

# Doubly Robust Estimator for Ranking Metrics with Post-Click Conversions

---

RecSys論文読み会 20/10/11

齋藤優太 (<https://usaito.github.io/>)

半熟仮想/東京工業大学

# 問題設定

# Implicit Feedbackのログデータを用いたオフライン評価

## アマゾンの検索で「statistics」というqueryを投げてみる

ESL (カステラ本) をクリック！



$$\{(u, i, z_{u,i})\}$$

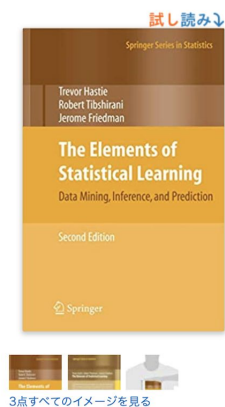
(user, item, click)で構成される  
ログデータが収集される



# 実際に多くのサービスにおいてfeedbackは2段階(以上)

Amazonの例では、クリックが発生した後にアイテム個別のページにて  
購買有無 (conversion) が発生

- 我々が最大化したい直接的な変数
- 他のアイテムの影響を受けない正確なデータ



著者をフォロー



Robert Tibshirani

+ フォロー

The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition (Springer Series in Statistics) (英語) ハードカバー – 2009/3/1

Trevor Hastie (著), Robert Tibshirani (著), Jerome Friedman (著)

★★★★☆ 311個の評価

ベストセラー1位 - カテゴリ Bioinformatics

> その他 (3) の形式およびエディションを表示する

Kindle版 (電子書籍)  
¥8,668  
獲得ポイント: 87pt

今すぐお読みいただけます: 無料アプリ

ハードカバー  
¥9,218  
獲得ポイント: 92pt prime

¥9,107 より 6 中古品  
¥8,636 より 19 新品

6/5 金曜日 8:00-12:00 にお届けするには、今から3 時間 33 分以内にお届け日時指定便を選択して注文を確定してください (有料オプション)。Amazonプライム会員は無料) 詳細を見る

This book describes the important ideas in a variety of fields such as medicine, biology, finance, and marketing in a common conceptual framework. While the approach is statistical, the emphasis is on concepts rather than mathematics. Many examples are given, with a liberal use of colour graphics. It is a valuable resource for statisticians and anyone interested in data mining in science or industry. The book's coverage is broad, from supervised learning (prediction) to unsupervised learning. The many

< 続きを読む

不正な製品情報を報告。

シェアする    

¥9,218  
参考価格: ¥9,239  
OFF: ¥12  
ポイント: 92pt (1%)  
詳細はこちら

お届け日時指定便 無料  
残り3点 (入荷予定あり)  
Kindle版は今すぐお読みいただけます。Kindle無料アプリがあれば、さまざまなデバイスで読書が可能。在庫状況について


この商品は、Amazon.co.jp が販売、発送します。

数量:

 カートに入れる

 今すぐ買う

◎ 斎藤 優太 - 152-0012 にお届け

ほしい物リストに追加する 

## 実際に多くのサービスにおいてfeedbackは2段階(以上)

---

- Amazon推薦では、アイテムページをクリックした後に購買有無を決定
- Spotify推薦では、楽曲ページをクリックした後に最後まで聞くかを決定
- Netflix推薦では、動画ページをクリックした後に最後まで視聴するかを決定

Click (first-stage) -> Conversion (second-stage)

ほとんどの場合、2段階目のfeedbackが収益やユーザ体験に直接的に関与

# Implicit Feedbackによるオフライン評価の例

例) あるUserに対して10個のアイテムを並べ替えるとき

Ranking	Recommender A	Recommender B
1	🔴: Click=1	🔴✕: Click=0
2	🔴: Click=1	🔴: Click=1
3	🔴: Click=1	🔴✕: Click=0
----	----	----
9	🔴✕: Click=0	🔴: Click=1
10	🔴✕: Click=0	🔴: Click=1

