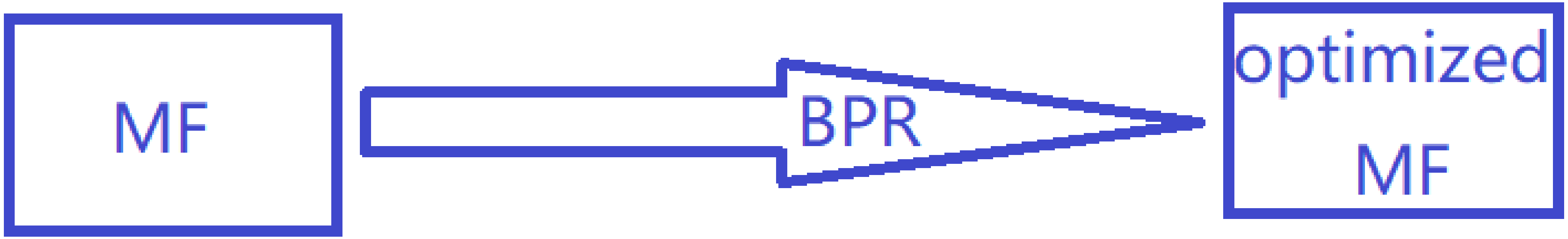
BPR

Bayesian Personalized Ranking

- 1 Background
- **2 Related Knowledge**
- **3 Core**
- **4 Application**
- 5 Achievement

Background

BPR: Bayesian Personalized Recommendation, an optimization of implicit feedback of recommendation.

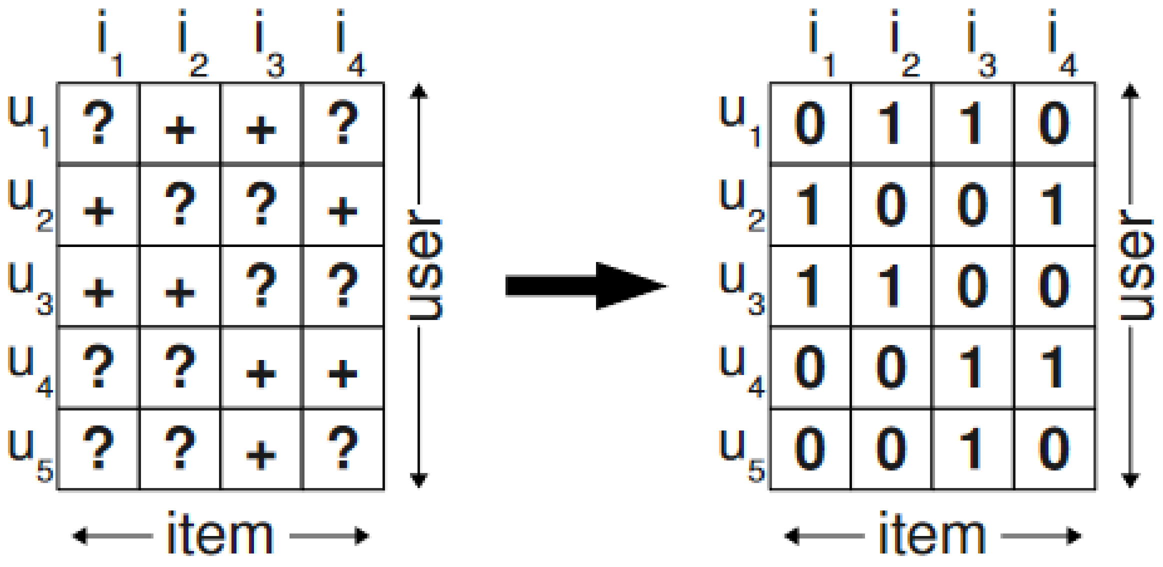


KNN,MF:

(1)Not designed for ranking

(2)Non-personalized

(3)Overfitting



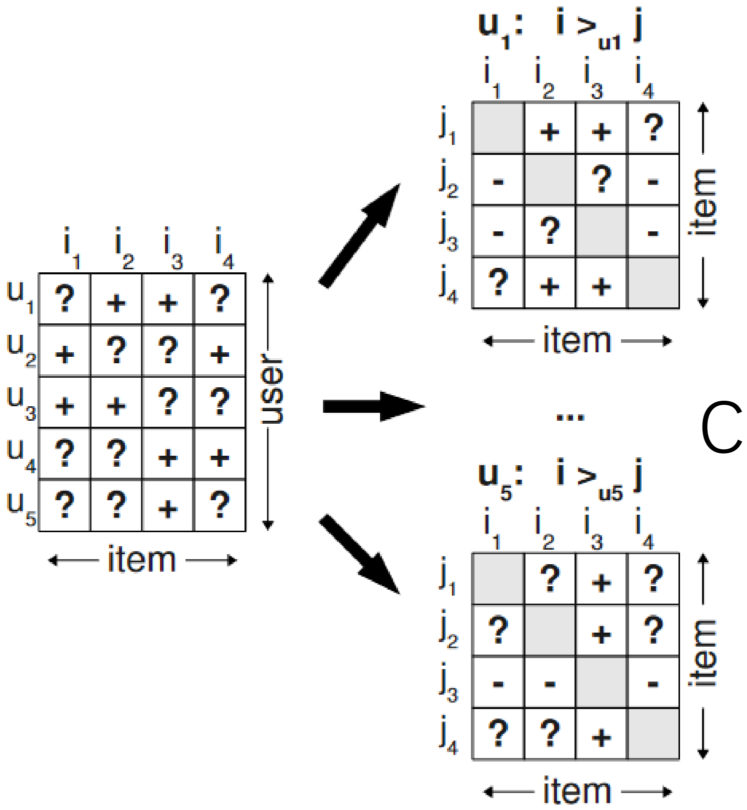
Simply recognizing all missing actions negative

BPR:

(1)Specific for ranking

(2)Personalized

(3)Only use the known data(won’t guess)



Comparison between items based on known data

Related Knowledge

- **Partial order: A subset of the Cartesian product of two collision.**

$$\{1,2,3\} \quad X \quad \{1,2\} \quad = \quad \left\{ \begin{bmatrix} (1,1) & (1,2) & (1,3) \\ (2,1) & (2,2) & (2,3) \\ (3,1) & (3,2) & (3,3) \end{bmatrix} \right\}$$

(Cartesian product)

$\{(2,1) \quad (3,1) \quad (3,2)\}$ **Partial order: > (the former element is larger than the latter)**

$\forall i, j \in I : i \neq j \Rightarrow i >_u j \vee j >_u i$ 2!=1, so whether 2>1 or 1>2 will be true

$\forall i, j \in I : i >_u j \wedge j >_u i \Rightarrow i = j$ Cannot be both 1>2 and 2>1

$\forall i, j, k \in I : i >_u j \wedge j >_u k \Rightarrow i >_u k$ 2>1, 3>2, then 3>1

The pair fitting the formula will be included in the partial order set

I = {item1,item2……item n}

$$
I * I = \begin{Bmatrix} <item1, item1>, <item1, item2> \cdots & <item1, item\,n> \\ <item2, item1>, <item2, item2> \cdots & <item2, item\,n> \\ \ldots\ldots\ldots\ldots & \\ <item\,n, item1>, <item\,n, item2> \cdots & <item\,n, item\,n> \end{Bmatrix}
$$

$$
>_u \in S \subseteq I * I
$$

Element $>_u$ means that to user u,
he prefers item I than item k

Related Knowledge

- **Bayesian formula:**

  If uses feature A to predict event B, the probability of A on event B occurred observed is called posterior probability; while the probability of the occurrence of event B on feature A known is called prior probability

$$P(AB) = P(A)*P(B|A)=P(B)*P(A|B)$$

$$
\begin{aligned}
\text{So}\quad P(B|A) &= P(AB) \;/\; P(A) \\
&= P(AB)*P(B) \;/\; P(A)*P(B) \\
&= (\,P(AB) / P(B)\,) * (\,P(B) \;/\; P(A)\,) \\
&= P(A \mid B) * P(B) \;/\; P(A) \\
&\propto P(A \mid B) * P(B)
\end{aligned}
$$

posterior probability

prior probability

Core

- **Optimization criterion: maximize posterior probability**

Maximize: $p(\Theta | >_u) \propto p(>_u | \Theta)\, p(\Theta)$

Θ : The parameter vector of a model to be optimized, representing a model.

