

BPR: Bayesian Personalized Ranking

Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, Lars
Schmidt-Thieme

Presenter: 山田倫太郎

February 17, 2020

研究背景

推薦システムにおいて個人の嗜好を予測することは重要

大別すると 2 種類のデータが考えられる

- Explicit feedback
 - ▶ ユーザの興味が明示的に与えられているデータ
例) Amazon の☆や facebook の「いいね」
- **Implicit feedback**
 - ▶ ユーザの興味が明示的に与えられてないデータ
例) 購入履歴や閲覧履歴

目標

Implicit feedback を元にユーザの嗜好をランキングする

従来法

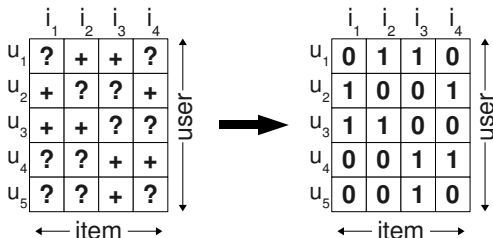
閲覧されたか否かを $\{0,1\}$ のラベルで表現

u : ユーザ

i : アイテム

$+$: 閲覧済み

$?$: 未閲覧



問題点

未閲覧のデータ中で以下を区別できない

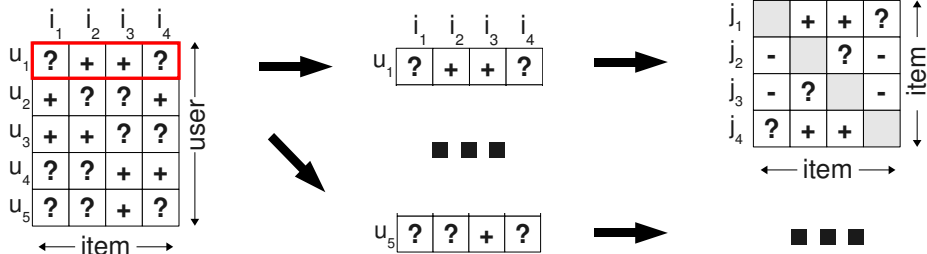
- ユーザの興味がないもの
- ユーザの興味があるが閲覧していないもの

提案手法

方針

ユーザの選好関係をモデル化する

仮定： 閲覧済みアイテムの興味 > 未閲覧アイテムの興味



記法

- U : ユーザの全体集合
- I : アイテムの全体集合
- U_i^+ : アイテム i を閲覧したユーザの全体集合
- I_u^+ : ユーザ u が閲覧したアイテムの全体集合
- D_s : フィードバックが得られているデータセット
 - ▶ $D_s := \{(u, i, j) \mid i \in I_u^+ \wedge j \in I \setminus I_u^+\}$
- $>_u$: ユーザ u の嗜好情報
- Θ : モデルパラメータ
- x_{ui} : ユーザ u のアイテム i の選好度合い

x_{ui} は推論モデルによって既定される値なのでパラメータ Θ に依存

D_s よりパラメータ Θ を導出したいので $p(\Theta \mid D_s)$ を最適化

モデリング

ベイズの定理より次の式が導かれる：

$$p(\Theta \mid D_s) \propto p(D_s \mid \Theta)p(\Theta)$$

ユーザ, アイテムの好みが独立であると仮定し, 尤度を以下で定める：

$$\prod_{(u,i,j) \in D_s} p(i >_u j \mid \Theta)$$

ここでシグモイド関数を σ として：

$$p(i >_u j \mid \Theta) := \sigma(\hat{x}_{ui} - \hat{x}_{uj})$$

目的関数: BPR-Opt

- 事前分布 $p(\Theta)$ は正規分布 $N(0, \lambda_{\Theta} I)$ とする
- MAP 推定を行う

$$\begin{aligned}\hat{\Theta} &= \arg \max_{\Theta} \ln p(\Theta \mid D_s) \\ &= \arg \max_{\Theta} \ln p(D_s \mid \Theta) p(\Theta) \\ &= \arg \max_{\Theta} \ln \prod_{(u,i,j) \in D_s} \sigma(\hat{x}_{ui} - \hat{x}_{uj}) p(\Theta)\end{aligned}$$

- 以下の目的関数を最大化する

$$\text{BPR-Opt} := \sum_{(u,i,j) \in D_s} \ln \sigma(\hat{x}_{ui} - \hat{x}_{uj}) - \frac{1}{2\lambda_{\Theta}} \|\Theta\|^2$$

目的関数: BPR-Opt

- BPR-Opt を微分すると

$$\begin{aligned}\frac{\partial \text{BPR-Opt}}{\partial \Theta} &= \sum_{(u,i,j) \in D_s} \frac{\partial}{\partial \Theta} \ln \sigma(\hat{x}_{ui} - \hat{x}_{uj}) - \frac{1}{2\lambda_{\Theta}} \frac{\partial}{\partial \Theta} \|\Theta\|^2 \\ &= \frac{-e^{\hat{x}_{ui} - \hat{x}_{uj}}}{1 + e^{\hat{x}_{ui} - \hat{x}_{uj}}} \cdot \frac{\partial}{\partial \Theta} (\hat{x}_{ui} - \hat{x}_{uj}) - \frac{\Theta}{\lambda_{\Theta}}\end{aligned}$$

