华南理工大学

South China University of Technology

The Experiment Report of *Machine Learning*

SCHOOL: SCHOOL OF SOFTWARE ENGINEERING

SUBJECT: SOFTWARE ENGINEERING

Author:
Anran Lin
Mei Zhang
Haolin Pan

Supervisor:
Qingyao Wu

Student ID:
201630665045
201630666325
201630665441

Grade:
Undergraduate

January 8, 2019

Recommender System Based on Matrix Decomposition

***Abstract*—The experiment is a approach of building a recommender system based on matrix decomposition.The main purpose of the experiment is to explore the construction of recommended system as well as to understand the principle of matrix decomposition.**

I. Introduction

NOWADAYS,there are far too much information on the Web.It is of a great importance to have an algorithm to recommend things that we may get interested to us automatically.As a result,there are plenty of available recommendation algorithms.And in this experiment,we choose matrix decomposition approach to help to construct the recommender system.

II. Methods and Theory

A. Basic Idea

Given that each users have rated some items in the system, we would like to predict how the users would rate the items that they have not yet rated, such that we can make recommendations to the users.
The task of predicting the missing ratings can be considered as filling in the blanks (the hyphens in the matrix) such that the values would be consistent with the existing ratings in the matrix.

B. matrix decomposition

The intuition behind using matrix factorization to solve this problem is that there should be some latent features that determine how a user rates an item.
In trying to discover the different features, we also make the assumption that the number of features would be smaller than the number of users and the number of items.Also,set the blank in R zero and initial the blank in $\hat{R}$ as zero. The normal form of matrix decomposition is :

$$R_{m\times n} \approx P_{m\times k} \times Q_{k\times n} = \hat{R}_{m\times n} \tag{1}$$

In this way,each row of P would represent the strength of the associations between a user and the feature.Similarly,each row of Q would represent the strength of the associations between an item and the features.

C. Using matrix decomposition to predict

According to equation(1),the problem is changed to find every item of matrix $P_{m\times k}$ and $Q_{k\times n}$.
Define the loss function:

$$e_{i,j}^2 = (r_{i,j} - \hat{r}_{i,j})^2 = (r_{i,j} - \sum_{k=1}^{K} p_{i,k}q_{k,j})^2 \tag{2}$$

Eventually,the aim is to find the minimal loss:

$$\min loss = \sum_{r_{i,j}\neq 0} e_{i,j}^2 \tag{3}$$

D. minimize the loss function using stochastic gradient decent

Calculate the negative gradient:

$$\frac{\partial}{\partial p_{i,k}} e_{i,j}^2 = -2(r_{i,j} - \sum_{k=1}^{K} p_{i,k}q_{k,j})q_{k,j} = -2e_{i,j}q_{k,j} \tag{4}$$

$$\frac{\partial}{\partial q_{k,j}} e_{i,j}^2 = -2(r_{i,j} - \sum_{k=1}^{K} p_{i,k}q_{k,j})p_{i,k} = -2e_{i,j}p_{i,k} \tag{5}$$

Update P and Q:

$$p_{i,k} = p_{i,k} - \alpha\frac{\partial}{\partial p_{i,k}} e_{i,j}^2 = p_{i,k} + 2\alpha e_{i,j}q_{k,j} \tag{6}$$

$$q_{k,j} = q_{k,j} - \alpha\frac{\partial}{\partial q_{k,j}} e_{i,j}^2 = q_{k,j} + 2\alpha e_{i,j}p_{i,k} \tag{7}$$

where the α is the learning rate.

E. SGD with Regularization

A common extension to this basic algorithm is to introduce regularization to avoid overfitting. This is done by adding a parameter β and modify the squared error as follows:

$$e_{i,j}^2 = (r_{i,j} - \sum_{k=1}^{K} p_{i,k}q_{k,j})^2 + \frac{\beta}{2}\sum_{k=1}^{K}(||P||^2 + ||Q||^2) \tag{8}$$

The new parameter β is used to control the magnitudes of the user-feature and item-feature vectors such that P and Q would give a good approximation of R without having to contain large numbers.
The update rules:

$$p_{i,k} = p_{i,k} + \alpha\frac{\partial}{\partial p_{i,k}} e_{i,j}^2 = p_{i,k} + \alpha(2e_{i,j}q_{k,j} - \beta p_{i,k}) \tag{9}$$

$$q_{k,j} = q_{k,j} + \alpha\frac{\partial}{\partial q_{k,j}} e_{i,j}^2 = q_{k,j} + \alpha(2e_{i,j}p_{i,k} - \beta q_{k,j}) \tag{10}$$