$>_u$: To user u, a certain item i_1 is more preferred than i_2(inferred form the data)

$$\prod_{u \in U} p(>_u |\Theta) = \prod_{(u,i,j) \in D_S} p(i >_u j|\Theta)$$

$$p(i >_u j|\Theta) := \sigma(\hat{x}_{uij}(\Theta))$$

$$= \text{sigmoid}(\ MF_{\Theta}\ (u,i)\ -\ MF_{\Theta}\ (u,j\ )\ )$$

The predicted result of pair<u,i> on using model θ

$p(i >_u j|\Theta)$ It represents the probability for user u to prefer I than j on using model θ

$$
\begin{aligned}
\text{BPR-OPT} &:= \ln p(\Theta \mid >_u) \\
&= \ln p(>_u \mid \Theta)\, p(\Theta) \\
&= \ln \prod_{(u,i,j)\in D_S} \sigma(\hat{x}_{uij})\, p(\Theta) \\
&= \sum_{(u,i,j)\in D_S} \ln \sigma(\hat{x}_{uij}) + \ln p(\Theta) \\
&= \sum_{(u,i,j)\in D_S} \ln \sigma(\hat{x}_{uij}) - \lambda_\Theta ||\Theta||^2
\end{aligned}
$$

Bayesian formula

Equivalence

property of ln

Property of
Normal distribution

$$P(\theta) \sim N(0, \lambda_\theta I)$$

$$lnP(\theta) = -\lambda||\theta||^2$$

Core

- **Learning Algorithm: stochastic gradient-descent**

We want to optimize the model represented by parameter vector θ

$$\frac{\partial \text{BPR-Opt}}{\partial \Theta} = \sum_{(u,i,j)\in D_S} \frac{\partial}{\partial \Theta} \ln \sigma(\hat{x}_{uij}) - \lambda_\Theta \frac{\partial}{\partial \Theta} ||\Theta||^2$$

$$\propto \sum_{(u,i,j)\in D_S} \frac{-e^{-\hat{x}_{uij}}}{1+e^{-\hat{x}_{uij}}} \cdot \frac{\partial}{\partial \Theta} \hat{x}_{uij} - \lambda_\Theta \Theta$$

$$\Theta \leftarrow \Theta + \alpha \left(\frac{e^{-\hat{x}_{uij}}}{1+e^{-\hat{x}_{uij}}} \cdot \frac{\partial}{\partial \Theta} \hat{x}_{uij} + \lambda_\Theta \Theta \right)$$