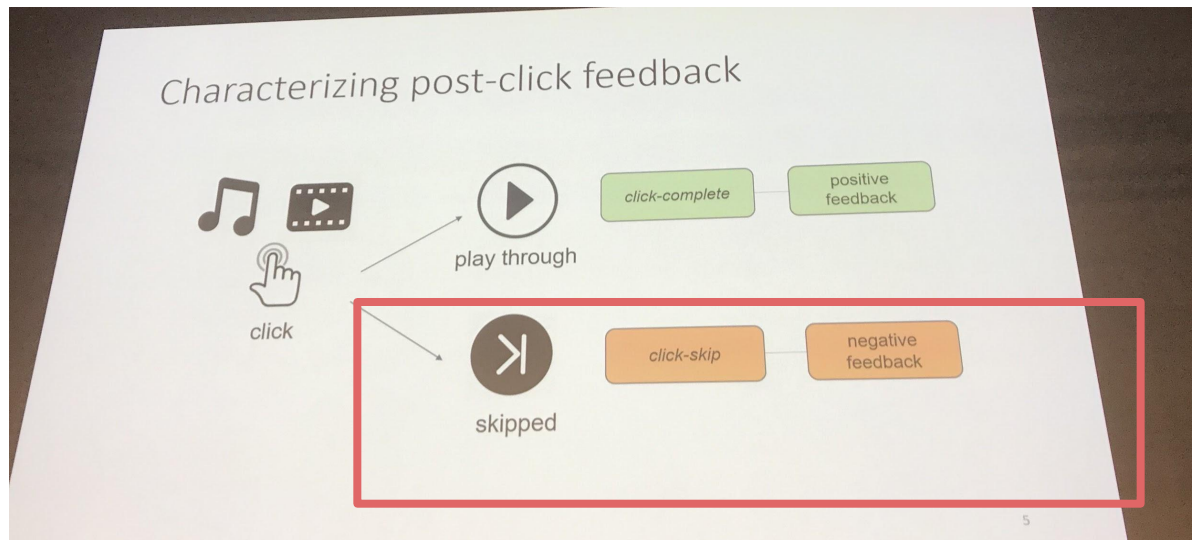
Implicit Feedbackのログを用いた  
標準的なランキング評価  
=クリックされるアイテムを多く上位で  
推薦するモデルが好評価  
(この例では、Recommender A)

本当にこのオフライン評価で  
良いのか？

# Implicit Feedbackはとてもnoisy (from RecSys2019)

## 音楽/映画推薦の例

一度clickしても  
その後すぐにskip  
していたらnegative



RecSys2019@コペンハーゲン参加時に撮影

[Leveraging Post-click Feedback for Content Recommendations \(Wei et al., RecSys'19\)](#)

## 2段階feedbackを用いたオフライン評価を考える

例) あるUserに対して10個のアイテムを並べ替えるとき

Ranking	Recommender A	Recommender B
1	✕: CV=0	✕: CV=0
2	✕: CV=0	◎: CV=1
3	✕: CV=0	✕: CV=0
----	----	----
9	◎: CV=1	✕: CV=0
10	✕: CV=0	◎: CV=1

CV確率が高いアイテムを  
上に並べられる推薦モデルを  
高く評価したい  
(この例だと、**Recommender B**)

CV情報を用いて真のランキング性  
能を定義するのが良さそう



## 2段階feedbackを用いた真のランキング性能

真のランキング性能

= 関数 $c(\cdot)$ で重み付けたコンバージョン数のユーザ平均

$$\mathcal{R}_{GT}(\hat{Z}) = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}} p_{u,i}^{cvr} \cdot c(\hat{Z}_{u,i})$$

評価対象の推薦モデルによる出力(予測関連スコア)

user

item

コンバージョン率

ranking function (DCG, Recallなど)

## Implicit Feedbackのログデータを用いたオフライン評価

---

ランキング関数  $c(\cdot)$  は、色々な種類が存在(今回はさほど重要ではない)

Average Relevance Position:

$$c(\hat{S}_{u,i}) = \text{rank}(\hat{S}_{u,i} \mid \{\hat{S}_{u,j}\}_{j \in \mathcal{I}})$$

Discounted Cumulative Gain:

$$c(\hat{S}_{u,i}) = \log_2(1 + \text{rank}(\hat{S}_{u,i} \mid \{\hat{S}_{u,j}\}_{j \in \mathcal{I}}))^{-1}$$

## 2段階feedbackを用いたオフライン評価の肝

観測データのみからどうやって真のランキング性能を推定するか？

真の性能



代替案  
(推定量)

$$\mathcal{R}_{GT}(\hat{Z}) = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}} \underline{p_{u,i}^{cvr}} \cdot c(\hat{Z}_{u,i})$$



$$\hat{\mathcal{R}}(\hat{Z}) = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}} \underline{???} c(\hat{Z}_{u,i})$$

