Abstract

As anime's global popularity surges, accurately predicting ratings is essential for improving recommendation systems, especially for newcomers. This project focuses on predicting anime ratings to deliver reliable recommendations, even for the ones that lack ratings, such as newly released, niche, or upcoming anime.

Using data with attributes like genre, type, episode count, and community engagement, the XGBoosting Regressor achieved a Root Mean Squared Error (RMSE) of 0.5, indicating an average prediction error of less than 0.5 rating points.

Data Preprocessing

There are 2 datasets.

First is the anime data. Each entry provides information on the anime's name, genre, type, number of episodes, number of community members, and rating.

anime_id	name	genre	type	episodes	rating	members
32281	Kimi no Na wa.	Drama, Romance, School, Supernatural	Movie	1	9.37	200630
5114	Fullmetal Alchemist: Brotherhood	Action, Adventure, Drama, Fantasy, Magic, Mili...	TV	64	9.26	793665
28977	Gintama°	Action, Comedy, Historical, Parody, Samurai, S...	TV	51	9.25	114262
9253	Steins;Gate	Sci-Fi, Thriller	TV	24	9.17	673572
9969	Gintama'	Action, Comedy, Historical, Parody, Samurai, S...	TV	51	9.16	151266

Second is the rating data, which provides information of each user's rating for each anime. Rating is -1 if the user watched it but didn't assign a rating.

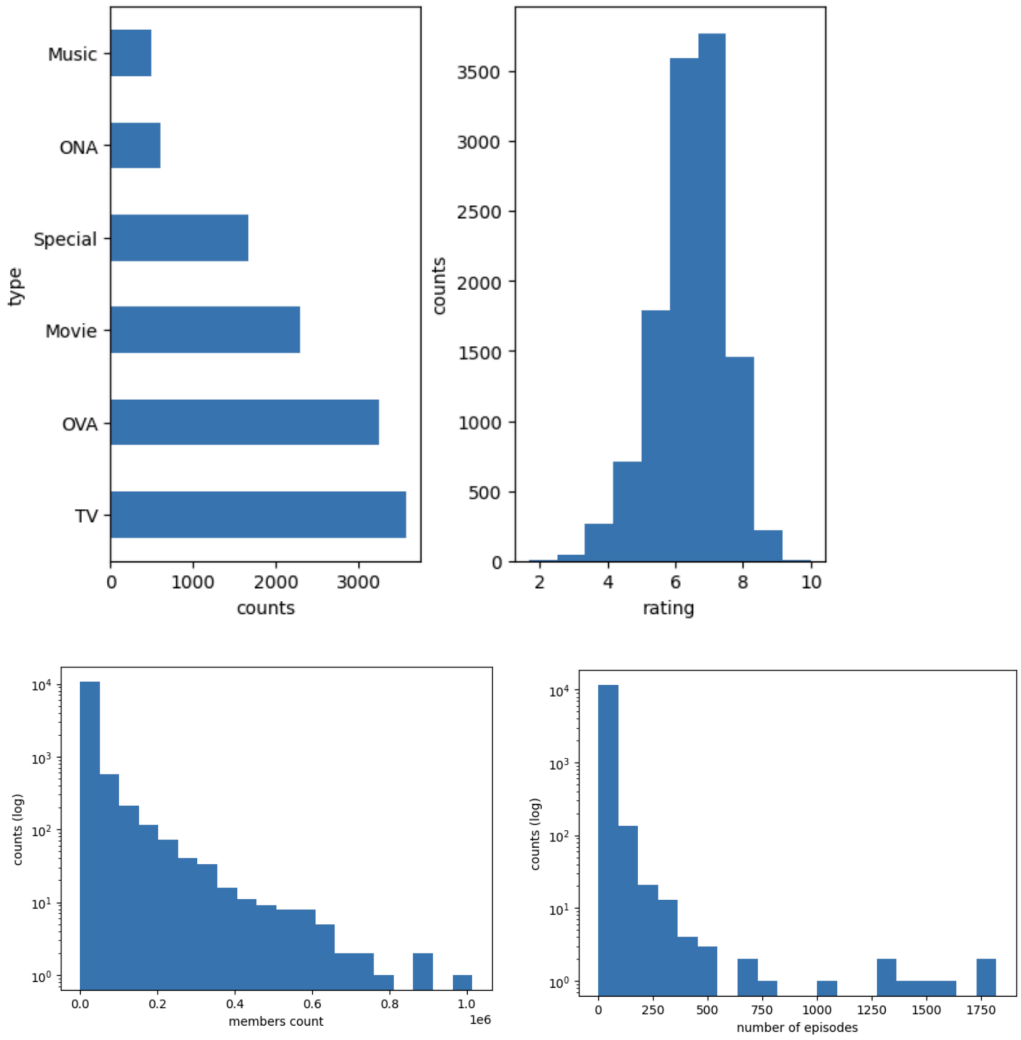
	user_id	anime_id	rating
0	1	20	-1
1	1	24	-1
2	1	79	-1
3	1	226	-1
4	1	241	-1