F. minimize the loss function using ALS

The loss function:

$$L(P,Q) = \sum_{i,j}[(r_{i,j} - p_i^T q_j)^2 + \lambda(|p_i|^2 + |q_j|^2)] \tag{11}$$

where λ is the regularization coefficient. Calculate the partial derivative $\frac{\partial L}{\partial pi}$ obtain the answer and update P:

$$p_i = (Q^T Q + \lambda I)^{-1} Q^T ri \tag{12}$$

Calculate the partial derivative $\frac{\partial L}{\partial qj}$ obtain the answer and update Q:

$$q_j = (P^T P + \lambda I)^{-1} P^T rj \tag{13}$$

until it have reached convergence or reached the maximum iteration number.

III. Experiments

A. Dataset

The experiment is using MovieLens-100k dataset.
u.data – Consisting 10,000 comments from 943 users out of 1682 movies. At least, each user comment 20 videos. Users and movies are numbered consecutively from number 1 respectively. The data is sorted randomly.
u1.base / u1.test are train set and validation set respectively, seperated from dataset u.data with proportion of 80% and 20%. It also make sense to train set and validation set from u1.base / u1.test to u5.base / u5.test.

B. Implementation

1) minimize the loss function using SGD:

a) initialization and parameters: Read the data set and divide it.Build the original scoring matrix against the raw data, and fill 0 for null values.Then initialize the user factor matrix $P_{n\times k}$ and the item,film,factor matrix $Q_{k\times m}$,where K is the number of potential features.Set K to 20.Set the Iteration times to 400.

b) process: According to equation(8),calculate the loss.Set the learning rate α to 0.0002 and $\beta = 0.02$.
Use the stochastic gradient descent method to decompose the sparse user score matrix, get the user factor matrix and item (movie) factor matrix:
1. Select a sample from scoring matrix randomly;
2. Calculate this sample's loss gradient of specific row(column) of user factor matrix and item factor matrix;
3. Use SGD to update the specific row(column) of P and Q;
4. Calculate the Loss on the validation set, comparing with the Loss of the previous iteration to determine if it has converged. Repeat the 4 steps above for several times and get a satisfactory user factor matrix P as well as the item factor matrix Q. Draw a Loss curve of validation set.

c) result: It is obvious that the final score prediction matrix $\hat{R}$ is obtained by multiplying the user factor matrix P and the transpose of the item factor matrix Q.
And the result is shown in Fig.1.

2) minimize the loss function using ALS:

a) initialization and parameters: Read the data set and divide it.Build the original scoring matrix against the raw data, and fill 0 for null values.Then initialize the user factor matrix $P_{n\times k}$ and the item,film,factor matrix $Q_{k\times m}$,where K is the number of potential features.Set K to 40.Set the Iteration times to 100.

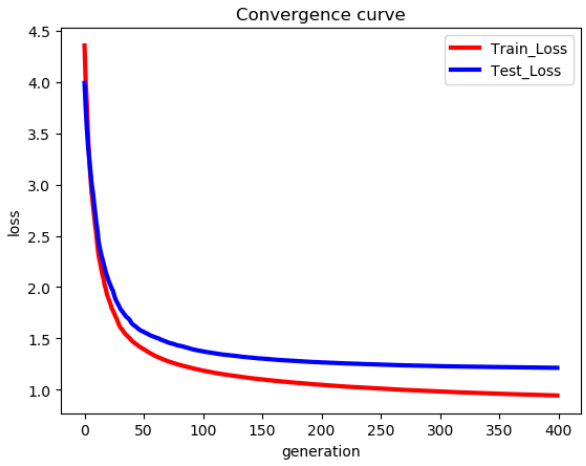

Fig. 1. The loss of matrix decomposition using SGD

b) process: According to equation(8),calculate the loss.Set the learning rate λ to 0.1. Use the ALS method to decompose the sparse user score matrix, get the user factor matrix and item (movie) factor matrix:
1. With fixd item factor matrix, find the loss partial derivative of each row (column) of the user factor matrices, ask the partial derivative to be zero and update the user factor matrices.
2. With fixd user factor matrix, find the loss partial derivative of each row (column) of the item factor matrices, ask the partial derivative to be zero and update the item
3. Calculate the loss on the validation set, comparing with the loss of the previous iteration to determine if it has converged. Repeat the 3 steps above for several times and get a satisfactory user factor matrix P as well as the item factor matrix Q. Draw a Loss curve of validation set.

