was found to have an accuracy of 41.5% and 49.2% for movies and TV shows respectively. These relatively high accuracy values for a naive model provide confirm the high imbalance present in our dataset.

Due to the aforementioned imbalance, an accuracy score on its own will not prove particularly useful. Recall, precision and f1 scores will allow for proper weighting of results as they will provide a more holistic insight into model performance.

dataset	accuracy	balanced accuracy	recall	precision	f1
movies	0.42	0.2	0.42	0.17	0.24
shows	0.49	0.17	0.49	0.24	0.32

Table 1: Zero-R Performance Metrics

4.5 Decision Tree

As has been established, this is a classification problem. This makes supervised learning techniques that do not expect linear relationships appropriate - including decision trees.

We did not extensively reduce input features, bearing in mind that decision trees have automatic feature selection as part of their implementation. Furthermore, our MI scores were not particularly promising, so we decided to let the embedded feature selection find correlations itself (our comparison model will be used to judge the efficacy of our correlation).

The dataset was subjected to 10-fold cross-validation to ensure robust evaluation with Hyperparameter tuning. This was varied from 1-29 (note the high possible depth due to our encoding of categorical features). The use of 10-fold cross-validation ensures that the model's performance is robust and less prone to overfitting on a particular subset of the data. By averaging scores over multiple folds, a more generalized metrics of the model's performance is obtained.

For movies, the best performance seemed to be for tree depths between 4 or 5 with a peak accuracy of approximately 0.54. Beyond this range, any improvement may likely just be overfitting, but more often than not there is actually a decrease in performance.

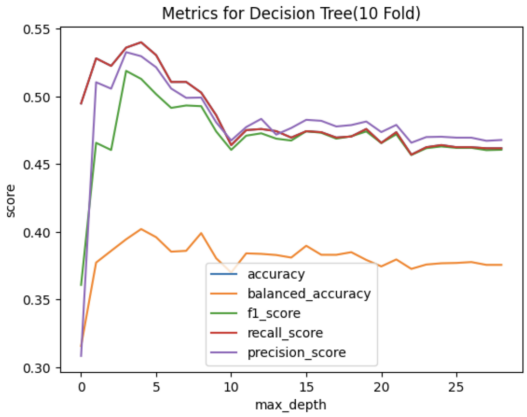


Figure 8: Metrics for Decision Tree Hyperparameter Tuning (Movies)

For TV shows, the model peaked in accuracy at roughly 0.58 when the tree depth was 5, and the best balanced accuracy was around 0.47 at a tree depth of 11. In addition, the highest F1 score was noted at a tree depth of 11, approximately 0.56. It is worth noting, however, that these results plateau after 5-folds - it is possible any imporvementts after this stage may just be overfitting and so 5 folds was once again selected.

The performance metric results for our 5-fold movies model and show model have been summarized in tabular form. Confusion matrices were also generated.

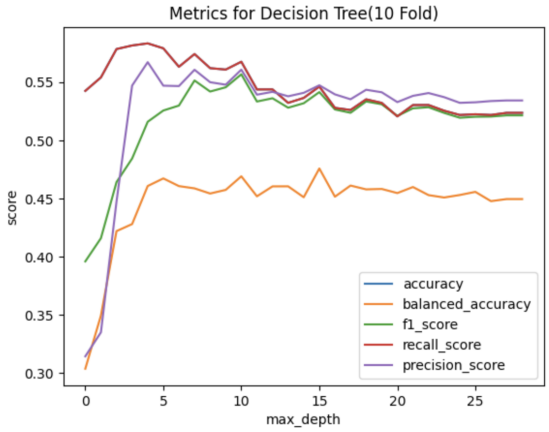


Figure 9: Metrics for Decision Tree Hyperparameter Tuning (TV shows)

dataset	accuracy	balanced accuracy	recall	precision	f1
movies	0.54	0.40	0.52	0.54	0.53
shows	0.58	0.47	0.58	0.58	0.56

Table 2: DT final results(max_depth = 5)

