COMP4434 Big Data Analytics
Individual Project Mid Report

Ruixiang JIANG
19079662D

March 11, 2021

Contents

1 Introduction

This is the midterm report for the individual project. In this report, we are going to briefly present the works we have done previously, as well as indicate our plan for later works. This report is organized like a data science competition pipeline.

2 Exploratory Data Analysis (EDA)

Exploratory Data Analysis is usually the first step in a data science competition. The motivation of it is to get a quick overview of the dataset. In this section, we will briefly present some of the result of EDAs. For more details, please refer to final report and codes.

2.1 Distribution Visualization

Notice: some outliers in Fig.1 are temporarily removed for better display quality. In fig.2, the whole dataset are plotted pairwisely, the diagonal are same as fig.1, except that outliers are kept. It could be observed that there are outliers in some features. The "genre" feature are intentionally ignored at this stage. The more details could be found in section 3.2.

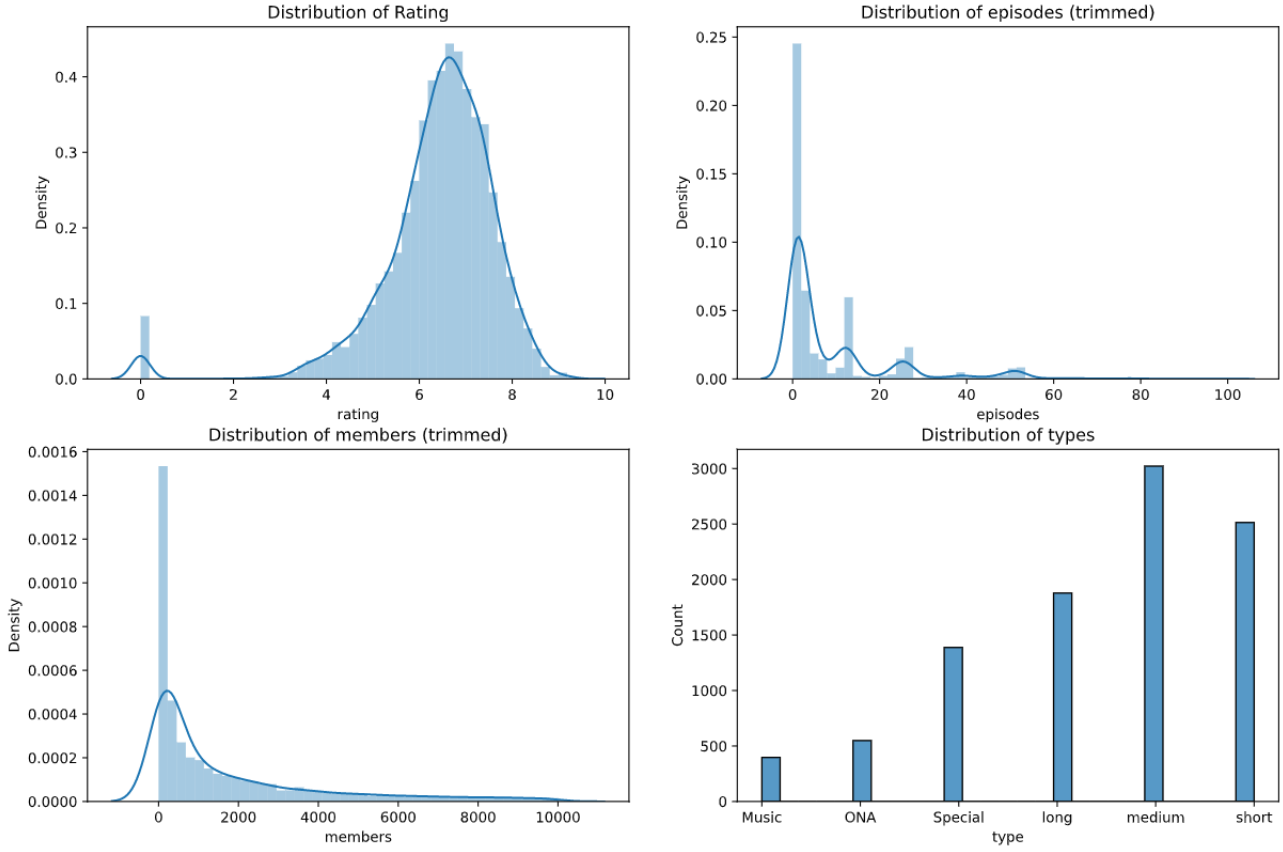

Figure 1: Distribution of some input features

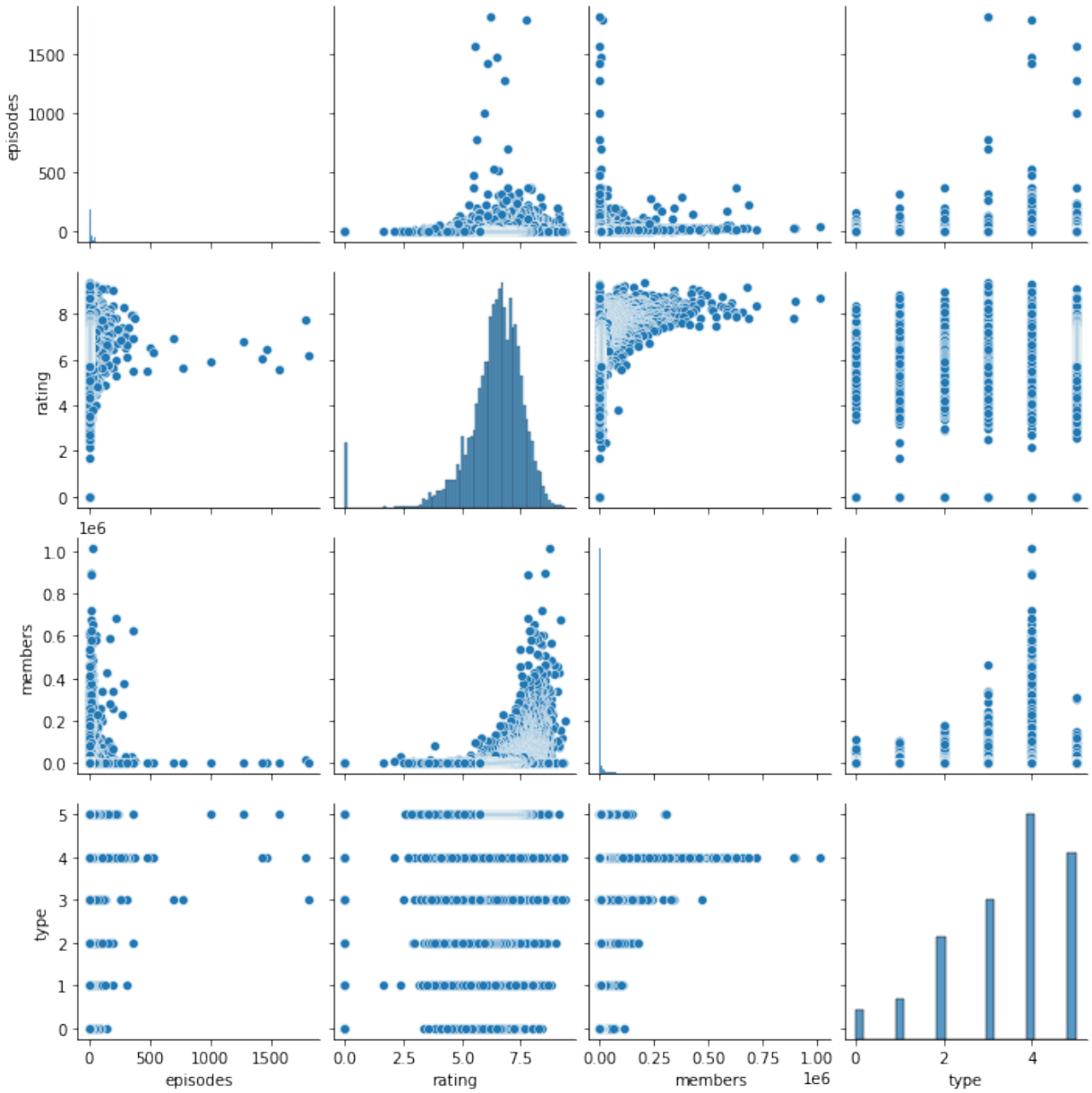

Figure 2: Pairwise graph

2.2 Correlation Analysis

Correlation analysis checks the relation between features. Our analysis suggests that "members" and "rating" show a relatively strong correlation in this dataset.