Data Wrangling

To ensure the dataset was clean and suitable for analysis, several preprocessing steps were taken:

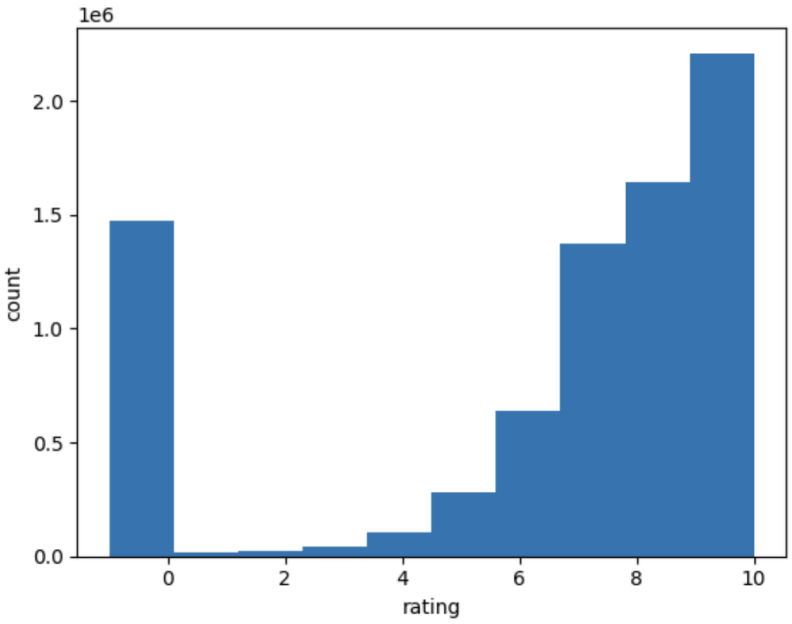
[anime_df]

- **Handling missing values:**
    a. **Ratings:** Dropped rows without ratings, as they are the target variable and are essential for the analysis.
    b. **Genre:** Filled missing genre values by referencing information from similar entries within the same series, identified through name similarities.
    c. **Number of episodes:** For all types except TV and ONA, null episode values were replaced with the mean episode count, as these types typically have episode numbers within 4 standard deviations.
    d. **Remaining missing values:** Dropped records with remaining missing values, which accounted for about 1% of the total dataset, to maintain data quality.
- **Data type conversion:** Converted features to their appropriate data types, such as converting object types to strings for names and categorical data for types.
- **Check for duplicates:** Verified that there were no duplicate records in the dataset for unique name and type pairs.
- **Genre splitting:** Since the genre column contains arrays of values, each unique genre was split into its own column with boolean values indicating the presence or absence of that genre for each anime.
- **Check for outliers:** The distributions of member count and episode count were left-skewed with a few large values. Spot checks confirmed the accuracy of these counts, as they correspond to popular and well-known anime titles.