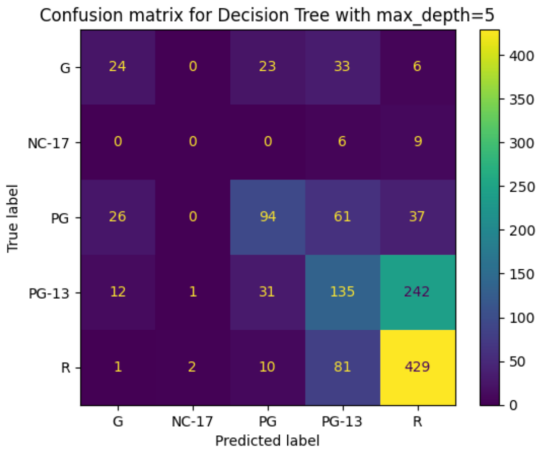


Figure 10: DT confusion matrix (Movies)

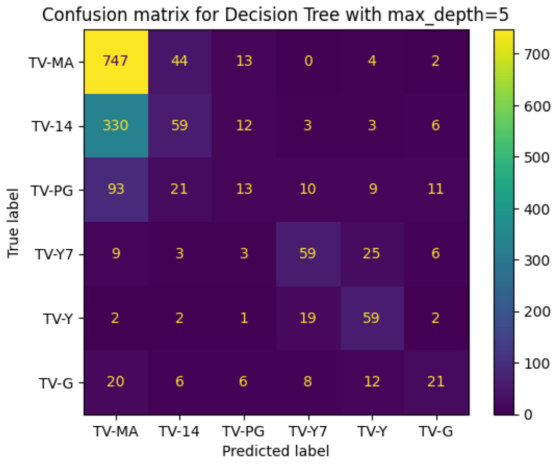


Figure 11: DT confusion matrix (TV shows)

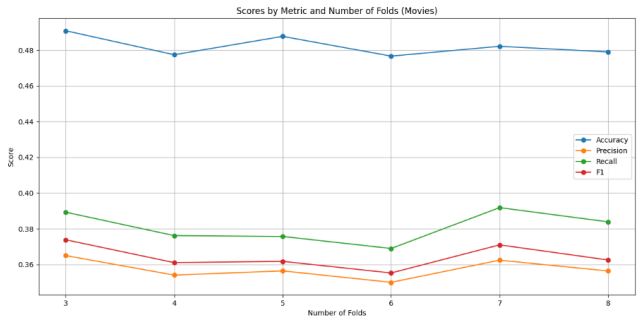
4.6 KNN

Another supervised learning technique that can be utilized for classification and compared to our tree is KNN. Since KNN does not have embedded feature selection, attributes identified as being potentially correlated with age certification from the previous section were utilized, namely production countries, genres, tmdb popularity and imdb votes for movies, and runtime, genres, productions and imdb votes for TV shows. Firstly, the appropriate number of folds were determined. For movies, accuracy seemed to peak at 3 folds, precision at 8, recall at 7, and f1 also at 7. This lead to the use of 7 folds. Similar analysis was done for TV shows and 7 folds was determined to again be the best choice.

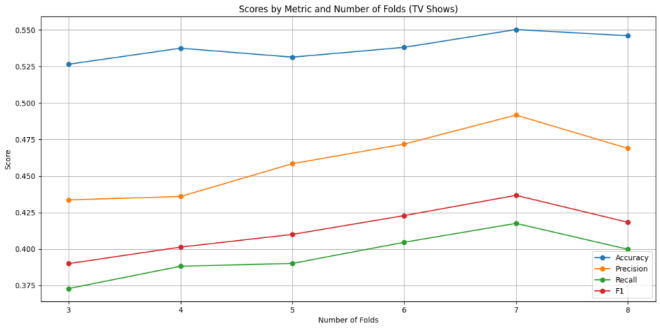
Next, the appropriate number of neighbors was determined and the determined values were 5 and 3 for movies and TV shows respectively (see graph). Normally, it would be preferred if the features and neighbours parameters could be tuned simultaneously, but we assume that they are independent to simplify model construction.

The final results for the two separate KNN variants are summarized.

