

TC1 Project: Know your customers

Aris Tritas

Laurent Cetinsoy

November 18, 2016

Abstract

The objective of this analysis is to predict a rating that would be assigned by customers to items (which could be e.g. a firm's services). The problem was designed as a case of sequential prediction. Moreover, it can be viewed as a collaborating filtering. It is solved using the following methods: firstly, an Non-negative Matrix Factorization is applied. Then, and autoencoders.

1 Introduction

In collaborative filtering, each item i of a collection is rated by one or more members of a population n_u . The aim is to predict the rating that a person would give to an unseen item, so as to be able to recommend new items to users such that it fits their preferences with high probability.

In this setup, let R represent a $n \times m$ matrix. The rating of the j^{th} item from the i^{th} user is r_{ij} .

1.1 Data

We are given a training set describing the ratings assigned by customers to items in some given order. Dataset size: 2536705 examples in the training set, 1306637 in the test set. Total number of users 93705 (of which only 92088 are in the training set) and 3561 items. The dataset is very sparse, less than 1% of the ratings matrix is filled. Ratings distribution: [114214, 251596, 728518, 860463, 581913] In proportion of the total number of ratings: [0.0450, 0.0991, 0.2872, 0.3392, 0.2293].

1.2 Hidden Markov Model

It is possible to model the problem of sequential prediction using a hidden Markov model. In order to do that efficiently, it is assumed that

1.3 Non negative matrix factorization

In non negative factorizations we seek SPD matrices U and V such that the rating product matrix can be factorized as $R = UV^T$. It is possible to find such matrices that minimize the reconstruction error defined below by using an iterative optimization method (e.g gradient descent):

$$\begin{aligned} \arg \min_{U,V} & \frac{1}{2} \|R - UV\|^2 \\ & + \alpha \lambda_1 (\|U\|_1 + \|V\|_1) \\ & + \frac{1}{2} \alpha \lambda_2 (\|U\|^2 + \|V\|^2) \end{aligned} \quad (1)$$

The reconstruction error is computed over all non negative coefficients of the R matrix. In other words, the algorithm only tries to recover the available ratings.

1.4 ALS-WR

The Alternating Least Squares with Regularization algorithm [2] minimizes (??), which is non-convex as both U and V are unknown. It does so by alternatively updating U while keeping V fixed and then updating V while keeping U fixed. The parameters used were $\lambda = 0.01$ and a latent dimension $d = 100$.

1.5 Auto-encoder

The auto-encoder is a neural network approach used in unsupervised learning for dimensionality reduction. It optimizes the following loss function:

$$L_{2,\alpha,\beta} = \alpha \left(\sum_{j \in \mathcal{C}(\tilde{x})} [nn(\tilde{x}_j) - x_j]^2 \right) + \beta \left(\sum_{j \ni \mathcal{C}(\tilde{x})} [nn(\tilde{x}_j) - x_j]^2 \right) +$$

. where The first term is a loss over corrupted version of the inputs which the network seeks to denoise. The second term is the reconstruction loss.

In the supervised classification task, the auto-encoder's objective is to lower a prediction error over a training set.

The approach proposed by [1] was used to tackle our collaborating filter problem. Two different auto-encoders can be trained depending on whether the network seeks to make user or item vectors dense:

- U-CFN is defined as $\hat{\mathbf{u}}_i = nn(\mathbf{u}_i)$
- V-CFN is defined as $\hat{\mathbf{v}}_i = nn(\mathbf{v}_i)$

2 Temporal dynamics

One idea to add temporal dynamics is to bias either of the matrix factors.

3 Results

The metrics used to evaluate the performance of each training method for this type of problem is either the RMSE (root mean squared error) or the MAE (mean absolute error). The scores below represent the RMSE and were computed on a validation set selected uniformly at random amongst the training set:

| NMF | ALS-WR | User AE | Item AE |
|------|--------|---------|---------|
| 3.70 | 0.934 | 0.913 | 0.938 |

4 Conclusion

Different methods were compared in order to solve the problem of customer affinity prediction. When sparsity becomes an issue, no single method works best: exploiting the problem structure and combining multiple approaches has the potential of achieving the best results.

References

- [1] Florian Strub, Jeremie Mary, and Romaric Gaudel. Hybrid collaborative filtering with autoencoders. *arXiv preprint arXiv:1603.00806*, 2016.
- [2] Yunhong Zhou, Dennis Wilkinson, Robert Schreiber, and Rong Pan. Large-scale parallel collaborative filtering for the netflix prize. In *International Conference on Algorithmic Applications in Management*, pages 337–348. Springer, 2008.