実際に使えるのはclickが発生した場合のconversionのみ

オフライン評価に使えるログデータ

$$\{ (u, i, \underline{y_{u,i}}) \mid \underline{z_{u,i}} = 1 \}$$

conversionラベル                      click発生データ

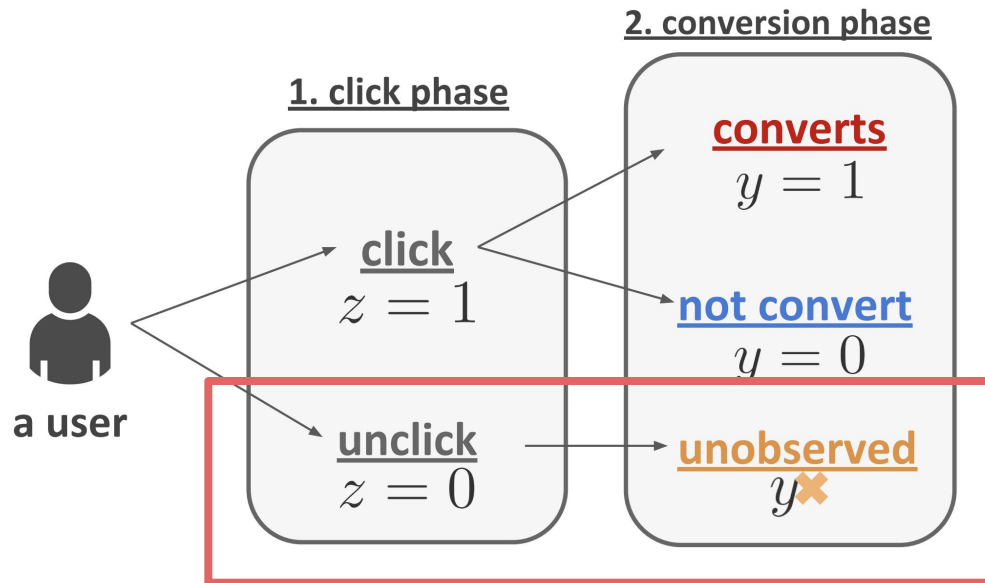
clickが発生したデータに関するconversionラベル

## 実際に使えるのはclickが発生した場合のconversionのみ

2段階feedbackが得られる過程を  
モデル化してみる(右図)

clickが発生したデータに  
ついてはconversionが観測

一方、clickが未発生だと  
conversionも未観測...



(b) user behavior pattern

## 2段階feedbackを用いたオフライン評価の肝

---

観測データのみからどうやって真のランキング性能を推定するか？

真の性能



代替案  
(推定量)

$$\mathcal{R}_{GT}(\hat{Z}) = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}} \underline{p_{u,i}^{cvr}} \cdot c(\hat{Z}_{u,i})$$
$$\hat{\mathcal{R}}(\hat{Z}) = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}} \underline{???} c(\hat{Z}_{u,i})$$

## ここまでのまとめ

---

- 多くの実務現場において推薦モデルのランキング性能は implicit feedbackを使ってオフライン評価されるが...
  - clickしたからといってpositiveな反応だとは言いきれない
  - clickしなくても単にアイテムのことを知らなかっただけかもしれない
- 多くの場合、ユーザーfeedbackは2段階 (click -> conversion)となっており、往々にして2段階目のデータが直接的にKPIに関与

**2段階目のfeedbackに対する  
ランキング性能のオフライン評価について考えたい**

手法



## ベースライン手法①: Naive Estimator

---

真の性能



代替案  
(推定量)

$$\mathcal{R}_{GT}(\hat{Z}) = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}} \underline{p_{u,i}^{cvr}} \cdot c(\hat{Z}_{u,i})$$



$$\hat{\mathcal{R}}_{Naive}(\hat{Z}) = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}} \underline{z_{u,i} y_{u,i} c(\hat{Z}_{u,i})}$$

clickが発生した(conversionが観測されている)データに絞って計算

## ベースライン手法①: Naive Estimator

---

この方法は実装は非常に簡単で多くの実務現場や論文の実験評価で使われているが、以下のように**真のランキング性能に対してバイアスを持つ**

$$\mathbb{E} \left[ \hat{\mathcal{R}}_{Naive}(\hat{Z}) \right] \neq \mathcal{R}_{GT}(\hat{Z})$$