[rating_df]

- **Check for missing values:** Confirmed that there are no missing values in the dataset.
- **Remove duplicates:** Removed duplicate records based on the `user_id` and `anime_id` pair to ensure unique user-anime interactions.
- **Check rating distributions:** Observed that most ratings are 6 or above. Some records had a rating of -1, indicating that the user watched the anime but did not provide a rating.



Add features to anime_df

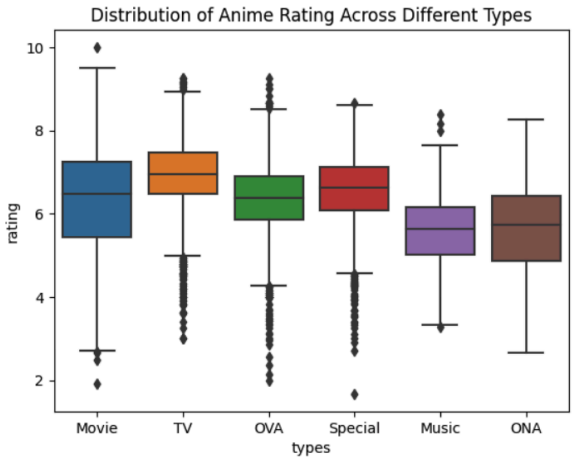
Added the following features to the anime_df for the next phase of analysis.

- number of reviews
- number of views without reviews

Data Exploratory Analysis

Rating by type

ANOVA results indicate that anime type has a significant impact on ratings. The low p-value suggests that the differences in mean ratings across different types are statistically significant. Additionally, the high F-value indicates that there is more variance between the types than within them.

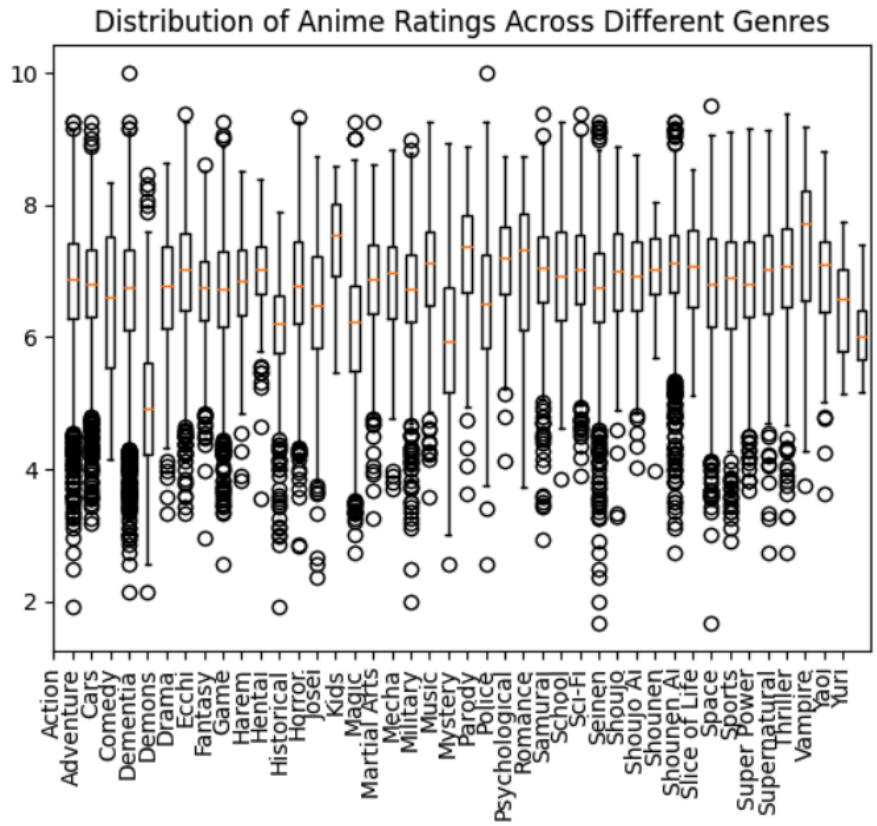



```
anova_results = pingouin.anova(data=anime_df,
                               dv="rating",
                               between="type")
anova_results
```

