Music Recommendation System

Yuwei Tu Yiran Xu Weisen Zhao
New York University

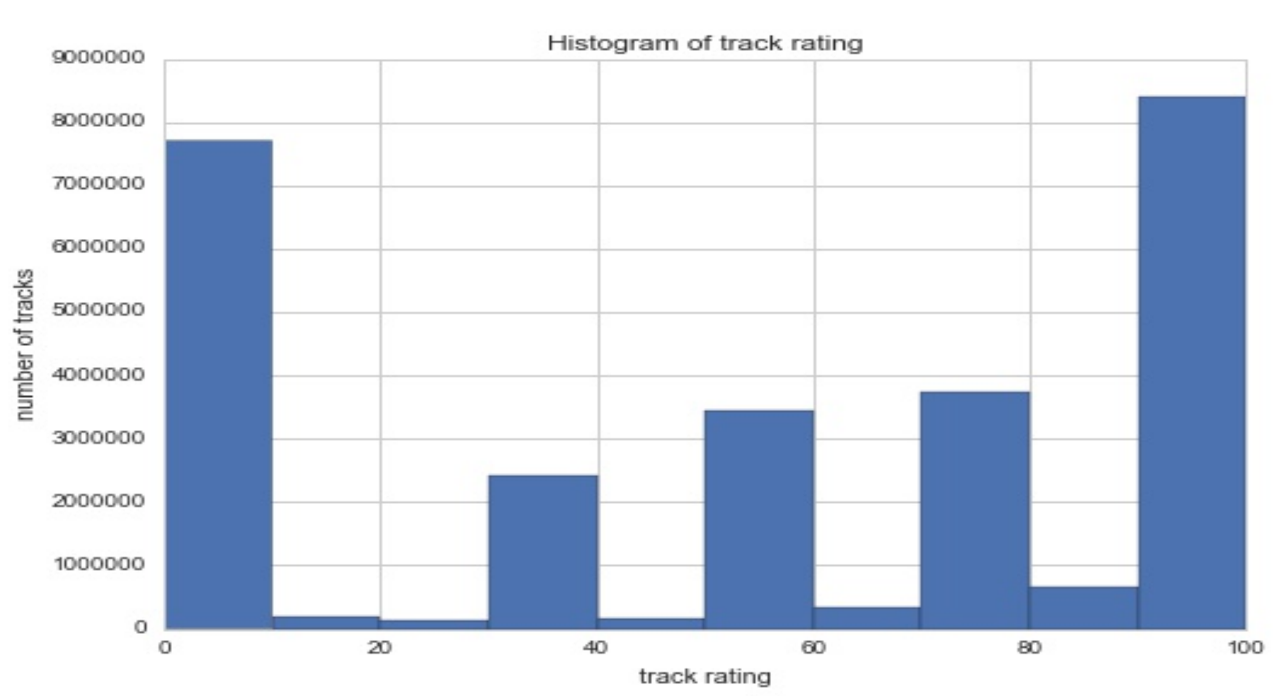
♦ Abstract

We build a Music recommendation system based on the predicted score of rating on a specific song for a user. First, we predict the rating of user's rating on a specific song. Then we rank songs for each user based on the first step's predicted rating.

♦ Data

Raw Data Summary
We use Yahoo! Music rating data by Yahoo Research Lab. There are five data files in the raw data: rating data, track data, album data, artist data and genre data. The rating data lists the ratings of users grouped by user ID. For each rating of a user, the data consists of item ID, rating(0 - 100) and time stamp. In summary, the raw data have 1,000,990 users, 624,961 items and 252,800,275 ratings in total.

Data Grouping
We divide the data into 5 different groups by every 20 rating interval. We find that our dataset is unbalanced. The number of data samples in those intervals: 20-40, 40-60 and 60-80 are significantly less than the number of data samples in rest of intervals.