selection bias (=観測データは全体を反映しない)を無視しているため

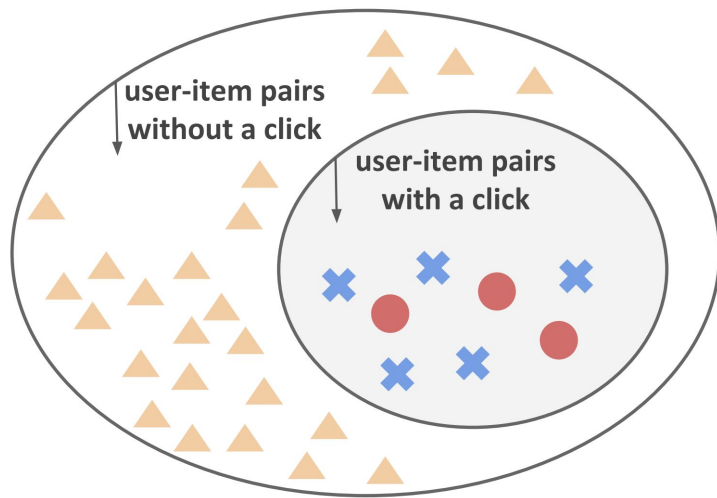
## Challenge 2: Selection Bias

観測されるconversionデータは全体を代表するサンプルになっていない

例えば、上位で推薦されやすかったアイテムに関するconversionデータは他のアイテムのデータよりも集まりやすい



click発生データは全体の偏った一部分  
(本来は散在しているイメージ)



(a) selection bias problem

## ベースライン手法: IPS Estimator (Yang et al. 2018)

---

真の性能



$$\mathcal{R}_{GT}(\hat{Z}) = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}} \underline{p_{u,i}^{cvr}} \cdot c(\hat{Z}_{u,i})$$




代替案  
(推定量)

$$\hat{\mathcal{R}}_{IPS}(\hat{Z}) = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}} \frac{z_{u,i} y_{u,i}}{\underline{p_{u,i}^{ctr}}} c(\hat{Z}_{u,i})$$

## ベースライン手法: IPS Estimator

---

$$p_{u,i}^{ctr} = \mathbb{P}(z_{u,i} = 1)$$

$$\hat{\mathcal{R}}_{IPS}(\hat{Z}) = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \sum_{i \in I: z_{u,i}=1} \frac{y_{u,i}}{p_{u,i}^{ctr}} c(\hat{Z}_{u,i})$$


clickが発生する確率 (conversionが観測される確率)の逆数で重み付け  
= ログデータ上でなかなか観測されにくいデータをあえて過大に評価

## ベースライン手法: IPS Estimator

---

IPS推定量は**真のランキング性能に対して不偏 (unbiased)**

$$\mathbb{E} \left[ \hat{\mathcal{R}}_{IPS}(\hat{Z}) \right] = \mathcal{R}_{GT}(\hat{Z})$$

この不偏性自体は望ましい性質ではあるが...

- IPS推定量は分散が大きくなってしまいう問題が知られている
- とりわけ、推薦の設定だとCTRが非常に小さい(スパース)ので大きな問題になり得る

## ベースライン手法: IPS Estimator

---

実際に分散を調べてみると、CTRの逆数の和が現れる

IPS推定量の分散 (Theorem 3.3)

$$\mathbb{V} \left( \hat{\mathcal{R}}_{IPS}(\hat{Z}) \right) = \frac{1}{|\mathcal{U}|^2} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}} \left( \frac{1}{p_{u,i}^{ctr}} - p_{u,i}^{cvr} \right) p_{u,i}^{cvr} c \left( \hat{Z}_{u,i} \right)^2$$

不偏性は望ましいが、結局数少ないclickが発生した  
(観測されている)データしか使っていないので効率が悪い

## 提案手法: Doubly Robust (DR) Estimator

---

真の性能



代替案

$$\mathcal{R}_{GT}(\hat{Z}) = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}} \underline{p_{u,i}^{cvr}} \cdot c(\hat{Z}_{u,i})$$



$$\hat{\mathcal{R}}_{DR}(\hat{Z}) = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}} \underline{\left( \frac{z_{u,i}}{p_{u,i}^{ctr}} (y_{u,i} - \hat{p}_{u,i}^{cvr}) + \hat{p}_{u,i}^{cvr} \right)} c(\hat{Z}_{u,i})$$



