Report MovieLens Project

jmeydam

2019-06-03

Introduction

This project replicates some basic but vital results of the Netflix challenge (https://en.wikipedia.org/wiki/Netflix_Prize):

> The Netflix Prize was an open competition for the best collaborative filtering algorithm to predict user ratings for films, based on previous ratings without any other information about the users or films, i.e. without the users or the films being identified except by numbers assigned for the contest.

The goal is to minimize the root mean square error (RMSE) of the predicted ratings. A RMSE of close to one means, roughly, that a typical prediction is off by one star. During the Netflix challenge it became clear that improvements beyond a RMSE of around 0.857 are virtually impossible, given the data.

The original Netflix dataset is no longer available. Instead, the MovieLens 10M dataset (https://grouplens.org/datasets/movielens/10m/) is used. Some key facts:

- 10 million ratings
- range ratings: between 0.5 and 5 stars
- 10,000 movies
- 72,000 users
- Released 1/2009

The model is very basic, relying only on the overall mean, the regularized movie effect and the regularized user effect.

This approach and various properties of the data were covered in class and are documented here:

> https://rafalab.github.io/dsbook/large-datasets.html#recommendation-systems

Much of the code was already provided by Prof. Irizarry and is used here with minor modifications.

Method

The approach is described in a blog post by Edwin Chen (http://blog.echen.me/2011/10/24/winning-the-netflix-prize-a-summary/):

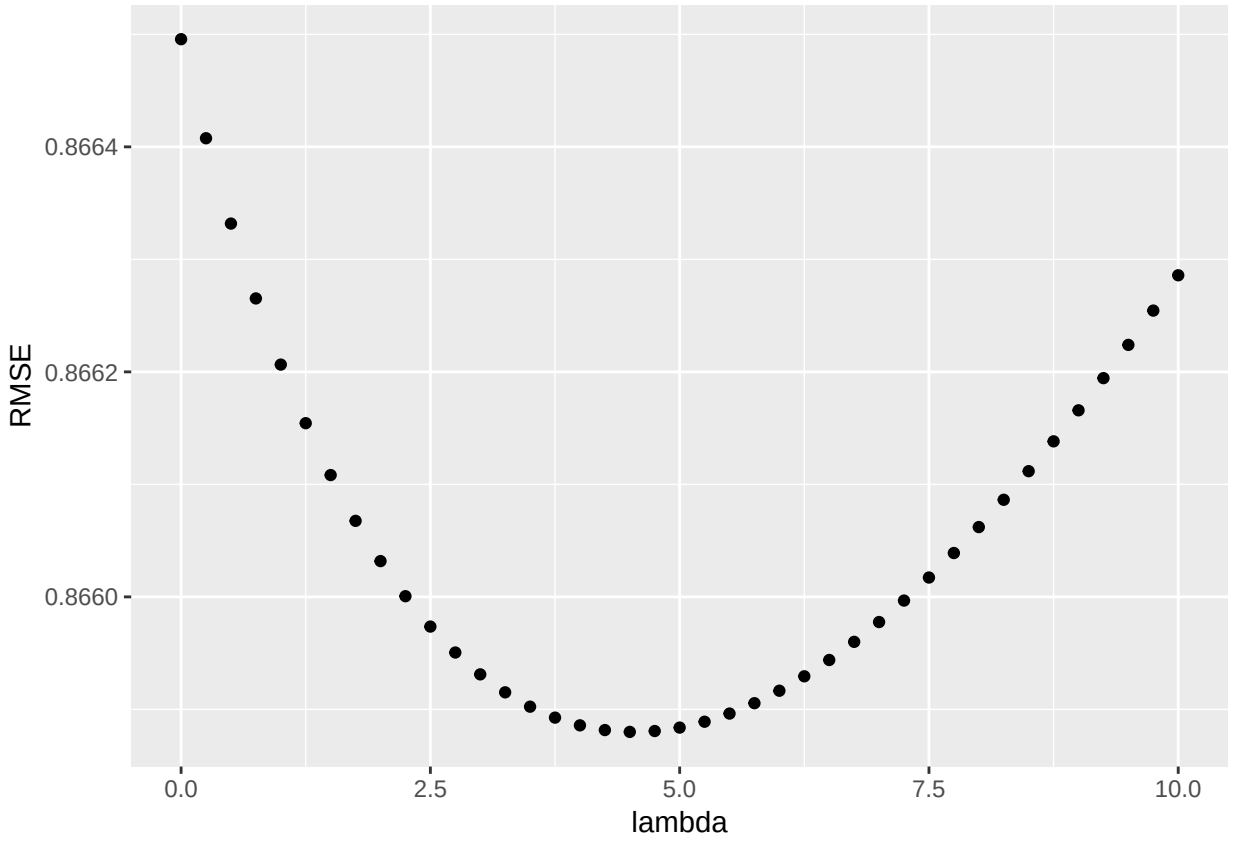
> Suppose Alice rates Inception 4 stars. We can think of this rating as composed of several parts:
>
> - A **baseline rating** (e.g., maybe the mean over all user-movie ratings is 3.1 stars).
> - An **Alice-specific effect** (e.g., maybe Alice tends to rate movies lower than the average user, so her ratings are -0.5 stars lower than we normally expect).
> - An **Inception-specific effect** (e.g., Inception is a pretty awesome movie, so its ratings are 0.7 stars higher than we normally expect).
>
> [...]
>
> And, in fact, modeling these biases turned out to be fairly important: in their paper describing their final solution to the Netflix Prize, Bell and Koren write that
>
> > Of the numerous new algorithmic contributions, I would like to highlight one – those humble baseline predictors (or biases), which capture main effects in the data. While

> the literature mostly concentrates on the more sophisticated algorithmic aspects, we have learned that an accurate treatment of main effects is probably at least as significant as coming up with modeling breakthroughs.

In this project, the edx dataset is split into a training set and a test set, and the parameter lambda is determined using the test set. The model (with this parameter lambda) is then trained on the complete edx dataset before being evaluated using the validation set. This is the key difference compared to what was covered in the course material.

Results

RMSE (test set) for lambda between 0 and 10



lambda minimizing RMSE (test set)

```
lambda
```

```
## [1] 4.5
```

RMSE for validation set

```
rmse_submission
```

```
## [1] 0.8648242
```

Conclusion

A basic model relying only on the overall mean, the regularized movie effect, and the regularized user effect comes very close to the best possible result.

Further improvements would not only require substantially more effort, but also more computing resources, in particular enough memory for operations on large matrices (even if sparse matrices are used).