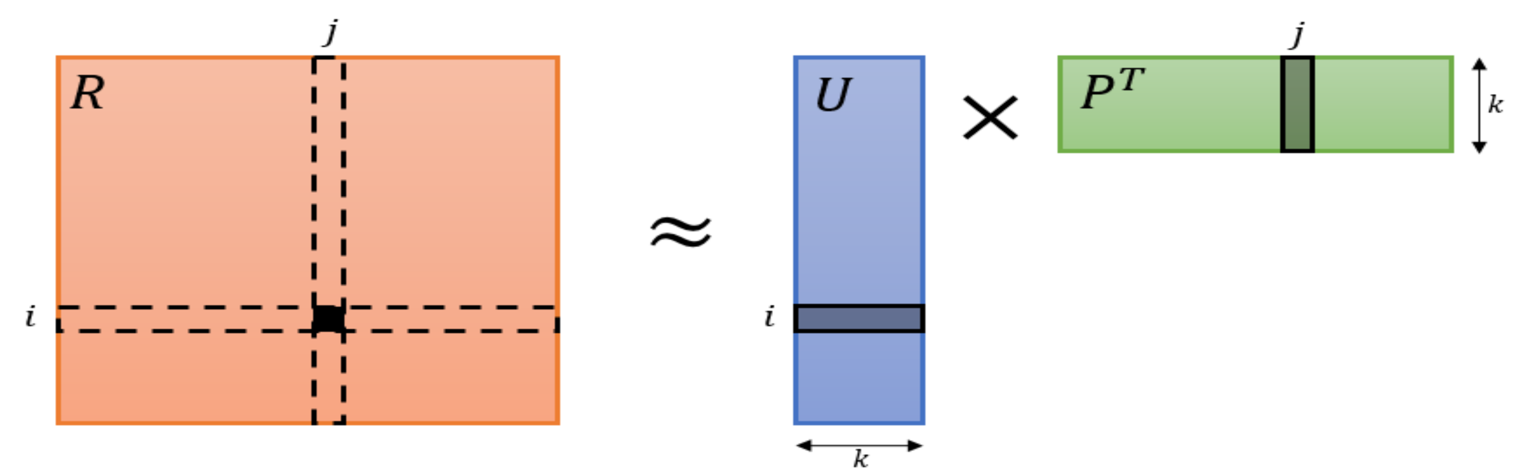


Data Preprocessing
In order to predict the rating of each song with corresponding user, we generate a matrix that consists of rating, items corresponding to the rating and users by combining the rating data file and track data file in the raw data.

	userID	itemID	rating
0	172493	100010	100
1	174695	100010	90
2	174994	100010	30
3	175009	100010	70
4	177123	100010	100

♦ Features Engineering

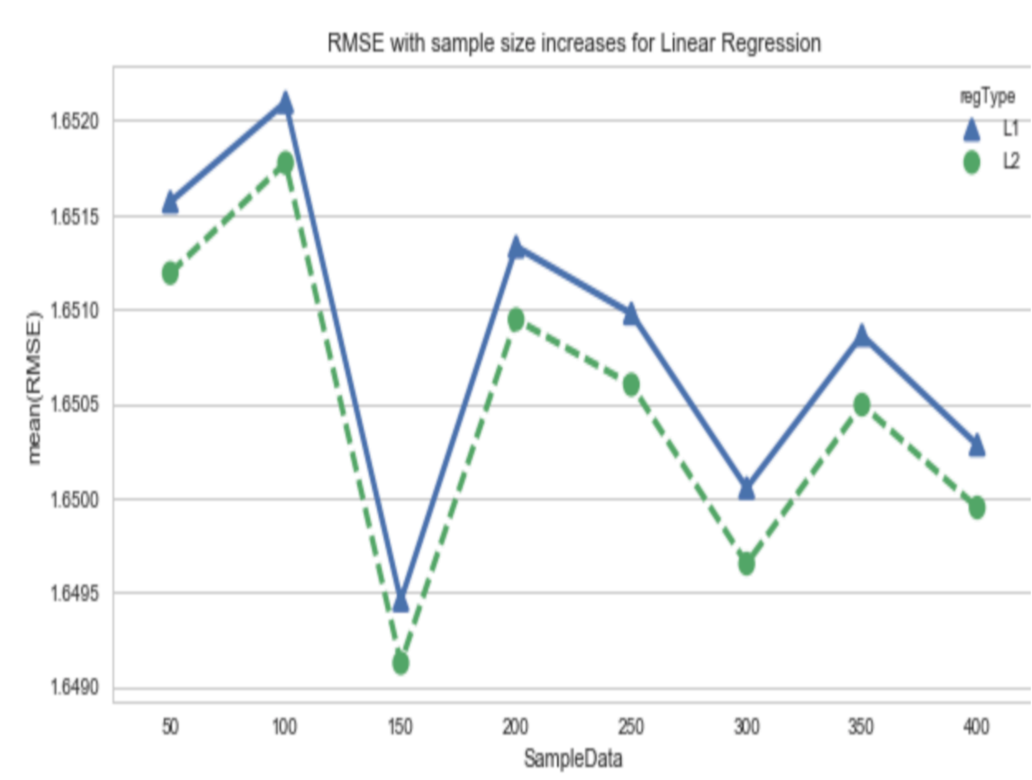
Alternating Least Squares
ALS rotates between fixing U and P. When U is fixed, the system computes P by solving a least-squares problem per item, and vice versa. Applying this technique on input matrix R will end up factorizing into two matrices U and P such that R ≈ U × P
We use ALS factorization algorithm to generate two different feature matrix: item-based and user-based. Each matrix has 50 features.



