1 Problem Statement

Given the following information about a movie or TV-show title, to what extent can its popularity be predicted?

- The year at which the title was first released,
- The run-time length of the title.
- The genre of the title.
- The director(s) and writer(s) of the title.
- The main principal(s) in this title.

2 Data Preprocessing

The official IMDB Datasets lists about six million titles and millions of names (including directors, writers, and principals.)

The name attributes of each title required heavy pre-processing. Encoding these features directly would likely require an unrealistic amount of memory. Due to the lack of correlation between each names, neither might that be very fruitful. Hence (after consulting members of the McGill AI Society,) for each title, the only name feature being stored would be the average rating of the first three directors, writers, and principals.

Some of the names do not have any rating information in our records. The assumption is that directors play a main role in the outcome of a movie. Hence, for titles where no rating information is available for the principals (actors,) that field will be filled with the rating for the director. In the very unfortunate case where no rating information is available for the director, the title would be discarded.

When compared to the names, the cardinality for the genre(s) field of each title is far more manageable. This field is encoded with a multi-label binarizer.

The average rating information is initially continuous between 0 and 10. Due to the classifying nature of this model, that range is divided into 20 subclasses: $0.0 \leq x \leq 0.5, \ldots, 9.5 \leq x \leq 10.0]$.

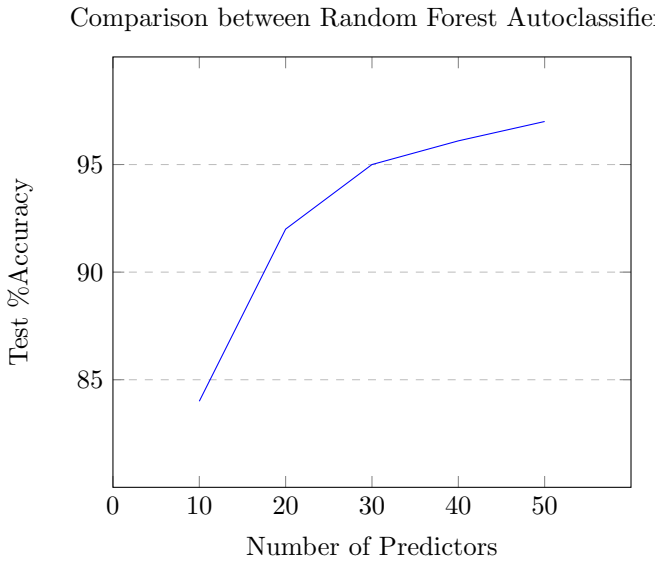
3 Machine Learning Model

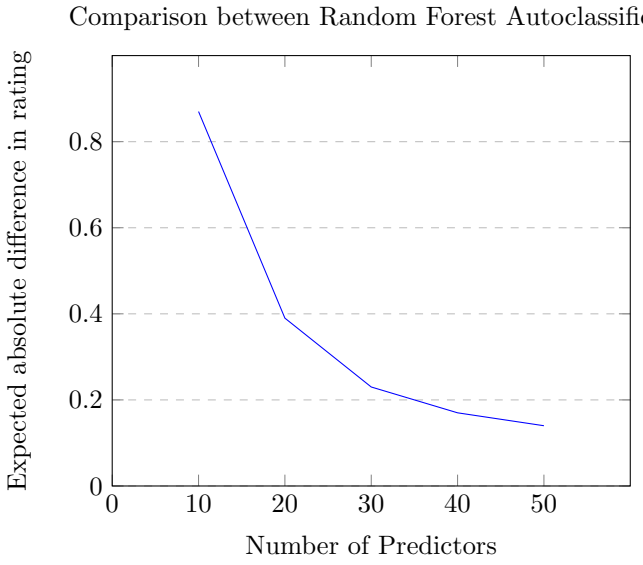
Roughly 30 % of the dataset is set aside for testing purposes. Due to the continuous nature of the average rating, the expected difference between prediction and expected outcome would likely be a better benchmark for the models.

4 Preliminary Results

Random Forest

- With 10 estimators ("trees") and an unlimited depth, the auto-classifier achieved an 84 % accuracy on the testing set. The mean difference between predicted rating and actual rating is around 8.7 % (0.87 out of 10.)
- With 20 estimators ("trees") and an unlimited depth, the auto-classifier achieved an 92 % accuracy on the testing set. The mean difference between predicted rating and actual rating is around 3.9 % (0.39 out of 10.)
- With 30 estimators and an unlimited depth, the auto-classifier achieved a 95 % accuracy on the testing set. The mean difference between predicted rating and actual rating is around 2.3 % (0.23 out of 10.)
- With 40 estimators and an unlimited depth, the auto-classifier achieved a 96 % accuracy on the testing set. The mean difference between predicted rating and actual rating is around 1.7 % (0.17 out of 10.)
- With 50 estimators and an unlimited depth, the auto-classifier achieved a 97 % accuracy on the testing set. The mean difference between predicted rating and actual rating is around 1.4 % (0.14 out of 10.)





5 Next Steps

- Experiment with other auto-classifiers and neural networks.
- Build an interactive web-app for making predictions and comparing outcomes from different auto-classifiers.