	Source	ddof1	ddof2	F	p-unc	np2
0	type	5	11863	362.475092	0.0	0.132528

Rating by genre

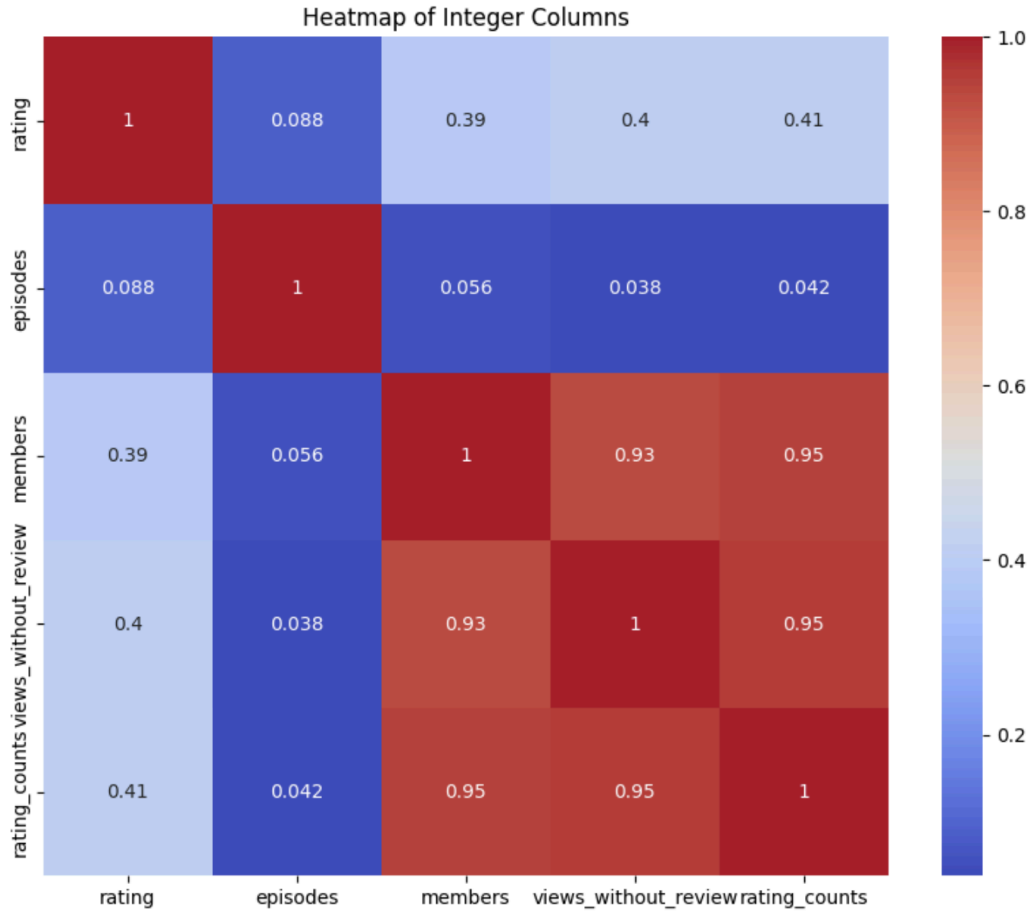
Similarly, ANOVA results show that anime genre significantly affects ratings. The low p-value confirms that the differences in mean ratings across genres are statistically significant, and the high F-value indicates greater variance between genres compared to within them.



	Source	ddof1	ddof2	F	p-unc	np2
0	genre	42	35263	90.641355	0.0	0.097439

Rating correlation with numerical features

The heatmap analysis shows that ratings are moderately correlated with member engagement metrics such as `members`, `views_without_review`, and `rating_counts`. However, the number of episodes has a weak correlation with ratings and other features, indicating that episode count does not play a significant role in determining anime ratings. The strong correlations among `members`, `views_without_review`, and `rating_counts` suggest that these metrics are driven by a common underlying factor, likely related to the overall popularity and visibility of the anime.