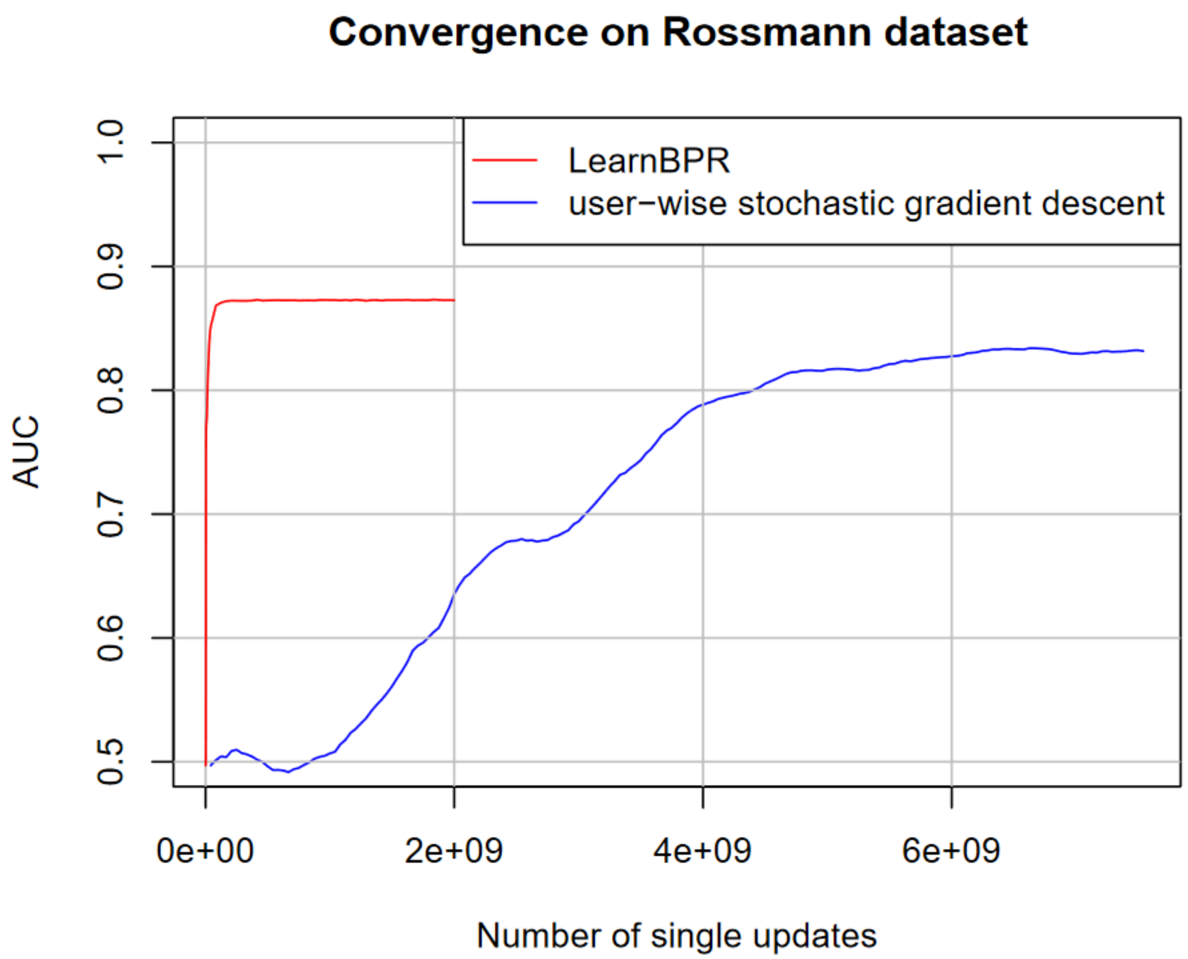
procedure LearnBPR(D_S, Θ)
    initialize Θ
    repeat
        draw (u, i, j) from D_S
        $\Theta \leftarrow \Theta + \alpha \left(\frac{e^{-\hat{x}_{uij}}}{1+e^{-\hat{x}_{uij}}} \cdot \frac{\partial}{\partial \Theta} \hat{x}_{uij} + \lambda_\Theta \cdot \Theta \right)$
    until convergence
    return $\hat{\Theta}$
end procedure

Stochastic gradient descent :
draw randomly from the data
Train until convergence

procedure **Normal**
    initialize Θ
    repeat
    iterate every <u,i,j> form Ds
        $\Theta \leftarrow \Theta + \alpha \left(\frac{e^{-\hat{x}_{uij}}}{1+e^{-\hat{x}_{uij}}} \cdot \frac{\partial}{\partial \Theta} \hat{x}_{uij} + \lambda_\Theta \cdot \Theta \right)$
    until convergence
    return Θ
end procedure

