Matrix Factorization

Yujie Wang, Lihao Xiao

03/28/2020

Factorization Method

The formula shown there is the simplest model:

$$\hat{r}_{ui} = q_i^T p_u$$

, without any bias terms or temporal effect. The purpose of this section is to give you a rough pipeline of how the method works. If you are assigned with Regularization terms as we will mention in section 2, you should modify the procedure (e.g: the objective function 1, 2 and 3 as well as the what paramters to be updated in the model) to reach your goal.

Gradient Descent

Nonprobabilistic

f: dimension of latent factors

$q_i \in \mathbb{R}^f$: factors associated with item i, measure the extent to which the item possesses those factors.

$p_u \in \mathbb{R}^f$: factors associated with user u, measure the extent of interest the user has in items that are high on the corresponding factors.

Generalized form of Gradient Descent method:

$$min_f \sum_{(u,i)\in K} V(r_{ui} - f(x_{i,u})) + \lambda * R(f)$$

Here V is an underlying loss function that describes the cost of predicting f(x) whe the true value is r_{ui}, the most common choice are square loss. λ is a parameter which controls the importance of the regularization term. R(f) is the typically chosen to impose a penalty on the complexity of f.

Note: if your team is not assigned to any penalized terms (none of the R1,R2,R3 from the Regularization Terms section, then you should not include the $\lambda * R(f)$)

for example: if we use the simplest model: $\hat{r}_{ui} = q_i^T p_u$: the objective function is the following with penalize magnitudes:

objective function 1:

$$min_{q^*p^*} \sum_{(u,i)\in K} (r_{ui} - q_i^T p_u)^2 + \lambda(||q_i||^2 + ||p_u||^2)$$

if it is the simplest model without penalize magnitudes the objective function becomes:

$$min_{q^*p^*} \sum_{(u,i)\in K} (r_{ui} - q_i^T p_u)^2$$

Probabilistic

Assumptions:

1. Conditional distribution over the observed ratings: $r_{iu}|q_i,p_u,\sigma^2 \sim N(\hat{r_{ui}}|q_i,p_u,\sigma^2)$
2. $q_i \sim N(0,\sigma_q^2)$
3. $p_u \sim N(0,\sigma_p^2)$

Using Bayes Rule

$p(q,p|r) = \frac{p(r,q,p)}{p(r)} \propto p(r,q,p) = p(r|q,p)p(q)p(p)$

Maximize $logp(q,p|r)$ is equivalent to minimizing the sum-of-squared-errors objective function with quadratic regularization terms:

objective function 2:

$$E = \frac{1}{2}\sum_{i=1}^{M}\sum_{u=1}^{U} I_{iu}(r_{ui} - q_i^T p_u)^2 + \frac{\sigma}{2\sigma_q}\sum_{i=1}^{M}||q_i||^2 + \frac{\sigma}{2\sigma_p}\sum_{u=1}^{U}||p_u||^2$$

Alternating Least Squares

General Formula of the Alternating Least Squares:

$$min_f \sum_{(u,i)\in K} V(r_{ui} - f(x_{i,u})) + \lambda * R(f)$$

Here the R is often the Tikhonov regularization term as mentioned in the paper. Again if you are not assigned to any penalize magnitudes, then the general form becomes

$$min_f \sum_{(u,i)\in K} V(r_{ui} - f(x_{i,u}))$$

Again for example: if we use the simplest model with penalize magnitudes: $\hat{r}_{ui} = q_i^T p_u$, the objective function is the following

objective function 3:

$$min_{q*p*} \sum_{(u,i)\in K} (r_{ui} - q_i^T p_u)^2 + \lambda(\sum_i n_{q_i}||q_i||^2 + \sum_u n_{p_u}||p_u||^2)$$

if if we use the simplest model without any penalize magnitudes, the objective function becomes:

$$min_{q*p*} \sum_{(u,i)\in K} (r_{ui} - q_i^T p_u)^2$$

- Step 1 Initialize matrix q by assigning the average rating for that movie as the first row, and small random numbers for the remaining entries. - Step 2 Fix q, solve p by minimizing the objective function; - Step 3 Fix p, solve q by minimizing the objective function similarly; - Step 4 Repeat Steps 2 and 3 until a stopping criterion is satisfied.