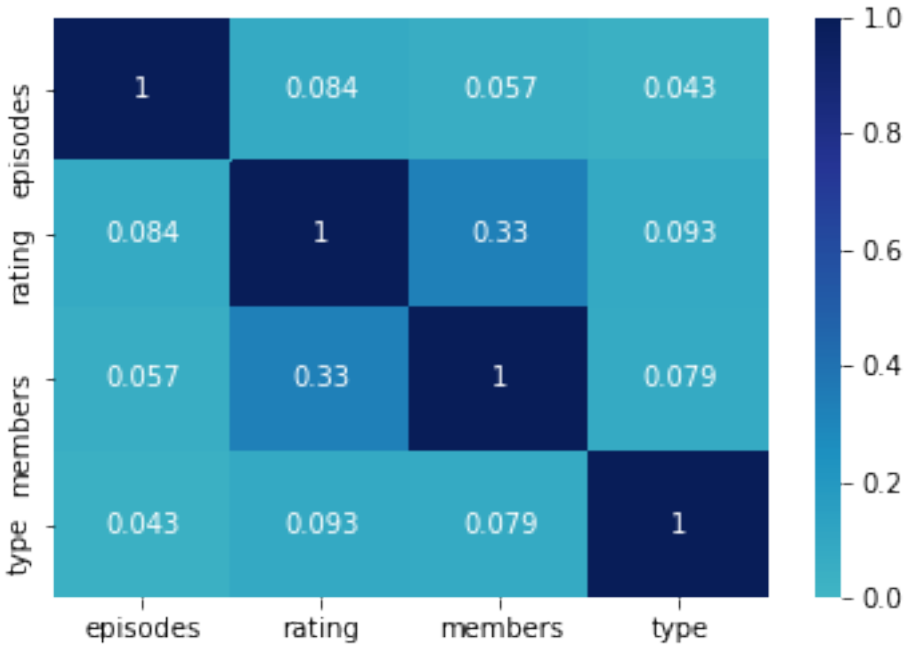

Figure 3: Correlations between some features

3 Preprocessing and Feature Engineering

3.1 Data Cleansing

For Null values, we just set the null feature as 0 at current stage. We might use more specific method to deal with null values.

For outliers, we adopted capping method. Large values are capped to a small value. Our experiment shows that such simple technique doubles the random forest model performance.

3.2 Feature Importance

The feature importance are derived base on Random Forest. The result of it might suggest possible dimensionality reduction. Again, the "genre" feature are intentionally omitted. As fig.4 shows, the importance of "type" are low, suggesting that its contribution to the regression target (i.e. rating) might be lower than other features. This may due to our improper way of encoding, we will check this later. The importance of "members" are at a relatively high level, which conforms with our previous correlation analysis in section 2.2.

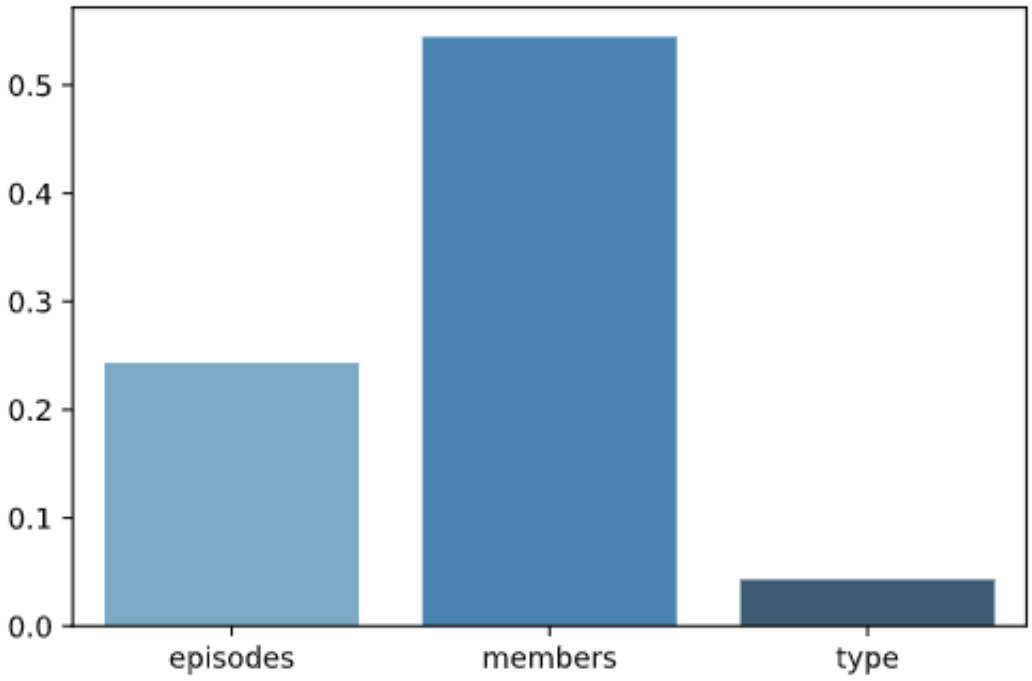

Figure 4: Feature importances

3.3 Feature Encoding

Some of the features in this dataset are not numerical. To run model on it, we need to encode them as numerical values. In this section we briefly present our encoding strategy.