c) result: It is obvious that the final score prediction matrix $\hat{R}$ is obtained by multiplying the user factor matrix P and the transpose of the item factor matrix Q.
And the result is shown in Fig.2,Fig.3 and Fig.4.

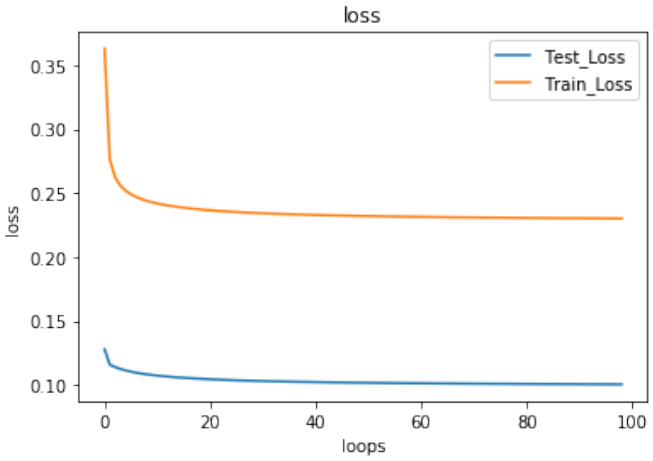

Fig. 2. The loss of the complete dataset u

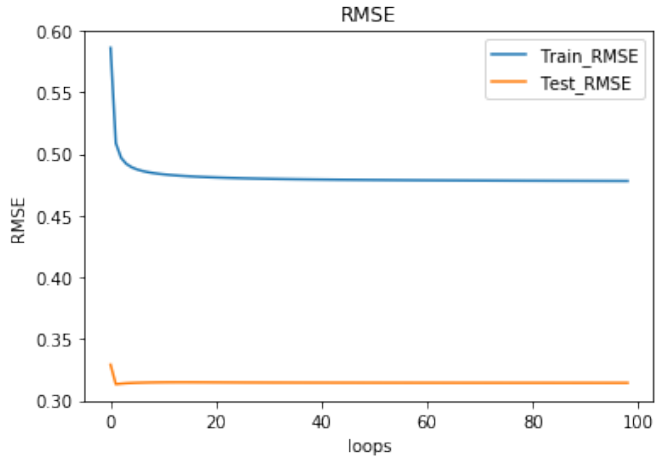

Fig. 3. The RMSE of the complete dataset u

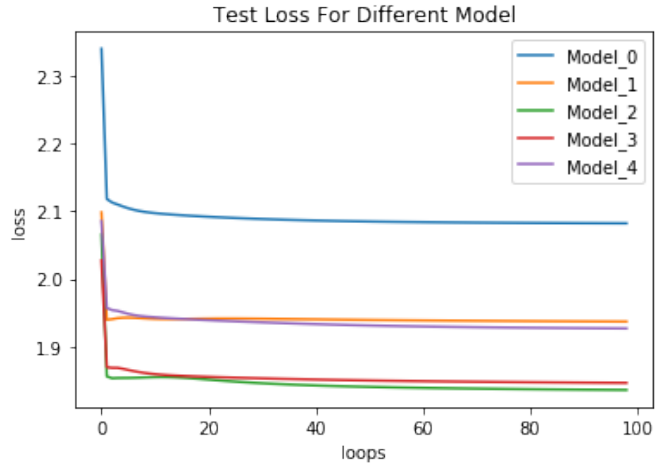

Fig. 4. The loss of all 5 divided datasets

IV. Conclusion

Through this experiment,we got a further understand of the principle of matrix decomposition as well as explored the construction of the recommended system.The use of gradient descent made us review the knowledge of gradient decent and naturally be more familiar to it.Although it cost sometime to run the experimental code,the result was of a great satisfaction to us.