Difficulty Summary

Number	Factorization Method	Difficulty	Paper

Number	Factorization Method	Difficulty	Paper
1	SGD to minimize objective function 1	2	1
2	GD to minimize objective function 2	4	3
3	Alternating Least Squares to minimize objective function 3	4	4

Regularization Terms

In this section, your team will explore multiple variations of the model that is introduced in the last section. For R1, you should include penalized term to avoid overfitting. For R2 and R3, the model itself is modified and may include: R2. bias and intercepts or R3. linear modeling of user biases with temporal effect.

R1. penalize magnitudes: if the ratings are predicted with formula

$$\hat{r}_{ui} = q_i^T p_u$$

then the penalty term in the objective function should be:

$$\sum_{(u,i) \in K} \lambda(||q_i||^2 + ||p_u||^2)$$

R2. bias and intercepts: some users to give higher ratings than others, and for some items to receive higher ratings than others.

$$b_{ui} = \mu + b_i + b_u$$

Ratings are written as

$$\hat{r}_{ui} = b_{ui} + q_i^T p_u$$

penalty term in the objective function:

$$\sum_{(u,i) \in K} \lambda(||q_i||^2 + ||p_u||^2 + {b_i}^2 + {b_u}^2)$$

R3. linear modeling of user biases: treat the user bias as a function of time

$$\hat{r}_{ui}(t) = q_i^T p_u + \mu + b_i + b_u + \alpha_u dev_u(t)$$

similar penalty formula can be formed here for the temporal dynamic one

Difficulty Summary

Number	Regularization Terms	Difficulty	Paper
1	penalize magnitudes	1	1
2	bias and intercepts	3	1
3	temporal dynamics	4	5

Postprocessing

1. global bias correction: Given a prediction P , if the mean of P is not equal to the mean of the test dataset, we can shift all predicted values by a fixed constant $\tau = mean(test) - mean(P)$.
2. KNN: define similarity between movie i_1 and i_2 as cosine similarity between q_{i1} and q_{i2}

$$s(q_{i_1}, q_{i_2}) = \frac{q_{i_1}^T q_{i_2}}{||q_{i_1}||||q_{i_2}||}$$

Then we can use k-nearest neighbor prediction using similarity s. For each user, use the average ratings of similar movies to predict the ratings of movies the user has not rated.

3. Kernel Ridge Regression:

Discard all weights p_{uk}, try to predict rating of movie i r_{iu} for each user u using q_{ik} as predictors.

Define y as vector of ratings by users u;

X: for each row of X, normalized vector of factors for movie rated by user u

$$x_i = \frac{q_i}{||q_i||}$$

$$y = X\beta$$

Solve ridge regession:

$$\hat{\beta} = (X^T X + \lambda I)^{-1} X^T y$$

Prediction:

$$\hat{r_i} = K(x_i^T, X)(K(X, X) + \lambda I)^{-1} y$$

4. find x^* that minimizes RMSE(P_x) where $P_x = (1 - x)P_0 + xP_1$

P_0 and P_0 are two different predictors

Difficulty Summary

Number	Postprocessing	Difficulty	Paper
1	global bias correction	1	4
2	Postprocessing SVD with KNN	2	2
3	Postprocessing SVD with kernel ridge regression	4	2
4	linearly combine predictors	1	4

Linear model for each item (paper 2):

Get a prediction using Algorithms & Regularizations and another prediction in your Post Processing part and combine all the predictions through linear regression.

Reference

1. Y. Koren. “Matrix Factorization Techniques for Recommender Systems.” Journal Computer, 42 , no. 8, 2009, pp. 30–37.

2. A. Paterek, “Improving Regularized Singular Value Decomposition for Collaborative Filtering,” Proc. KDD Cup and Workshop, ACM Press, 2007, pp. 39-42.

3. R. Salakhutdinov and A. Mnih, “Probabilistic Matrix Factorization,” Proc. Advances in Neural Information Processing Systems 20 (NIPS 07), ACM Press, 2008, pp. 1257-1264.

4. Y. Zhou et al., “Large-Scale Parallel Collaborative Filtering for the Netflix Prize,” Proc. 4th Int’l Conf. Algorithmic Aspects in Information and Management, LNCS 5034, Springer, 2008, pp. 337-348.

5. Y. Koren, “Collaborative Filtering with Temporal Dynamics,” Proc. 15th ACM SIGKDD Int’l Conf. Knowledge Discovery and Data Mining (KDD 09), ACM Press, 2009, pp. 447-455.