For "genre" feature, it is a multi-valued feature. We convert it into a vector. More specifically, we use "multi-hot encoding", that is to say, a vector with many '1's and '0's. The reason behind is that the genre is usually not exclusive. We do not use ordinal encoding because the distance between any pair of genre is undefined. However, this encoding technique may cause

	episodes	members	type	Kids	Seinen	Shounen Ai	Demons	Horror
4212	26.0	561.0	4.0	0.0	0.0	0.0	0.0	0.0
4798	1.0	1216.0	2.0	0.0	0.0	0.0	0.0	0.0
1843	1.0	22537.0	3.0	0.0	0.0	0.0	0.0	0.0
811	1.0	3309.0	3.0	1.0	0.0	0.0	0.0	0.0
2771	1.0	6812.0	2.0	0.0	0.0	0.0	0.0	0.0
...	...	...	...	...	...	...	...	...
7641	1.0	88.0	5.0	1.0	0.0	0.0	0.0	0.0

Figure 5: Typical data points after the encoding

"the curse of dimensionality", since the encoded feature will be a sparse and high-dimension matrix. We may check this later.

For "type" feature, we adopt ordinal encoding. The labels are encoded as scalars. Our intuition is that there shows a loosely order relationship between types ("long" > "medium" > "short")

Fig.5 exemplifies typical data points makeup.

4 Baseline Models

In this section, we present the machine learning models adopted for baseline regression. We give a high-level architectural description, while left the implementation details unspecified.

4.1 Models Desing

For baseline purpose, linear regression (LR), random forest (RF), and multi-layer perceptron (MLP) are used.

For LR, we used plain linear regression, and linear regression with polynomial expansion (expanded to degree = 2)

For RF, we used 100 estimators, each tree has unlimited depth (i.e could expand as long as it is impure).

For MLP, we used a 2 hidden-layer fully connected network. Hidden layer contains 300, 200 neurons, each with ReLu nonlinearity. Each hidden layer adopted L2 regularization, and between them are batch normalization layers. Such normalization is needed, since at this stage we did not perform feature scaling or normalization.

At current stage, we simply left all models untuned.

4.2 Models Evaluation

Models are evaluated using root mean square error (RMSE), with 5-fold cross validation. The result are presented in fig.6. Notice that the y axis is RMSE, so lower score means better performance.