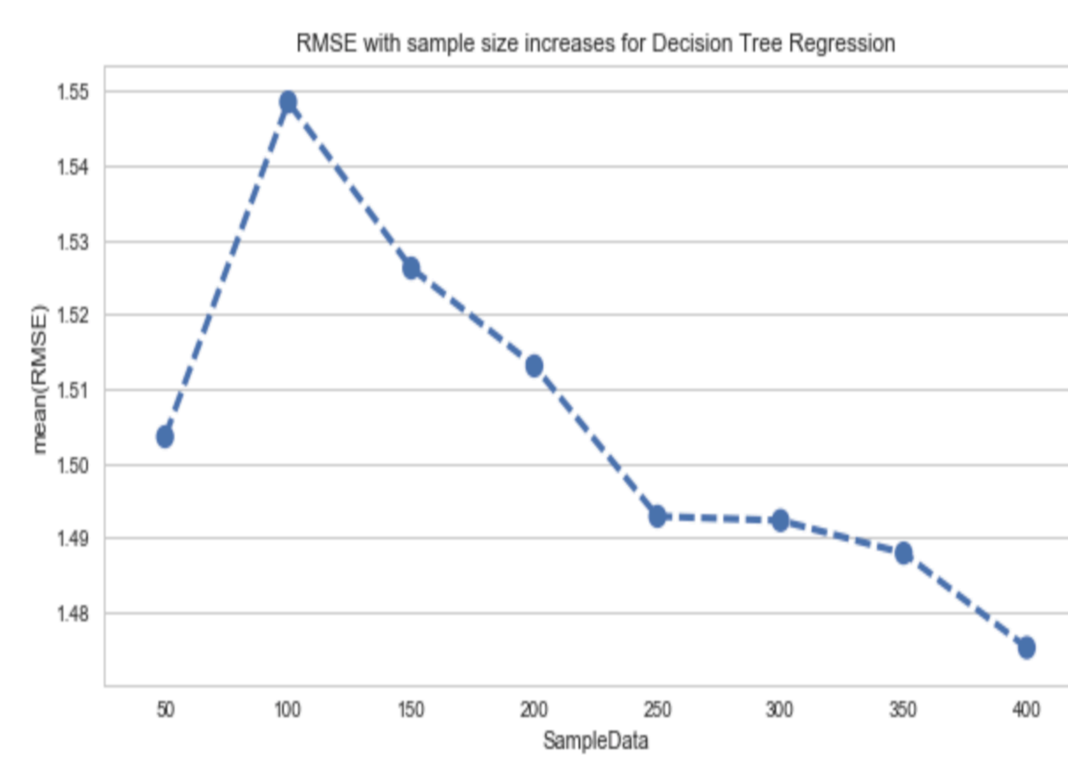
♦ Methods and Models

We try different regression methods in order to have better result for prediction. Those methods are Linear Regression, Decision Tree Regression, Gradient Boosted Regression and XGBoost Regression.