## 提案手法: Doubly Robust (DR) Estimator

---

ログとしてconversionが観測されていないデータについて  
事前に推定したCTRを補完してあげる

$$\hat{\mathcal{R}}_{DR}(\hat{Z}) = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}} \left( \frac{z_{u,i}}{p_{u,i}^{ctr}} (y_{u,i} - \hat{p}_{u,i}^{cvr}) + \hat{p}_{u,i}^{cvr} \right) c(\hat{Z}_{u,i})$$


ログデータからMFなどを用いてCVRを推定したもの

## 提案手法: Doubly Robust (DR) Estimator

---

DR推定量も期待値をとると、真のランキング性能に一致

$$\mathbb{E} \left[ \hat{\mathcal{R}}_{DR}(\hat{Z}) \right] = \mathcal{R}_{GT}(\hat{Z})$$

不偏性に関してはIPSとDRは同じ性質

## 提案手法: Doubly Robust (DR) Estimator

---

補完に使っているCVRの推定が悪くなければ分散が改善

$$\mathbb{V} \left( \hat{\mathcal{R}}_{DR}(\hat{Z}) \right) \leq \mathbb{V} \left( \hat{\mathcal{R}}_{IPW}(\hat{Z}) \right)$$

**分散改善のための十分条件** (推定誤差が $\pm 100\%$ 以内であれば良い)

$$0 \leq \hat{p}_{u,i}^{cvr} \leq 2p_{u,i}^{cvr}$$

# まとめ

---

## 観測可能な2段階feedbackから真のランキング性能を推定する手法

- **Naive:** selection biasを考慮しておらず、推定に統計的バイアスあり
- **IPS:** CTRの逆数による重み付けで不偏性を保つが、分散が大きい
- **DR (ours):** clickが発生していないデータのCVRを事前に推定して代用することで、不偏性を保ちつつ(緩い条件下で)分散を改善

実験

# 実データ実験 (with Yahoo! R3 and Coat)

## Yahoo! R3とCoatデータを用いてオフライン評価の正確さを評価

### R3 - Yahoo! Music ratings for User Selected and Randomly Selected songs, version 1.0 (1.2 MB)

This dataset contains ratings for songs collected from two different sources. The first source consists of ratings supplied by users during normal interaction with Yahoo! Music services. The second source consists of ratings for randomly selected songs collected during an online survey conducted by Yahoo! Research. The rating data includes 15,400 users, and 1000 songs. The data contains at least ten ratings collected during normal use of Yahoo! Music services for each user, and exactly ten ratings for randomly selected songs for each of the first 5400 users in the dataset. The dataset includes approximately 300,000 user-supplied ratings, and exactly 54,000 ratings for randomly selected songs. All users and items are represented by randomly assigned numeric identification numbers. In addition, the dataset includes responses to seven multiple-choice survey questions regarding rating-behavior for each of the first 5400 users. The survey data and ratings for randomly selected songs were collected between August 22, 2006 and September 7, 2006. The normal-interaction data was collected between 2002 and 2006. The size of this dataset is 1.2 MB.

Here are all the papers published on this Webscope Dataset:

- [Collaborative Prediction and Ranking with Non-Random Missing Data](#)
- [Missing Data Problems in Machine Learning](#)
- [Collaborative Filtering and the Missing at Random Assumption](#)

[See all publications](#)

Select this Dataset

View Cart

### Recommendations as Treatments: Debiasing Evaluation and Learning

All data and code is released under the [Creative Commons BY-NC license](#).

Code

Download

#### Installation

This code has been developed under Python 2.7. Assuming you have a copy of Python 2.7 already running, please install all required dependencies by following the steps below.

Linux:

```
pip install -r requirements.txt
```

Windows (assuming the Anaconda Distribution):

```
conda install --yes --file requirements.txt
```

#### First Steps

To run the code on the Coat dataset below (already included in source code), run the script files in the `examples/` subdirectory.

