

# **EFFICIENT BAYESIAN METHODS FOR GRAPH-BASED RECOMMENDATION**

**JAIME CASTRO**

Created: 2016-11-03 jue 11:50

# TABLE OF CONTENTS

- [Introducción](#)
- [Modelo](#)
- [Scoring functions](#)
- [Experimentos](#)
- [Resultados](#)
- [Referencias](#)

# INTRODUCCIÓN

# CONTEXTO

## CONTEXTO

Se ha mostrado que métodos basados en *random walks* sobre grafos bipartitos de usuarios e ítems dan buenas recomendaciones...

## **EL PROBLEMA**

Pero requieren demasiado espacio y tiempo. Para solventar esto se hace un muestreo, que reduce la calidad de las recomendaciones

# OBJETIVO

Ocupar enumeración de caminos cortos dentro de un grafo

# MODELO



## DEFINICIONES BÁSICAS

- Conjunto de usuarios  $U$
- Conjunto de ítems  $I$
- Conjunto de posibles ratings  $R$ . En este caso,  $R = \{0, 1\}$
- Conjunto de evaluaciones  $D = \{(u, i, r_{u,i} | u \in U, i \in I, r_{u,i} \in R)\}$

# GRAFO

## GRAFO

El grafo  $G$  se define como  $G = (U \cup I, E)$

El conjunto de aristas es tal que  $E = \{(u, i) \mid (u, i, r_{u,i}) \in D\}$

## CAMINOS DE LARGO 3

Dado un usuario  $u$ , el conjunto de los caminos de largo 3 desde él es

$$\mathcal{P}_u = \{(u, v, w, x) \mid v \in I_u, w \in U, x \in I, (u, v), (v, w), (w, x) \in E\}.$$

Todos los caminos de largo 3 a partir de algún usuario se define como  $\mathcal{P} = \cup_{u \in U} \mathcal{P}_u$

# VALUACIÓN

## VARIABLES ALEATORIAS

Para un ítem  $j$ , sea  $Y_j$  una variable aleatoria binaria que toma el valor 1 si recibe una calificación positiva, 0 si es una negativa.

$$\mathbb{P}(Y_j = 1) = \theta_j$$

## DISTRIBUCIÓN DE $\theta_j$

$$\theta_j \text{ Beta}(\hat{a}, \hat{b})$$

Cuando hay datos sobre  $R_j$  se actualiza a  $\text{Beta}(a, b)$

$$a = \hat{a} + |R_j^+| \text{ y } b = \hat{b} + |R_j^-|$$

## SCORING FUNCTION

$$f_u : I \rightarrow \mathbb{R}_+$$

$$f_u(x) = \sum_{(u,v,w,x) \in \mathcal{P}_u} s(u, v, w, x)$$

$s : \mathcal{P} \rightarrow \mathbb{R}_+$  es una scoring function.



# COMPLEJIDAD COMPUTACIONAL

## COMPLEJIDAD EN ESPACIO

$$\mathcal{O}(|U| \times \Delta_U)$$

Donde  $\Delta_U$  es el mayor grado de un nodo usuario

## COMPLEJIDAD EN TIEMPO

$$\mathcal{O}(|U| \times \Delta_I^2 \times \Delta_U)$$

Donde  $\Delta_U$  es el mayor grado de un nodo usuario y  $\Delta_I$  es el mayor grado de un nodo ítem

# SCORING FUNCTIONS

## POSTERIOR INEQUALITY SCORING (PIS)

$$s(u, v, w, x) = \begin{cases} \mathbb{P}(\theta_x > \theta_v | \mathcal{R}_x \mathcal{R}_v) & : v \neq x \\ 0 & : \text{o.c.} \end{cases}$$

## POSTERIOR PREDICTION SCORING (PPS)

$$s(u, v, w, x) = \begin{cases} \mathbb{P}(Y_v = 1 | \mathcal{R}_v) \times \mathbb{P}(Y_x = 1 | \mathcal{R}_x) & : v \neq x \\ 0 & : \text{o.c.} \end{cases}$$

## POSTERIOR ODDS RATIO SCORING (PORS)

$$s(u, v, w, x) = \begin{cases} \frac{\psi_x}{\psi_v} & : v \neq x \\ 0 & : \text{o.c.} \end{cases}$$

Donde  $\psi_i = \frac{\mathcal{P}(Y_i=1|R_i)}{\mathcal{P}(Y_i=0|R_i)}$  para  $i \in I$

# EXPERIMENTOS



# DATASET

**Table 1: Datasets properties.**

DataSet	$ D $	$ U $	$ I $	$\Delta_U$	$\Delta_I$	$\Gamma_U$	$\Gamma_I$
BookCrossing	42,137	1,842	2,065	964	225	22.87	20.40
Cds & Vynil	445,412	15,592	16,184	2,069	603	28.56	27.52
Electronics	347,393	20,247	11,589	317	1,376	17.16	29.98
Kindle	367,478	14,356	15,885	664	377	22.60	23.13
MovieLens 1M	998,539	6,040	3,260	2,233	3,428	165.32	306.30
FilmTrust	28,320	949	159	83	848	29.84	178.11
Epinions	300,304	10,706	8,945	525	1,491	28.05	33.57

Figure 1: Propiedades de los datasets

- Qué hacer con los ratings explícitos

Se calcula el promedio de los ratings por usuario  $\bar{r}_u, u \in U$

Se toma el rating como 1 si es mayor o igual a  $\bar{r}_u$ , 0 en otro caso.

## PARÁMETROS INICIALES

$$\hat{a} = 1$$

$$\hat{b} = 1$$

# BASELINES

Método
$P_{\alpha}^3$
$RP_{\beta}^3$
Most Popular (MP)
Bayesian Personalized Ranking Matrix Factorization (BPRMF)
Weighted Regularized Matrix Factorization (WRMF)

$$P_{\alpha}^3$$

Método basado en ideas Markovianas.

$P$  es una matriz de transición.

$P^3$  es la distribución de los *random walks* para tres pasos.

Este método ocupa esta matriz elevada a  $3 \cdot \alpha$

$$RP_{\beta}^3$$

Método basado en el anterior que mejora el anterior tomando en cuenta la popularidad de los ítems

# MÉTRICAS

- MAP
- MRR
- P@5
- P@10
- NDCG@5
- NDCG@10

# RESULTADOS

Para casi todos los datasets y métricas, PPS es mejor que los baselines y las otras propuestas de estos autores con significancia  $p < 0.01$ .

La excepción es el dataset de Movielens, en particular no logra batir a los algoritmos de factorización matricial

# REFERENCIAS

- Lopes, Ramon, Renato Assunção, and Rodrygo LT Santos. "Efficient Bayesian Methods for Graph-based Recommendation." Proceedings of the 10th ACM Conference on Recommender Systems. ACM, 2016.