AniML : Automatic scoring prediction of Japanese Animation

Valentin KAO and Emile JONAS

Department of Computer Sciences
Tsinghua University

Introduction

The Japanese anime industry is a major worldwide success thanks to the interest of other Asian countries. In 2017, the industry pulled in a record $17.7 billion last year, and boosted by the famous hit, Your Name, growing exports and revenues [1].
Being able to predict the future smash hits is of great importance in decision making specially in the marketing. In spite of the fact that there are many factors that influence the success of an anime. It is not generally clear how they influence, this project endeavors to determine these factors using machine learning strategies by predicting a score out of 10 based on the data provided by the famous website MyAnimeList.

Data

Data Collection

As no adequate dataset existed for this task, we built one, using a Python scraper on MyAnimelist. MyAnimeList is currently the biggest anglophone website about anime. It's a social cataloging application, which means it has both detailed information about anime and a corresponding users aggregated scores. After cleaning, the dataset roughly got 10,000 entries.

Data Cleaning and Improvement

As expected from raw input, a lot of data were incomplete. Either old entries missing information or new entries where the data didn't existed yet. We also removed data that were out of our scope (18+ for example).
We replaced non-numerical data by weighted ID. Data values were spread in a really inconsistent way. We used standardization to better fit our algorithm requirement.

Method

Feature selection

Currently, the algorithm is using some features like the genre, the date (month and year), weighted related projects, format (episodes/duration), etc. It also use multiple information about users opinion like the total score, how it is spread between values, "favourites", dropped, etc.
More complex features (weighted voice actors score, weighted staff score) are to be added in the next versions. We abandoned features like licensors because they didn't had any beneficial impact on the result. To select (or remove) a features, we determine how correlated the feature is with the others.
Moreover, we also used The Recursive Feature Elimination to select the most appropriate features. It uses the model accuracy to identify which attributes (and combination of attributes) contribute the most to predicting the target attribute.

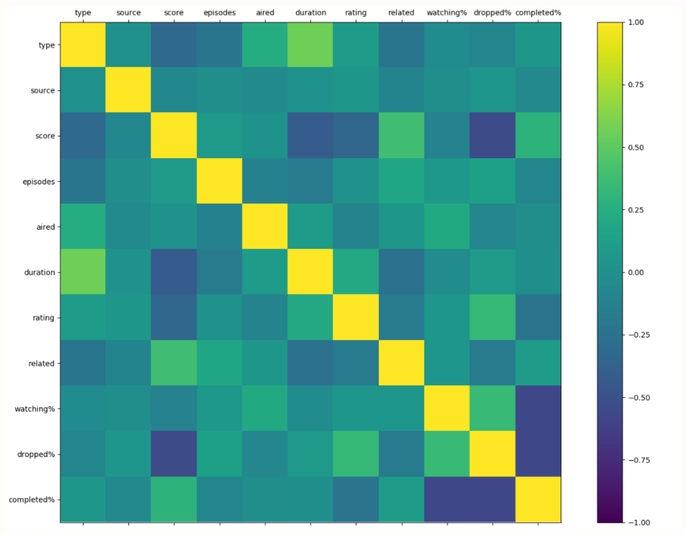

Figure 1: Correlations graph between attributes

Regression Algorithm Selection

Since we have small dataset, the selected size of the split of our dataset was 90% of the data for training and the remaining 10% for testing. Then, six different algorithm were selected to work on this regression problem. With the default tuning parameters, we performed 3 linear algorithms (Linear Regression, Lasso Regression and ElasticNet) and 3 nonlinear algorithms (Classification and Regression Trees, Support Vector Regression and k-Nearest Neighbors) on our dataset. For evaluating the predictions on these algorithms, we use the Mean Squared Error (or MSE) metric defined by the following formula:

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2.$$

with $\hat{Y}$ is a vector of n predictions, and Y is the vector of observed values of the variable being predicted.