Linux:

これらのデータはバイアスを含むtrainと含まないtestが分けられて収録されており、オフライン評価の正確さの評価が可能

# 実験手順

---

## オフライン評価の正確さの評価のための実験手順

1. オリジナルのtrainデータをtrainとvalに分ける
2. trainデータを用いて32個の推薦モデルを学習
3. **valデータと推定量を用いて推薦モデルの性能をオフライン評価**
4. testデータを用いて推薦モデルの真のランキング性能を推定
5. 3と4の値を比べることでオフライン評価の正確さを評価

より詳細なデータの処理は、論文を参照してみてください

## オフライン評価の正確さの評価指標

---

次の*relative-RMSE* によりオフライン評価の正確さを評価

$$relative-RMSE(\hat{\mathcal{R}}) = \sqrt{\frac{1}{|\mathcal{M}|} \sum_{\hat{Z} \in \mathcal{M}} \left( \frac{\mathcal{R}_{GT}(\hat{Z}) - \hat{\mathcal{R}}(\hat{Z})}{\mathcal{R}_{GT}(\hat{Z})} \right)^2}$$

評価対象の推定量  
(Naive, IPS, DR)

32個の異なる推薦モデルの集合



## 実験結果: Yahoo! R3

DRが最も良い推定精度・全体的に@Kが大きい方が推定が簡単

真のランキング性能

比較  
推定量

	Recall@5	Recall@10	Recall@50
<b>Naive estimator</b> (baseline)	0.615	0.442	0.207
<b>IPS estimator</b> (Yang et al., RecSys'18)	0.473	0.308	0.158
<b>DR estimator</b> (ours)	<u>0.397</u>	<u>0.261</u>	<u>0.101</u>

## 実験結果: Coat

DRが最も良い推定精度・全体的に@Kが大きい方が推定が簡単

真のランキング性能

比較  
推定量

	Recall@5	Recall@10	Recall@50
<b>Naive estimator</b> (baseline)	0.617	0.387	0.184
<b>IPS estimator</b> (Yang et al., RecSys'18)	0.605	0.374	0.181
<b>DR estimator</b> (ours)	0.599	<u>0.318</u>	<u>0.118</u>

## 全体のまとめ

---

- 2段階feedbackが得られているならば、conversionが発生しやすいデータを上位に並べられているかを評価する方がKPIに対して直接的な評価ができる
- すぐに思いつく or 既存の推定量は、selection biasに対応できていなかったり、分散の問題に直面(推薦において顕著)
- clickが発生しなかったがためにconversionが観測されないデータのCVRを推定値で補完することで分散を改善した推定量を提案、正確なランキング性能のオフライン評価を可能に

## その他

---

### 論文では今日省いた点も書いているでチェックしてみてください

- DRに必要なCVRの推定をデータから得る方法
- semi-syntheticデータを用いた詳細な実験評価
  - sparsityがひどい時や分散改善の十分条件を満たしていない場合の評価
- 推定量の分散に加えて、安定性の議論
  - DRの方がIPSに比べて大きな失敗をする確率が小さい
- 査読結果は、4(meta)/3/2/4/4でAccept。2点の人は勘違い気味

# IR Reading2020秋にて今回の話の完全verで招待講演します



ABOUT ▼

お知らせ ▼

カレンダー

## IR Reading 2020秋（オンライン）開催案内

IR Reading 2020Fall(Online)

2020年10月31日

2020年8月14日に投稿

#news #イベント

普段の研究スタイルについても話して欲しいと頼まれている..

### 概要

情報アクセス分野に関する主要国際会議（SIGIRやWSDMなど）の論文読み合わせを行うIR Readingをオンラインにて開催します。多人数で協力しあって論文を紹介しあうことで、論文への理解を深めたり、分野全体のトレンドを掴んだりする場となれば幸いです。これまでの勉強会の内容につきましては、[過去のページ](#)をご覧ください。

また、今回のIR Readingでは SIGIR 2020, RecSys 2020, WSDM 2020 など多くのトップカンファレンスでご活躍中の東京工業大学の齋藤優太さんにご講演いただく予定となっております！

どなたでも無料で参加いただけますので、是非下記から参加登録ください。

日時

[https://sigir.jp/post/2020-10-31-irreading\\_2020fall/](https://sigir.jp/post/2020-10-31-irreading_2020fall/)

# Thank you!



email: [saito.y.bj@m.titech.ac.jp](mailto:saito.y.bj@m.titech.ac.jp)

paper: <https://dl.acm.org/doi/abs/10.1145/3383313.3412262>

code: <https://github.com/usaito/dr-ranking-metric>