- $(u, i, j) \in D_s$ は膨大となりやすいので確率的勾配法を用いて解く

学習アルゴリズム: Learn-BPR

```
1: procedure LEARN-BPR( $D_s, \Theta$ )
2:   initialize  $\Theta$ 
3:   repeat
4:     draw (u,i,j) from  $D_s$ 
5:      $\Theta \leftarrow \Theta + \alpha \left( \frac{-e^{\hat{x}_{ui} - \hat{x}_{uj}}}{1 + e^{\hat{x}_{ui} - \hat{x}_{uj}}} \cdot \frac{\partial}{\partial \Theta}(\hat{x}_{ui} - \hat{x}_{uj}) - \frac{\Theta}{\lambda_{\Theta}} \right)$ 
6:   until convergence
7:   return  $\hat{\Theta}$ 
8: end procedure
```

- $\frac{\partial}{\partial \Theta}(\hat{x}_{ui} - \hat{x}_{uj})$ については次に示す

x_{ui} の具体例 1 : Matrix Factorization

次元削減を行って \hat{x}_{ui} を推定する

The diagram illustrates the matrix factorization process. On the left, a large yellow square matrix X is shown, with a vertical bracket on its left labeled 'U (user)' and a horizontal bracket on its top labeled 'I (item)'. This matrix is followed by an approximation symbol \approx . To the right of the symbol is a blue rectangular matrix W with a horizontal bracket on its top labeled 'k'. This is followed by a multiplication symbol \times and a green rectangular matrix H^t with a horizontal bracket on its top labeled 'I (item)'. A vertical bracket on the right side of H^t is labeled 'k'. To the right of this product is a double-bordered box containing the equation $\hat{X} := WH^T$.

$$\begin{matrix} \text{U (user)} \\ \left[\begin{array}{c} \text{I (item)} \\ X \end{array} \right] \approx \left[\begin{array}{c} k \\ W \end{array} \right] \times \left[\begin{array}{c} \text{I (item)} \\ H^t \end{array} \right] \begin{matrix} \\ k \end{matrix} \end{matrix} \quad \boxed{\hat{X} := WH^T}$$

$$\begin{aligned} \hat{x}_{ui} &= \langle w_u, h_i \rangle \\ &= \sum_{f=1}^k w_{uf} \cdot h_{if} \end{aligned} \quad \frac{\partial}{\partial \Theta} (\hat{x}_{ui} - \hat{x}_{uj}) = \begin{cases} (h_{if} - h_{jf}) & \text{if } \theta = w_{uf} \\ w_{uf} & \text{if } \theta = h_{if} \\ -w_{uf} & \text{if } \theta = h_{jf} \\ 0 & \text{else} \end{cases}$$

x_{ui} の具体例 2 : Adaptive k-Nearest Neighbor

観測済みアイテムとの類似度をもとに \hat{x}_{ui} を推定する

$$\hat{x}_{ui} = \sum_{i \in I_u^+ \wedge l \neq i} c_{il}$$

$$\frac{\partial}{\partial \Theta}(\hat{x}_{ui} - \hat{x}_{uj}) = \begin{cases} +1 & \text{if } \theta \in \{c_{il}, c_{li}\} \wedge l \in I_u^+ \wedge l \neq i \\ -1 & \text{if } \theta \in \{c_{jl}, c_{lj}\} \wedge l \in I_u^+ \wedge l \neq j \\ 0 & \text{else} \end{cases}$$

実験

- ① Rossmann オンラインショップの購入履歴から、ユーザが次に買いたい品物を予測
- ② Netflix の過去の映画の評価履歴をもとに、与えられた映画に評価を行うかを予測

	ユーザー（人）	アイテム（個）	観測済み（個）
Rossmann	10,000	4,000	436,612
Netflix	10,000	5,000	565,738

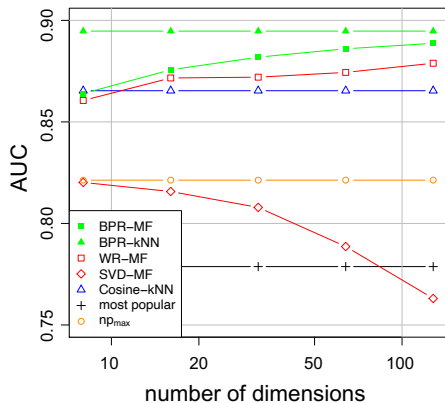
- ROC 曲線の AUC を評価

比較手法

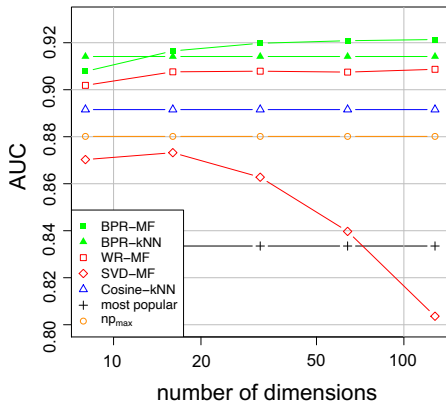
- **BPR-MF**
 - ▶ パラメータの更新に BPR-Opt を用いた MF
- **BPR-kNN**
 - ▶ パラメータの更新に BPR-Opt を用いた k 近傍
- SVD-MF
 - ▶ SVD（特異値分解）を適応したもの
- WR-MF
 - ▶ SVD（特異値分解）の過学習を抑えるように改良したもの
- Cosine-kNN
 - ▶ コサイン類似度を用いた k 近傍法
- most popular
 - ▶ 学習データの中で最も人気のものを推薦する
- np_{\max}
 - ▶ テストデータの中で最も人気のものを推薦する

結果

Online shopping: Rossmann



Video Rental: Netflix



実験において提案手法が他のモデルよりも優れた結果を示した

まとめ

BPR の手法

- 閲覧済みアイテムは未閲覧アイテムより興味があると仮定を置く
- ユーザのアイテム同士の選好関係をモデル化する
- パラメータ最適化に確率的勾配法を用いる

結果

BPR によるパラメータ最適化が他のモデルよりも優れている