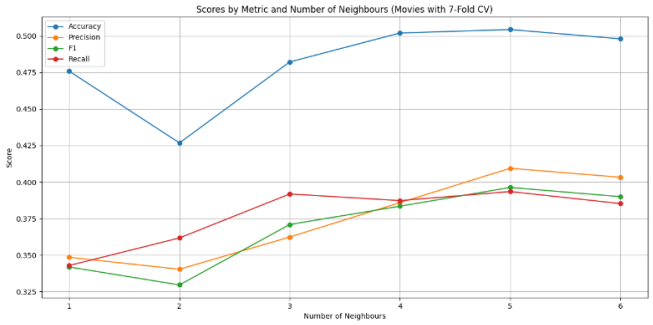
Overall, the performance seems to be lower than that of our decision tree, but the trends are the



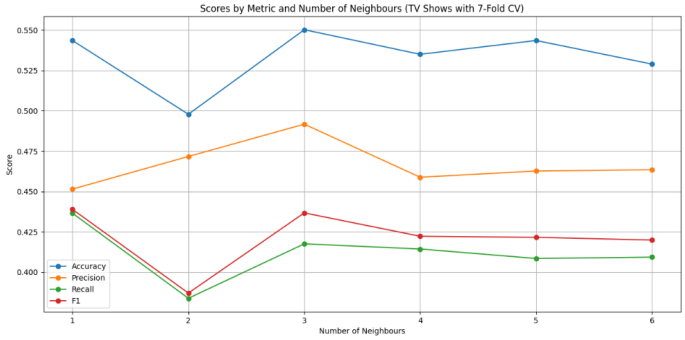
(a) Performance of StratifiedKFold across 3-8 number of folds



(b) Performance of StratifiedKFold across 3-8 number of folds



(c) Performance of 7 StratifiedKFold across 1-6 neighbors



(d) Performance of 7 StratifiedKFold across 1-6 neighbors

Figure 12: KFold and Neighbor Parameter Tuning for KNN

same.

Dataset	accuracy	recall	precision	f1
movies	0.43	0.31	0.30	0.53
shows	0.51	0.36	0.44	0.39

Table 3: KNN final results

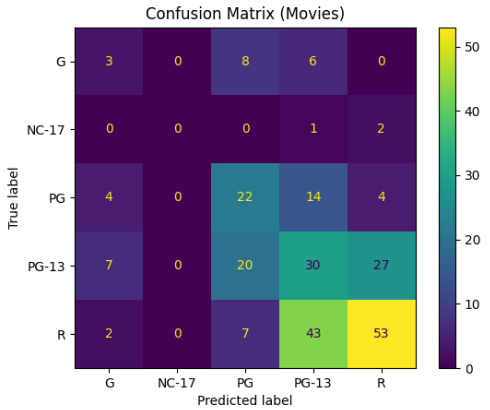


Figure 13: KNN confusion matrix for movies

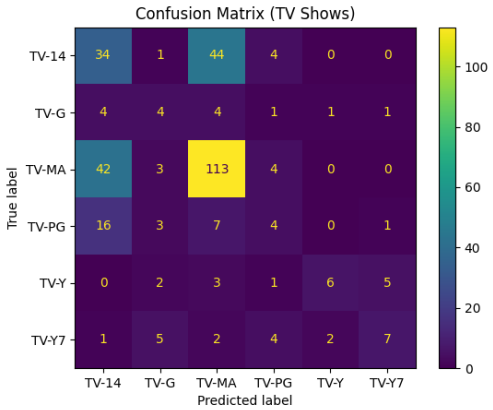


Figure 14: KNN Confusion matrix for TV

5 Evaluation and Limitations

In the case of the decision tree, modest improvements in accuracy were seen when compared to the Zero-R model. The improvements to the other metrics were more impressive, but the overall efficacy remains below what would be desired. The performance scores for TV shows seem to be better than those for movies - but the difference is limited. This suggests that TV show age-certifications are not necessarily easier to classify and the improvement is likely the result of the dataset having a slight tv-majority and by extension, more training data for one over the other.

