CoFi^{RANK}
Maximum Margin Matrix Factorization for Collaborative Ranking

Markus Weimer
Karatzoglou
Quoc Viet Le
Alex Smola
読む人：藤巻＠NEC

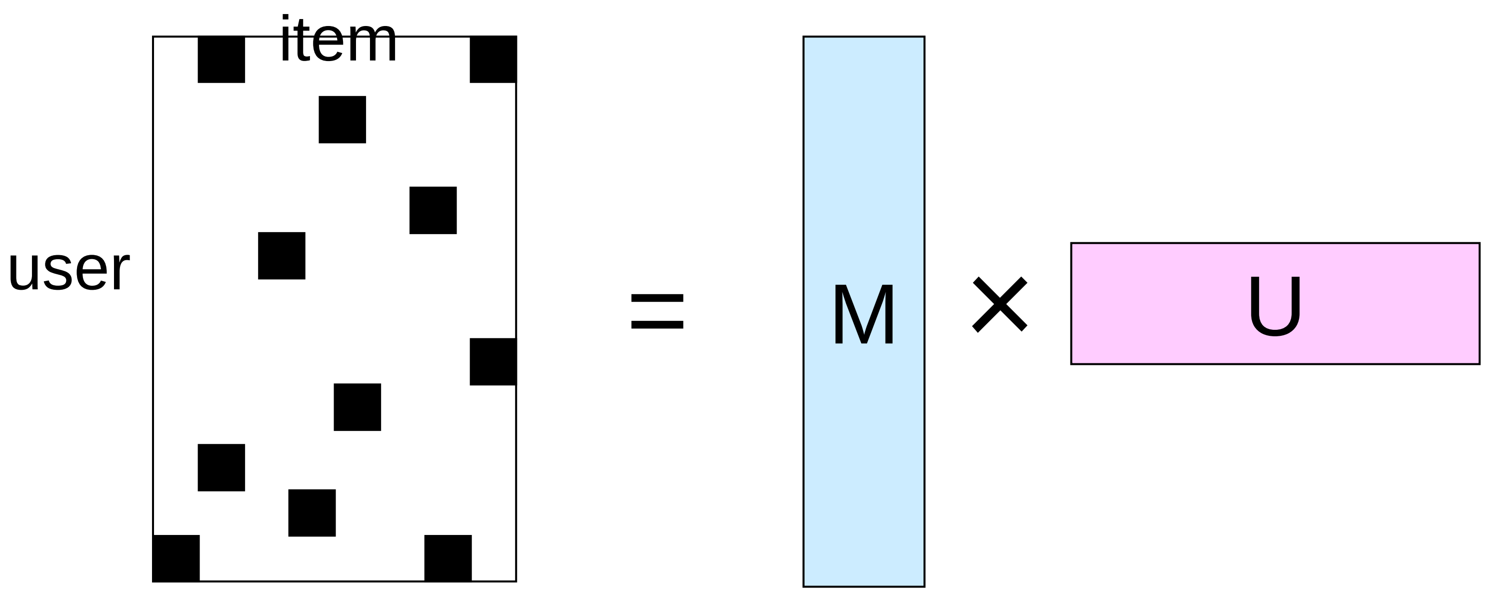
大体の流れ的な感じ

- 推薦系のシステムでは，「嫌いなもの」よりも「好きなもの」を当てるのが大事
- ランキングに適したスコア（ロス）を使おう
- 非凸だから凸に緩和（**Convex Upper Bound**）
- 問題は**MMMF**に定式化する
- 最適化は**Bundle Method**を適用

Convex Upper Bounds
Maximum Margin Matrix Factorization
Bundle Method
とかの技を知れてちょっとお得

Collaborative Rating

- Ranking
- Regression
- Matrix Factorization
  - Nonnegative Matrix Factorization 低ランク＋非負制約を利用
  - Maximum Margin Matrix Factorization トレースノルム（or maxノルム）を利用



■に合うようにMとUを推定して全体を復元する

推薦系システムではユーザーの嗜好上位のものを当てる事が大事

(Normalized) Discounted Cumulative Gain

$y \in \{1, \ldots, r\}^n$ アイテムのレートベクトル(降順)

π 1～nの順列

π_i i番目のアイテムのランク

$k \in \mathbb{N}$ 適当な閾値(上位幾つ推薦するか)

$$\mathrm{DCG@k}(\pi, y) = \sum_{i=1}^{k} \frac{(2^{y_i} - 1)}{\log(\pi_i + 1)}$$

$$\mathrm{NDCG@k}(\pi, y) = \frac{\mathrm{DCG@k}(\pi, y)}{\mathrm{DCG@k}(\mathrm{argsort}(y), y)}$$

yがπによって降順にソートされる場合にDCGは最大
(上位のものほどスコアが大きくなる)

$$R(\mathcal{X}, F, Y) := \sum_{i \in \mathcal{X}} \mathrm{NDCG}(\Pi^i, Y^i)$$

Rを最大化するようなFを推定する → Rは非凸で扱いにくい

Hop-Step-Jump : Non-Convex → Convex

Hop : Rを各ユーザーに分解し, -NDCGを最小化

$$\Delta(\pi, y) := \mathrm{NDCG}(1_\pi, y) - \mathrm{NDCG}(\pi, y)$$ （$\pi=1_\pi$で最小）