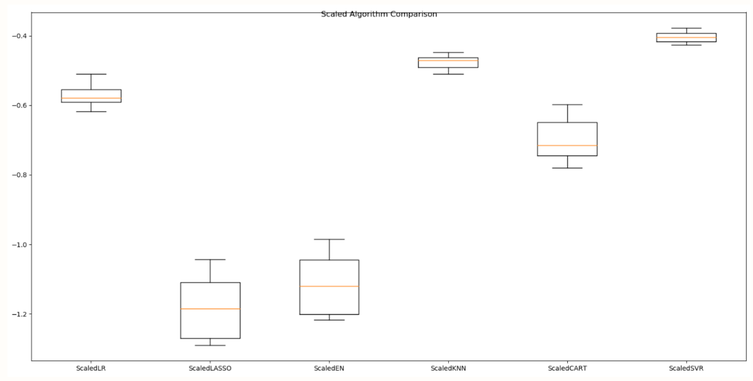

Figure 2: Differences in Algorithm Performance on Standardized dataset

Then, we can see that Support Vector Regression has both a tight distribution of error and has the lowest score. To find the most optimal values, we use the Grid search to parameter tuning that will methodically build and evaluate a model for each combination of algorithm parameters specified in a grid. We obtained the following parameters:

Table 1: SVR Settings

Parameter	Value
C	5.0
Gamma	'0.1'
Kernel	'rbf'
Epsilon	0.1
Tol	1e-3

After implemented the SVR algorithm, we decided to implement an ensemble method. We evaluated four different ensemble machine learning algorithms, two boosting and two bagging methods.

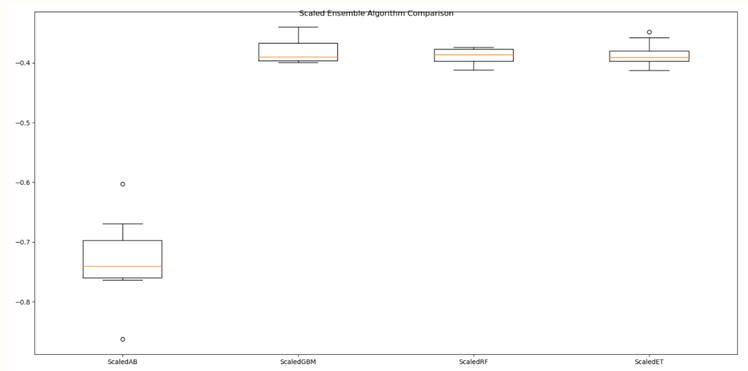

Figure 3: Differences in Ensemble Algorithm Performance on dataset

According to the statistics, the scores are better than the linear and nonlinear algorithms. We decided to tune the Gradiant Boosting algorithm since its MSE is the lowest, and also, the Extra Tree Regressor algorithm since its derivation is the lowest. As well as the SVR algorithm tuning, we use Grid Search method to obtain the following parameters.

Table 2: Ensemble Algorithm Settings

GBM		ET	
Parameter	Value	Parameter	Value
min samples leaf	1	learning rate	0.1
min samples split	6	max depth	7.0
n estimators	725	n estimators	600

Results

Table 3: Results from tuned algorithm.

Algorithm	Mean Squared Error
SVR	0.42640
Gradiant Boosting	0.33878
Extra Trees	0.38938

After having tuned our three algorithms, we calculated the final Mean Squared Error of each algorithm. We can see that the best score we have is from the Gradiant Boosting Algorithm. The mean squared error is 0.33878 meaning that in average, meaning that we have an error of 0.57 on the score out of 10, for any anime.

Amelioration

The current size of the dataset is really small, we hope to obtain the last 20,000 for the final report. Moreover, the features selected are based on numbers and identifiants without taking into account the title and the synopsis of the anime. A huge amelioration would be using the title and the synopsis as features, by analyzing the words containing.

Summary and conclusions

This project imagined by ourselves is a regression predictive modeling machine learning problem. By computing multiple linear, nonlinear and ensemble algorithms on a self-made dataset built from scratch, we finally predict with a error margin of 5.7% the score of a anime.

References

[1] *Rafael Antonio Pineda (2017, October). "Anime Industry Takes in Record 2.0 Trillion Yen in 2016" retrieved 4th june of 2018 from* https://www.animenewsnetwork.com/news/2017-10-25/anime-industry-takes-in-record-2.0-trillion-yen-in-2016/.123164

[2] *MyAnimeList website* https://myanimelist.net/