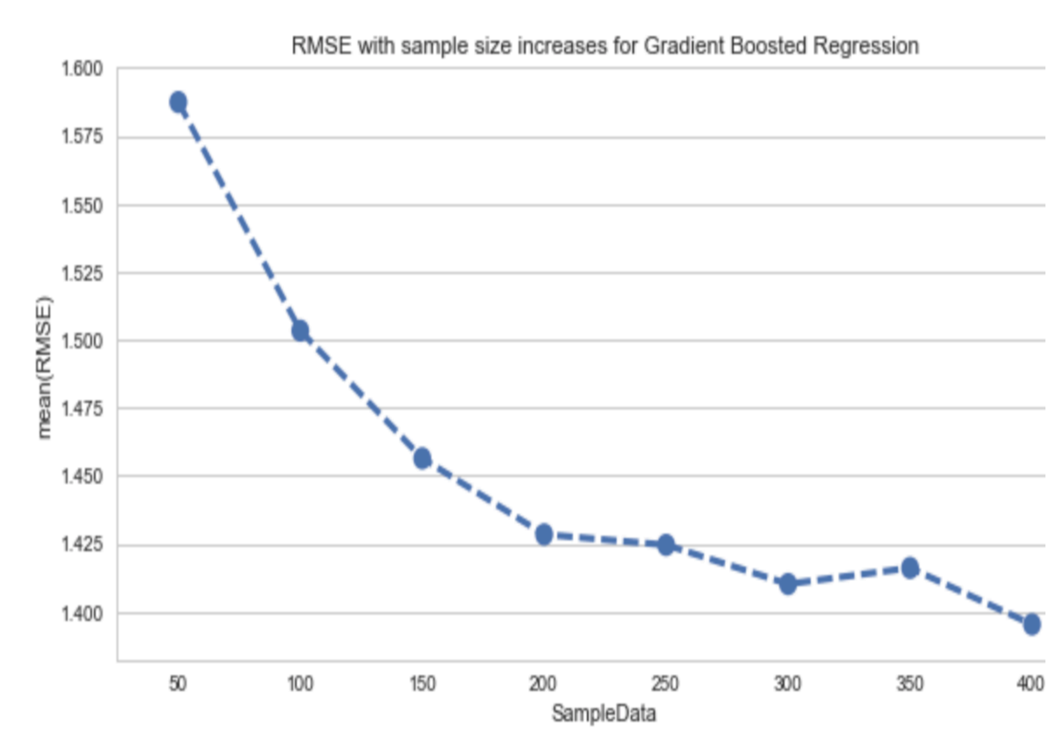
Linear Regression



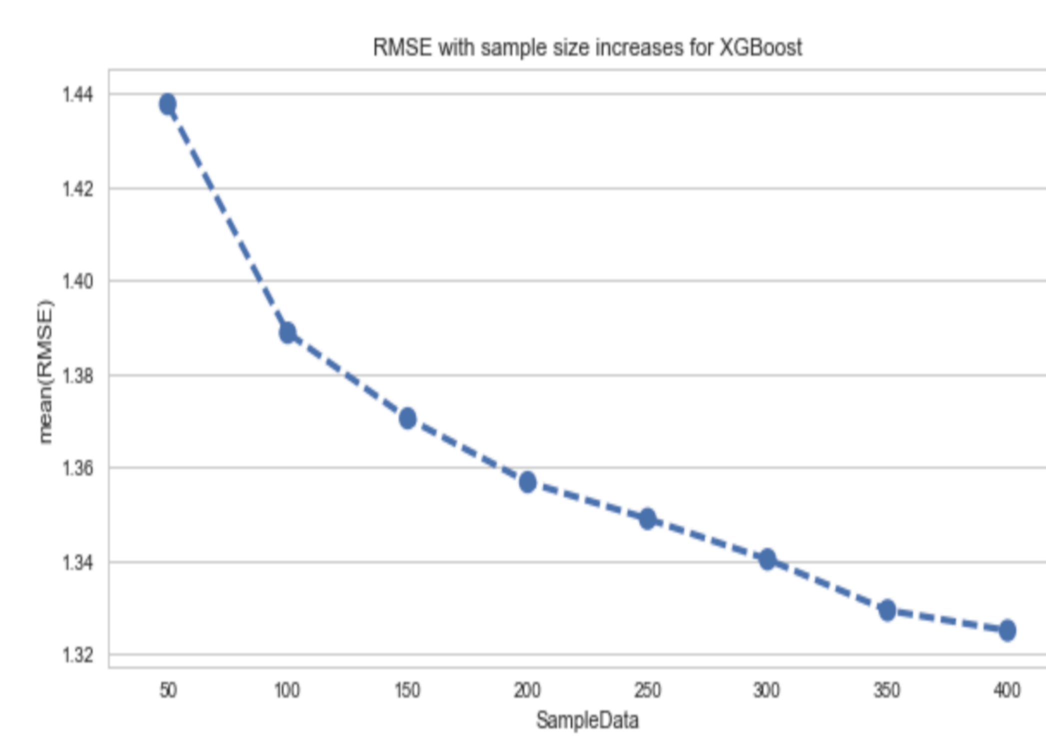
Decision Tree Regression



Gradient Boosted Regression

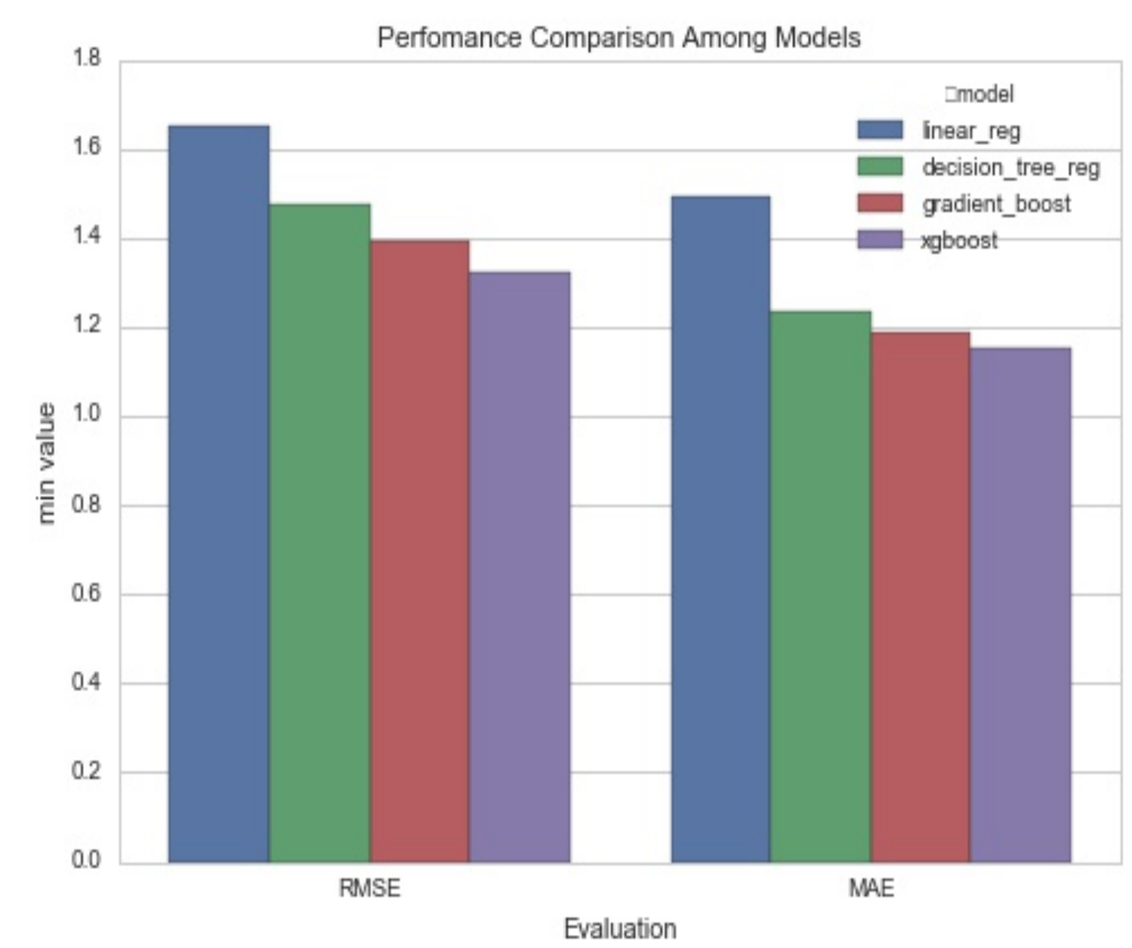


XGBoost Regression



♦ Evaluation

The Root Mean Square Error is a frequently used measure between values predicted by a model. And Mean Absolute Error is one of most effective ways to compare forecasts and eventual outcomes. To that point, we use RMSE and MAE to evaluate the performance of different methods that we propose. As a benchmark, we run Alternating Least Squares method and obtain 1.13 for RMSE with 2 millions input samples. We also obtain RMSE and MAE for following models.



♦ Discussion

There are some aspects that we could improve in the future. First of all, we can increase the sample numbers for training dataset. Because of the hardware limitation, we can only run 4 millions at this moment. We clearly find that Root Mean Square Error decreases as we increase the size of input samples. Besides that, we plan to adopt down sampling the data and split into 5 groups by every 20 interval or stratify in order to solve the unbalanced data distribution problem. After that, we could train 5 different models based on 5 split groups and average the results.

♦ References

Beel, Joeran, Stefan Langer, Marcel Genzmehr, Bela Gipp, Corinna Breitinger, "Research Paper Recommender System Evaluation."
Proceedings of the International Workshop on Reproducibility and Replication in Recommender Systems Evaluation – RepSys '13(2013): n. pag. Web.