Computational Intelligence Laboratory Project: Collaborative Filtering

Dina Zverinski*, Jan Wilken Dörrie† and Álvaro Marco Añó‡
Group: TheNonstandardDeviations
Department of Computer Science, ETH Zurich, Switzerland
*Email: *zdina@student.ethz.ch, †dojan@student.ethz.ch, ‡malvaro@student.ethz.ch*

***Abstract*—Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.**

I. Introduction

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

II. Models and Methods

A. Dataset

The dataset which was used throughout the development of our algorithm was made available on the project course webpage [1]. It contains the ratings of $10\,000$ users on 1000 movies on a scale between 1 and 5. Naturally, not every user rated every movie, in fact only $1\ 388\ 107$ of the $10\ 000\ 000$ possible ratings were present leading to a data sparsity of 86%.

B. Evaluation of Results

In order to train and test the developed algorithm the present ratings were randomly split in two disjoint sets of equal size. The first set then formed the training set, i.e. data that was trained on, and the second set was used for testing the algorithm. The chosen metric is “Root Mean Squared Error” (RMSE) which is defined in the following way [2]–[4]:

$$\sqrt{\sum_{(u,i)\in \text{TestSet}} \frac{\left(r_{ui} - \hat{r}_{ui}\right)^2}{|\text{TestSet}|}} \tag{1}$$

Here r_{ui} denotes the real rating while $\hat{r}_{ui}$ is the approximation through the algorithm. With ratings restricted to the interval $[1, 5]$ valid RMSE values range between 0 and 4, with lower scores being better. In addition, the raw CPU time was evaluated for the project grade. This favors solutions that are efficient with regard to time.

C. Data Imputation and Baseline Estimators

Given that standard Collaborative Filtering algorithms such as K-Means and Singular Value Decomposition (SVD) require dense instead of sparse matrices different strategies exist to fill in the missing values. The most simple one is to replace all missing values with a constant zero value. Accuracy can be improved by considering other constants such as the global mean μ of all existing entries. More advanced techniques try to approximate a given missing value for a user u and item i by taking a combination the global mean and the specific user and item means b_u and b_i [2]–[5]:

$$b_{ui} = \mu + b_u + b_i. \tag{2}$$

b_u and b_i can be directly computed from the data, however it is advisable to introduce regularize terms to overcome over-fitting when only very few ratings are available. This leads to the following equations, where $R(u)$ and $R(i)$ denote the sets of all ratings by user u and item i respectively [2]–[5]:

$$b_i = \frac{\sum_{u\in R(i)}(r_{ui} - \mu)}{\lambda_i + |R(i)|} \qquad b_u = \frac{\sum_{i\in R(u)}(r_{ui} - \mu - b_i)}{\lambda_u + |R(u)|} \tag{3}$$

λ_i and λ_u are regularize parameters that should be found via cross-validation. Finally it is also possible to estimate b_u and b_i by solving the regularized least squares problem [2]–[5]

$$\min_{b_*} \sum_{(u,i)\in\mathcal{K}} (r_{ui} - \mu - b_u - b_i)^2 + \lambda \left(\sum_u b_u^2 + \sum_i b_i^2\right). \tag{4}$$

Possible solution approaches include Alternating Least Squares (ALS) or Stochastic Gradient Descent (SGD). The baseline estimators are able to capture a lot of signal present in the data so that it is advisable to make this preprocessing step part of every more involved algorithm. For example, the RMSE score for a simple SVD solution improved drastically when missing data was imputed with optimized baseline estimators.

D. Factorized Models

Motivated by the progress a simple SVD approach could make we investigated more sophisticated approaches.

1) Regularized SVD: Shortly after the Netflix Challenge started in 2006 Simon Funk proposed the idea of a "regularized SVD" [6]. In contrast to an ordinary Singular Value Decomposition this approach does not rely on the imputation of missing values, but only considers actual present ratings for training. The algorithm tries to find two matrices P and Q that accurately represent user-item interactions. Both users and items get transformed to the same latent factor space of a fixed dimension where their dot product is taken to measure their compatibility. A rating r_{ui} is then approximated by $\hat{r}_{ui} = b_{ui} + q_i^T p_u$. In order to avoid over-fitting to the data a regularize term is added that penalizes large magnitudes of p_u and q_i. The associated least squares problem is the following:

$$\min_{q_*, p_*} \sum_{(u,i) \in \mathcal{K}} \left(r_{ui} - b_{ui} + q_i^T p_u\right)^2 + \lambda(\|q_i\| + \|p_u\|) \quad (5)$$

This problem again can be solved using either ALS or SGD. Here we assume that the biases have been estimated in a preprocessing step, however it is also possible to learn them at the same time as P and Q.

III. Results

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

IV. Discussion

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

V. Summary

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

References

[1] Computational Intelligence Lab. (2015). Collaborative Filtering, [Online]. Available: http://cil.inf.ethz.ch/applications/collaborative_filtering (visited on 06/16/2015).

[2] Y. Koren, "Factorization Meets the Neighborhood: A Multifaceted Collaborative Filtering Model," in *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '08, Las Vegas, Nevada, USA: ACM, 2008, pp. 426–434, ISBN: 978-1-60558-193-4. DOI: 10.1145/1401890.1401944.

[3] ——, "Factor in the Neighbors: Scalable and Accurate Collaborative Filtering," *ACM Trans. Knowl. Discov. Data*, vol. 4, no. 1, pp. 1–24, Jan. 2010, ISSN: 1556-4681. DOI: 10.1145/1644873.1644874.

[4] Y. Koren and R. Bell, "Advances in Collaborative Filtering," English, in *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds., Springer US, 2011, pp. 145–186, ISBN: 978-0-387-85819-7. DOI: 10.1007/978-0-387-85820-3_5.

[5] Y. Koren, R. Bell, and C. Volinsky, "Matrix Factorization Techniques for Recommender Systems," *Computer*, vol. 42, no. 8, pp. 30–37, Aug. 2009, ISSN: 0018-9162. DOI: 10.1109/MC.2009.263.

[6] S. Funk. (Dec. 2006). Netflix Update: Try this at Home, [Online]. Available: http://sifter.org/~simon/journal/20061211.html (visited on 06/16/2015).