The fact that the model even does work (slightly) implies issue with our mutual information calculations, which had scores low enough to expect little to no improvement from baseline as opposed to anything noteworthy. It is hypothesized that the MI scores would be considerably higher and the size of our dataset is the likely limitation (also further dicussed later). Beyond increasing the size of our dataset, a possible improvement could be using unsupervised learning to identify potential clusters of genres and production countries (remembering that each instance is not limited to only one vector for each of the two and can be produced in multiple countries and/or have multiple genres). A similar strategy could be utilized for run-time and IMDB/TMDB popularity scores - it is possible that our emphasis on domain knowledge and discrete bins that are consumer-informative as opposed to equal

frequency/equal-width caused a severe loss in precision during the discretization process - a problem that is already expected when discretizing continuous variables and could be exacerbated by choice of strategy. Having better MI scores would also enable potential adjustment of the class-weight parameter for the decision tree.

The improvements for specific age classifications did, however, surprise us. For movies, the 'R' rating actually performs relatively well. The ratio of true positives to other results is impressive when observing the confusion matrix - the recall is roughly 76% , a huge increase from the zero-r model. Now there is a valid concern: the dataset has a disproportionate ratio of R movies to the other age certifications which could be responsible for training the model to produce plenty of false positives. But if we look at the precision for R-rated movies, it also seems relatively high, sitting at 60%. It is important to emphasize the relativity - independently, these scores are not particularly impressive, but they do represent a stark improvement from the performance across the other age-certifications and the zero-R as a whole. In fact, observing the confusion matrix, there appears to be a roughly proportional relationship between the quantity of instances with an age-certification present in the dataset (see figure from section 2) and it's overall performance.

This trend is not limited to movies. The TV-MA certification, which remains the most frequent classification in the dataset, also performs relatively well with a precision of 65% and a recall of 82%. Both scores are an imporvement when compared to R-rated movies further supports the claim that the greater the amount of training data for each age-certification, the better our model will perform. This consolidates our improvement suggestion of a larger dataset, but also emphasizes the need for consideration of data distribution and ensuring ample instances of all possibilities in certification.

The KNN model was created as a way to compare the supervised learning techniques and, overall, did not perform as well as our tree. However, the trends found in the decision tree were also present and so the same suggested improvements, analyses, and interpretations apply. While it may be tempting to declare DT's the more suitable technique for our model, we would be hesitant to draw strong conclusions before further training and implementing the improvements for MI calculations that have already been suggested. If anything, the lower performance of the model only further cements that our feature analysis was inherently flawed as the findings of that stage were not utilized for the DT but were for our KNN tuning. In layman's terms, our KNN utilized our feature selection process and did worse than our decision tree which didn't, suggesting that our correlation analysis was inherently flawed.

As a final comment, it should be noted that the dataset in its original form may well have had enough instances for more effective feature selection as well as model training; nearly 3000 of the roughly 5.5k instances of our dataset were missing age-certifications and were unusable for training. Despite this, we still stand by our decision to discard said instances as opposed to utilizing mode imputation - we've established that the dataset is incredibly imbalanced towards the R and TV-MA rating and so mode imputation would have created incredibly unrepresentative training data. If our model was trained on such data, it would be incredibly ineffective on other movie/TV show datasets and would likely have many false positives for the aforementioned certifications.

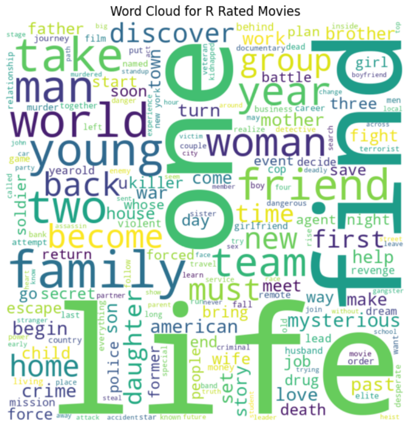
6 Conclusion

In summary, at first glance, our decision tree was not as successful as one would hope, but a deeper dive into the subsets of data within our dataset revealed that there were some noteworthy results. To be specific, the model worked relatively well for the most common age certifications in our dataset (R and TV-MA) and seems to follow a proportional relationship between number of instances in the dataset and performance. It is possible that our model could be more effective - the key improvements that would be required is a larger dataset that has better representation of age-certification subclasses and a re-evaluation of our discretizing process - possibly with the use of unsupervised learning. Unfortunately, the KNN model performed even worse, but further cemented our hypothesis that one of the primary issues was likely the feature selection process.

