Introduction

In this assignment, the goal was to build a recommendation system to propose movies to users. The only dataset I used was MovieLens-100k

Data analysis

From the data analysis one may conclude that the ratings means and stds, conditioned on user occupation or gender or on movie genres, tend to uniform distribution, thus niether of these splits tend to give (or get) ratings, different from the population's ones.

The dataset is biased towards male users and towards student users. The most common genres are drama, comedy, thriller, action, and romance.

Model Implementation

The task can be viewed as a regression problem: given user features (such as age, occupation, and gender) together with movie features (release date and multilabel genres), regress these features on the rating. For the regressors I have tried random forest and gradient boosting.

Model Advantages and Disadvantages

One of the pros of the selected model is it being fast and lightweight (one can also parallelize estimators of GB and then reduce their predictions to get the model output)

Another advantage of the model is its robustness to newly registered users: cold start is not an issue here as no similarity between users is utilized during inference

This non-utilization of group information, however, is also the main disadvantage of simple regression approach: it falls short of collaborative filtering with SVD decomposition if compared by RMSE loss.

Training Process

The boosting model was trained on 500 estimators, and from the evaluation on the validation set only first 101 estimators have shown to be useful. Each estimator has depth=8. Most important features are movie's release date, user's age and occupation.

Evaluation

For evaluation I have chosen to set loss (RMSE) as metrics, and MAP@10, 20, 50, 100

Results

Random recommendation baseline has RMSE = 1.829235598912978 and MAP@20 = 0.003225276340355705, "top 100 popular movies" baseline has RMSE=1.4808899951232035 and MAP@20 = 0.06474643578856318

Catboost regression [1] has achieved RMSE = 1.0603873418433587 (whereas collaborative filtering from surprise [2] has converged to RMSE = 0.9365) MAP results are not extremely satisfactory:

MAP @ 10 for GB model = 0.013480367585630745

MAP @ 20 for GB model = 0.015426614900136167

MAP @ 50 for GB model = 0.016964324005412

MAP @100 for GB model = 0.018811129566940216

The random baseline is beaten and top 100 baseline is beaten in RMSE, but CF outperforms the regression on ratings.

Therefore, the basic CF approach is a better solution, even despite not utilizing additional features. Regression approach could be used as a baseline to outperform or checkpoint to initialize from.

References

[1] https://catboost.ai/

[2] https://github.com/NicolasHug/Surprise