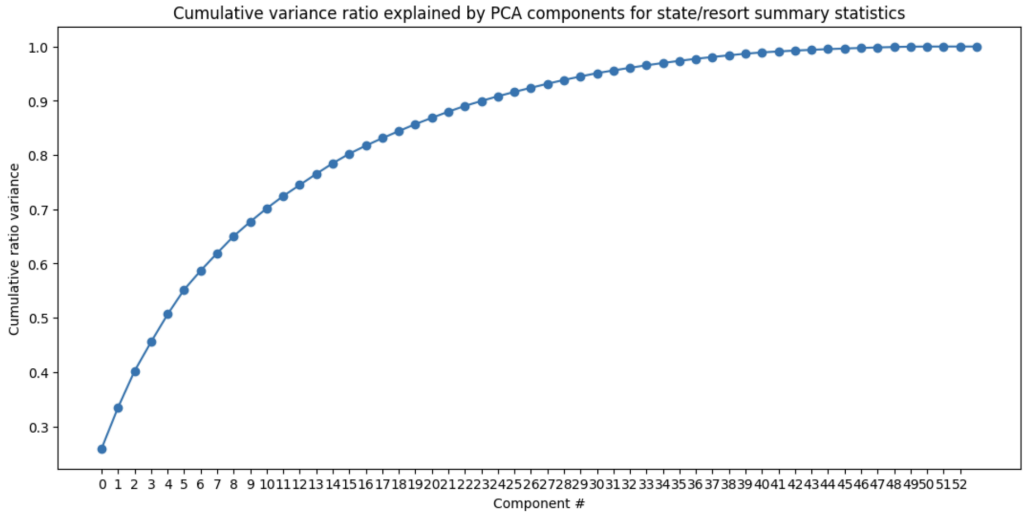


Feature engineering

In the process of preparing the dataset for analysis, several feature engineering steps were undertaken to ensure the model could effectively learn from the data:

1. **Categorical Feature Encoding:**
    - The `type` column, which represented different categories of anime (e.g., TV, Movie, ONA), was converted into dummy variables using one-hot encoding. This allowed the model to handle categorical data more effectively by creating a separate binary column for each category.
2. **Scaling Numerical Features:**
    - Numerical columns such as `episodes`, `members`, `views_without_review`, and `rating_counts` were normalized using MinMaxScaler. This scaling was necessary to ensure that these features, which have different ranges, contributed equally to the model during training.
3. **Feature Reduction (PCA):**

- Principal Component Analysis (PCA) was applied to the dataset to reduce dimensionality while retaining most of the variance. This step helped in simplifying the dataset and potentially improving model performance by focusing on the most important features.



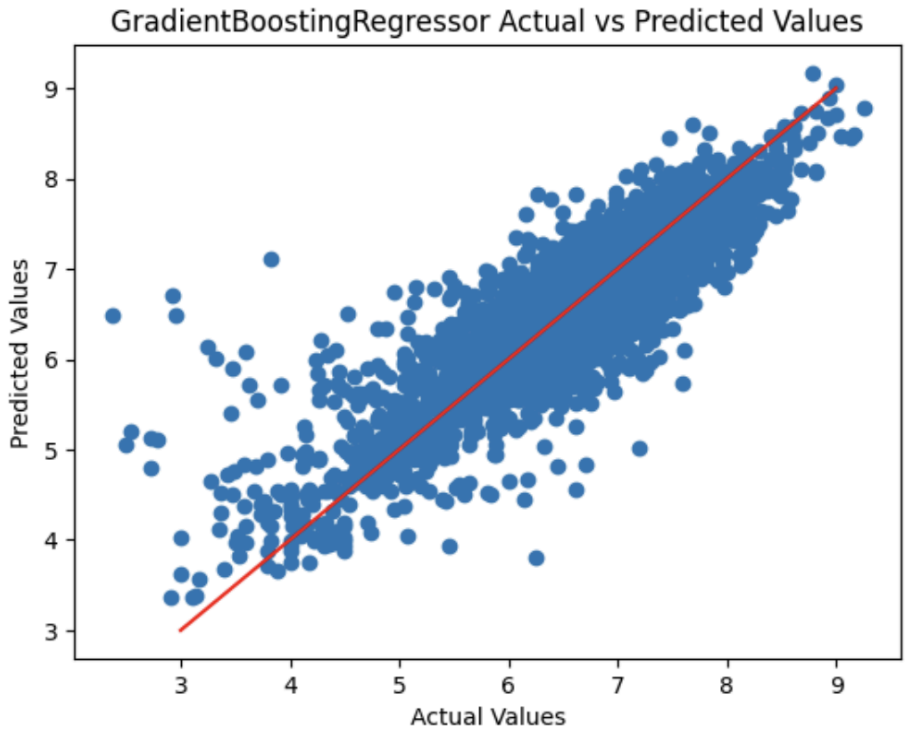
4. **Data Splitting:**
   - The dataset was split into training and testing sets, ensuring that the model could be evaluated on unseen data. A train-test split ratio is using the default of 75/25.