7 References

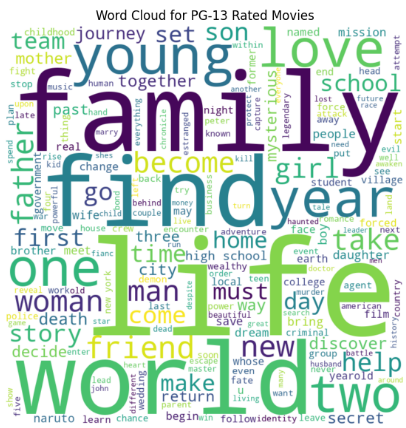
[1] https://joneswriter.medium.com/how-long-should-tv-shows-be-the-rise-of-shorter-tv-6f23a83d572d
[2] https://www.arcstudiopro.com/blog/what-is-a-feature-film
[3] https://movieweb.com/best-epic-movies/

8 Appendix



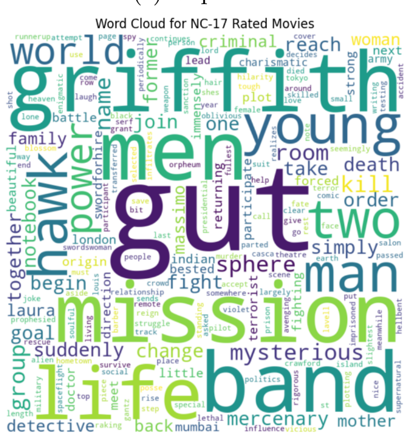
(a) Caption 1

(b) Caption 2

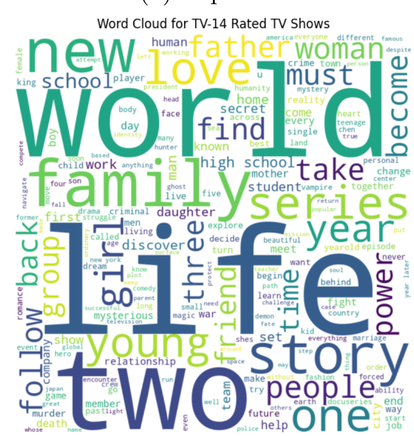


(c) Caption 3

(d) Caption 4

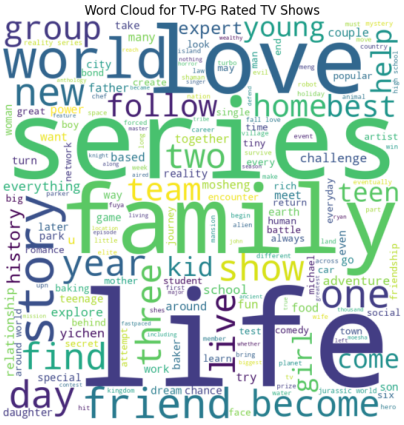


(e) Caption 5

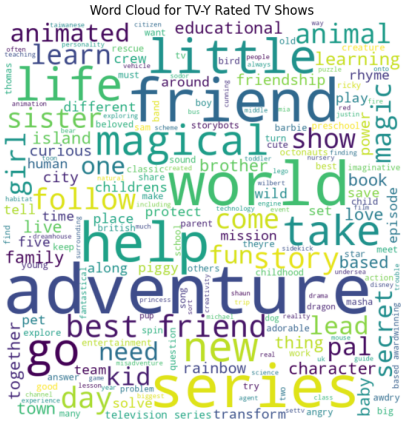


(f) Caption 6

Figure 15: Main Caption (Part 1)



(a) Caption 7



(b) Caption 8

(c) Caption 9

(d) Caption 10

(e) Caption 11

Figure 16: Main Caption (Part 2)