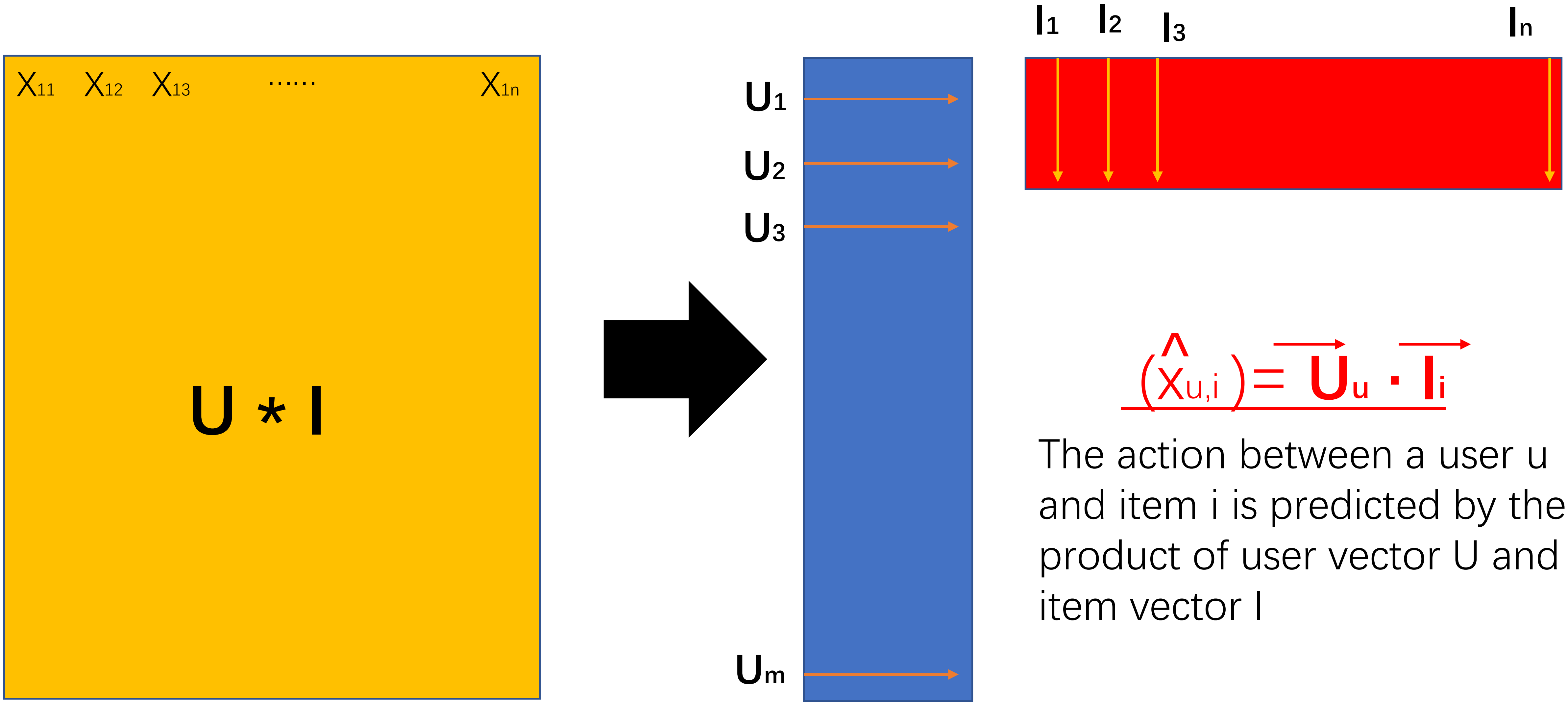
Full gradient descent :
draw every triple form the data
Train until convergence

LearnBPR did work



Application

Using MF to predict the action between a user and an item



$$\hat{X}_{u,i} = \vec{U}_u \cdot \vec{I}_i$$

The action between a user u and item i is predicted by the product of user vector U and item vector I

$$
\begin{aligned}
\text{BPR-OPT} &:= \ln p(\Theta| >_u) \\
&= \ln p(>_u |\Theta)\, p(\Theta) \\
&= \ln \prod_{(u,i,j)\in D_S} \sigma(\hat{x}_{uij})\, p(\Theta) \\
&= \sum_{(u,i,j)\in D_S} \ln \sigma(\hat{x}_{uij}) + \ln p(\Theta) \\
&= \sum_{(u,i,j)\in D_S} \ln \sigma(\hat{x}_{uij}) - \lambda_\Theta ||\Theta||^2
\end{aligned}
$$

$$\hat{x}_{uij} := \hat{x}_{ui} - \hat{x}_{uj}$$

$$= \overline{U}_u * \overline{I}_i - \overline{U}_u * \overline{I}_j$$

Θ is $\overline{U}_u, \overline{I}_i, \overline{I}_j$

Representing the model

procedure LEARNBPR(D_S, Θ)
  initialize Θ
  repeat
    draw (u, i, j) from D_S
    $\Theta \leftarrow \Theta + \alpha \left(\frac{e^{-\hat{x}_{uij}}}{1+e^{-\hat{x}_{uij}}} \cdot \frac{\partial}{\partial \Theta} \hat{x}_{uij} + \lambda_\Theta \cdot \Theta \right)$
  until convergence
  return $\hat{\Theta}$
end procedure

$$\frac{\partial}{\partial \theta} \hat{x}_{uij} = \begin{cases} (h_{if} - h_{jf}) & \text{if } \theta = w_{uf}, \\ w_{uf} & \text{if } \theta = h_{if}, \\ -w_{uf} & \text{if } \theta = h_{jf}, \\ 0 & \text{else} \end{cases}$$

Achievement

- Using AUC to evaluate all this model