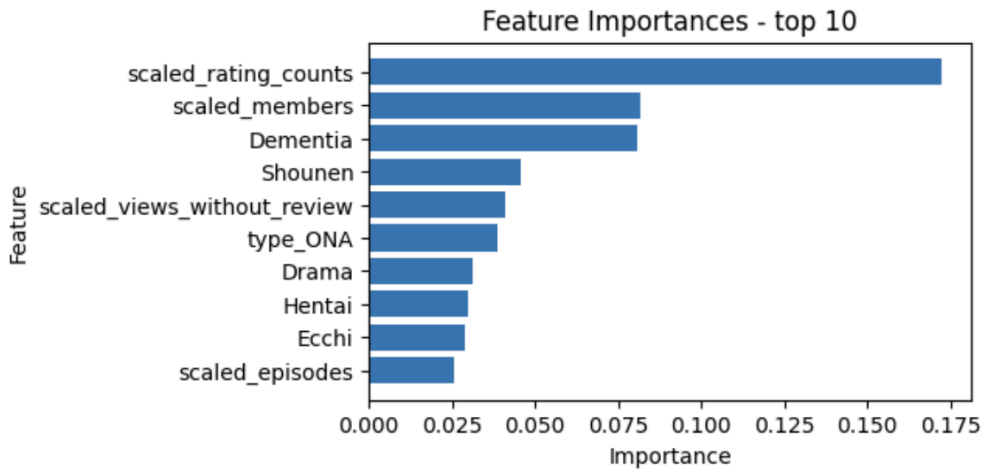
Model selection & performance

To identify the best model for predicting ratings, we evaluated several algorithms, comparing their performance based on metrics such as RMSE and R-squared. The initial results showed that the XGBRegressor outperformed the others, delivering the lowest RMSE and the highest R-squared.

	rmse	r2
LinearRegression	0.796414	0.366115
RidgeCV	0.796650	0.365740
GradientBoostingRegressor	0.547984	0.699899
RandomForestRegressor	0.518846	0.730965
XGBRegressor	0.508736	0.741347
SVR	0.750796	0.436654
KNeighborsRegressor	0.709074	0.497524

Building on this, further optimization was performed on the XGBRegressor. This included cross-validation and hyperparameter tuning, which led to an improvement in the model's performance, reducing the RMSE to 0.50 and increasing the R-squared to 0.74.





Next Steps

Model Enhancements

- **Experiment with Neural Networks:** Explore Neural Networks to potentially improve prediction accuracy by capturing more complex patterns in the data.
- **Advanced Feature Engineering:** Introduce interaction terms and integrate external data sources to enhance the model's ability to identify nuanced relationships.

Recommendation System Development

- **Leverage the Rating Predictor:** Utilize the current rating predictor to recommend highly rated anime to newcomers.
- **Personalized Recommendations**: Develop a user-based collaborative filtering system or content-based filtering to deliver personalized anime recommendations for existing users.