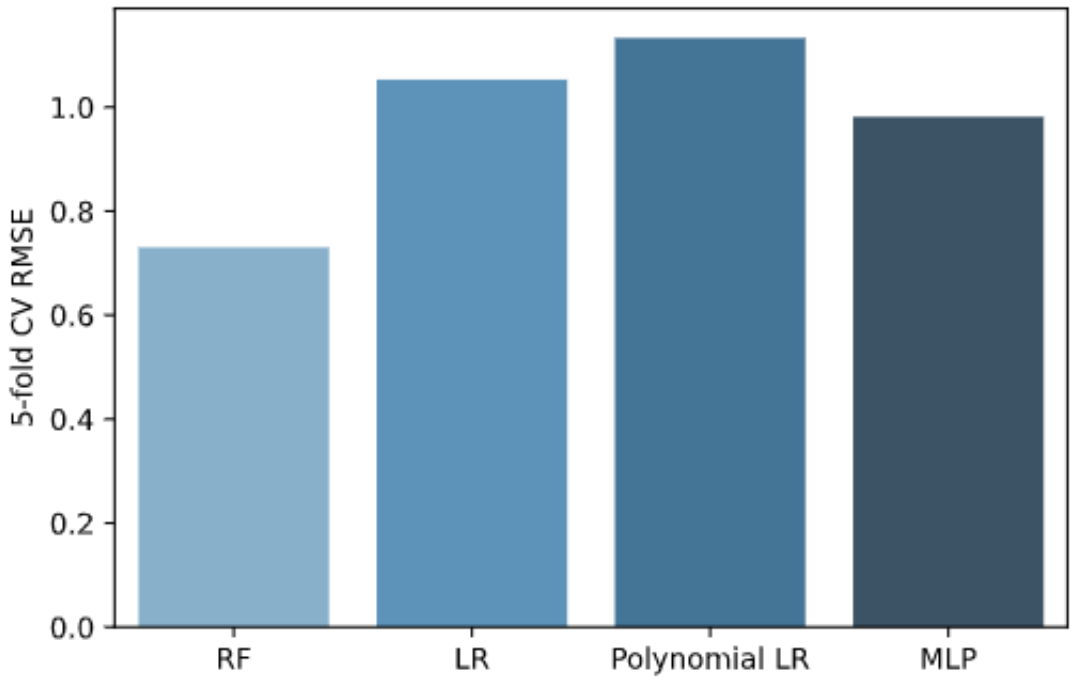

Figure 6: 5-fold cross validation RMSE score

Among all baselines, RF seems to outperform all other models. However this is just prelim-

inary observation, as models are untuned (for example, the learning curve of MLP in fig.7 suggests possible underfit).

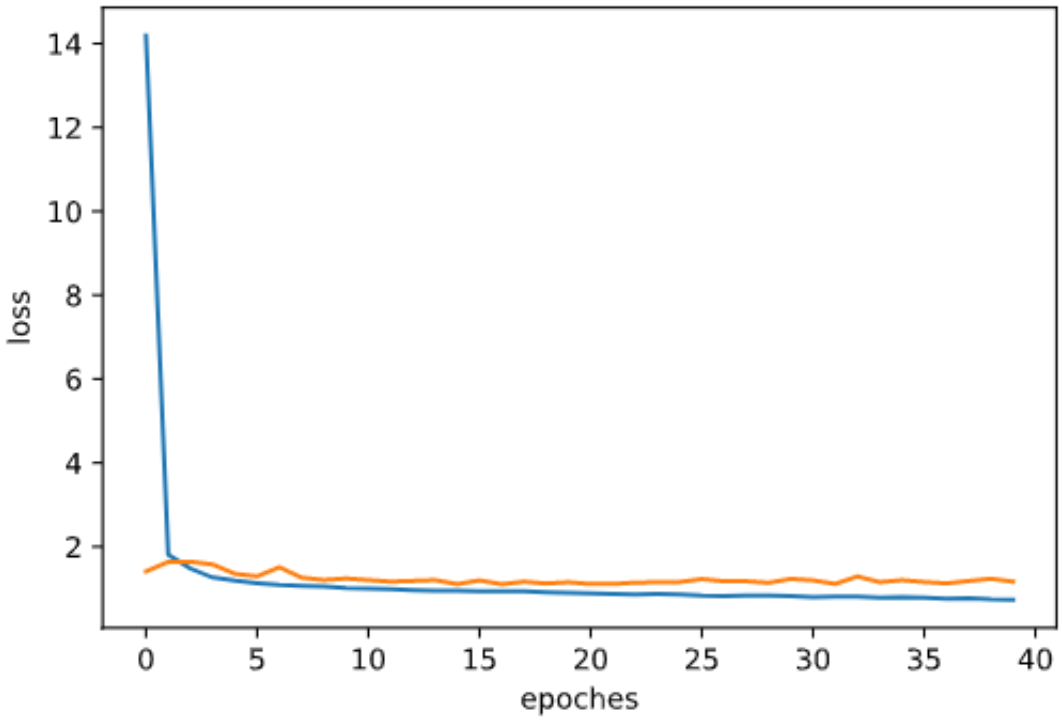

Figure 7: Training Curve of MLP (blue: train, orange: validation)

Moreover, polynomial LR performed worse than LR, the reason behind may be the way that we encode the "genre" feature (i.e multi-hot). As a result of such encoding, a polynomial expansion over all features will produce 1128 features, most of which are meaningless. Further work shall be done to restrict the features used in polynomial expansion.

5 Further Works

In this section we state our plan for further works. As for data processing, normalization and feature scaling is needed. We may also try to do more feature engineering, and next we may fine-tune each base models, and try a wide spectrum of models such as xgboosting. We also plan to do a model stacking (ensembling) over all fine-tuned base models to get a better performance. Beyond this, clear investigation shall be taken to avoid overfitting problem.