Step : 推定されたfとπによってソートしたfの差を考える

Polya-Littlewood-Hardy Inequality : $\langle a, b \rangle \le \langle \mathrm{sort}(a), \mathrm{sort}(b) \rangle$

$$\psi(\pi, f) := \langle \boxed{c}, \boxed{f_\pi} \rangle \qquad c_i = (i+1)^{-0.25}$$

任意の降順ベクトル　fをπで並べ替えたベクトル

Jump : 非凸なΔの変わりに凸なUpper Boundを考える

$$l(f, y) \ge \Delta(y, \pi^*)$$

$$l(f, y) := \max_\pi \left[\Delta(\pi, y) + \langle c, f_\pi - f \rangle \right]$$

Maximum Margin Matrix Factorization

Loss : $L(\mathcal{X}, F, Y) := \sum_{i \in \mathcal{X}} l(F^i \circ \Pi^i, \mathrm{sort}(Y^i))$

where $\Pi^i = \mathrm{argsort}(Y^i)$

Regularize : $\Omega[F] := \frac{1}{2} \min_{M,U} \left[\mathrm{tr}\, M M^\top + \mathrm{tr}\, U U^\top\right]$

subject to $MU = F$

結局. . .

$$\underset{M,U}{\mathrm{minimize}}\, L(\mathcal{X}_{\mathrm{train}}, MU, Y) + \frac{\lambda}{2}\left[\mathrm{tr}\, M M^\top + \mathrm{tr}\, U U^\top\right]$$

repeat

For fixed M minimize (8) with respect to U.

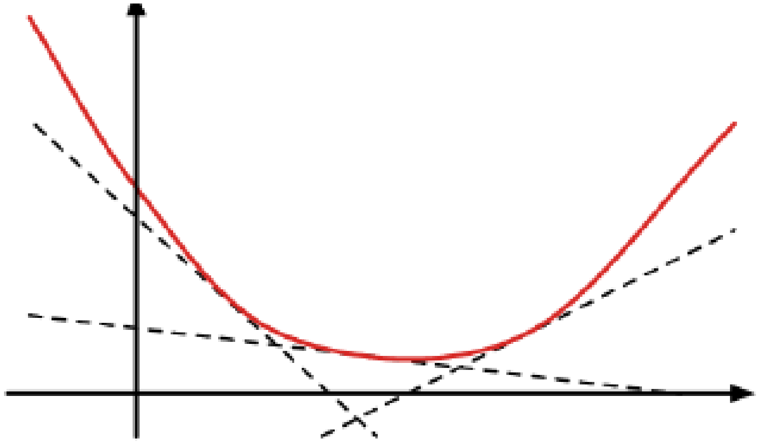
For fixed U minimize (8) with respect to M.

until No more progress is made or a maximum iteration

Bundle Methods

イメージ的には，凸損失のlower boundを線形近似で抑えておいて
lower boundの最大値を凸損失の代わりに使って近似を繰り返す

$$g(w) \geq \max_{\bar{w} \in W} \left[g(\bar{w}) + \langle w - \bar{w}, \partial_w g(\bar{w}) \rangle \right]$$



Algorithm 1 Bundle Method

Initialize $i = 1$, $w_0 = 0$, and $W = \{w_0\}$.
repeat
  Compute gradient a_i and offset b_i.
  Find minimizer $w_i := \mathrm{argmin}_w J_i(w)$
  Update $W \leftarrow W \cup \{w_i\}$ and $i \leftarrow i + 1$.
until converged

ちゃんと最適値に収束する

Optimization using Bundle Methods

Algorithm 1 Bundle Method(ϵ)

Initialize $t = 0, W_0 = 0, b_0 = 0$ and $H = \infty$
repeat
Find minimizer M_t and value L of the optimization problem

$$\underset{U}{\text{minimize}} \max_{0 \le j \le t} \left[\operatorname{tr} W_j^\top U + b_j\right] + \frac{\lambda}{2} \operatorname{tr} U^\top U.$$

Compute $W_{t+1} = \partial_U L(\mathcal{X}_{\text{train}}, M_t U, Y)$
Compute $b_{t+1} = L(\mathcal{X}_{\text{train}}, M_t U, Y) - \operatorname{tr} M_t W_{t+1}$
if $H' := \operatorname{tr} W_{t+1}^\top M_t + b_{t+1} + \frac{\lambda}{2} \operatorname{tr} U U^\top \le H$ **then**
Update $H \leftarrow H'$
end if
until $H - L \le \epsilon$

Computing the loss

理由が謎の式変形･･･

$$\Delta(\pi, y) := \mathrm{NDCG}(1_\pi, y) - \mathrm{NDCG}(\pi, y)$$

$$l(f, y) := \max_\pi \Big[\Delta(\pi, y) + \langle c, f_\pi - f \rangle \Big]$$

$$\mathrm{const.} - \sum_{i=1}^{n} a_{\pi_i} b(y)_i = \mathrm{const.} - \langle a_\pi, b(y) \rangle$$

$$a_i = \frac{1}{\log(i+1)} \quad b(y)_i = \frac{2^{y_i} - 1}{\mathrm{DCG}@k(1_\pi, y)}$$

$$l(f, y) = \mathrm{const.} + \max_\pi \big[\langle f_\pi, c \rangle - \langle a_\pi, b(y) \rangle \big]$$

Hungarian Marrigae algorithmで解ける

（πがpermutationだから整数計画になってるのが難しい）

Experiments

Dataset	Users	Movies	Ratings
EachMovie	61265	1623	2811717
MovieLens	983	1682	100000
Netflix	480189	17770	100480507