Doubly Robust Estimator for Ranking Metrics with Post-Click Conversions

YUTA SAITO, Tokyo Institute of Technology

Post-click conversion, a pre-defined action on a web service after a click, is an essential form of feedback, as it directly contributes to the final revenue and accurately captures user preferences for items, compared with the ambiguous click. However, naively using post-click conversions can lead to severe bias when learning or evaluating recommenders because of the *selection bias* between clicked and unclicked data. In this study, we address the offline evaluation problem of algorithmic recommendations with biased post-click conversions. A possible solution to address this bias is to use the *inverse propensity score* estimator, as it can provide an unbiased evaluation even with the selection bias. However, this estimator is known to be subject to variance and instability problems, which can be severe in the recommendation setting, as feedback is often highly sparse. To address these limitations with the previous unbiased estimator, we propose a *doubly robust* estimator for the ground-truth ranking performance of a given recommender. The proposed estimator is unbiased against the ground-truth ranking metric and improves the variance and estimation error tail bound of the existing unbiased estimator. Finally, to evaluate the empirical efficacy of the proposed estimator, we conduct empirical evaluations using semi-synthetic and two public real-world datasets. The results show that the proposed metric reveals a better model evaluation performance compared with existing baseline metrics, particularly in a situation with severe selection bias.

CCS Concepts: • **Information systems** → **Collaborative filtering**; • **Computing methodologies** → *Learning from implicit feedback*.

Additional Key Words and Phrases: post-click conversions, ranking metrics, selection bias, doubly robust, inverse propensity score.

ACM Reference Format:
Yuta Saito. 2020. Doubly Robust Estimator for Ranking Metrics with Post-Click Conversions. In *Fourteenth ACM Conference on Recommender Systems (RecSys '20), September 21–26, 2020, Virtual Event, Brazil.* ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/3383313.3412262

1 INTRODUCTION

Offline evaluation of algorithmic recommendations is a critical task in a wide range of recommendation settings. A reliable offline metric aids the selection of optimal recommenders without deploying it to an online environment [7, 8]. In addition to industrial settings, researchers in academia often evaluate recommendation algorithms with offline datasets, because online A/B tests are considerably expensive and irreproducible [31].

These offline evaluations are often conducted using *implicit feedback*, such as click signals, as this is the most prevalent form of feedback collected in the natural behaviors of users in recommender systems. Evaluation with implicit feedback is often performed with the assumption that a click indicates a positive preference, whereas no click is a negative indicator. This simplified assumption, however, can result in misinterpretations of user preferences, because implicit feedback is often logged before item consumption and therefore might be a reflection of only the first impression of a user [28]. In contrast, in e-commerce applications, *post-click conversions*, such as the purchase of an item after a click, are receiving significant attention, because they directly contribute to the gross merchandise volume and are

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
© 2020 Association for Computing Machinery.
Manuscript submitted to ACM

considered as strong preference signals [14, 28, 30]. Therefore, it is advantageous to use post-click conversion to evaluate recommender systems compared with ambiguous implicit feedback. The drawback is that post-click conversions are observable for user-item pairs with a click, and those of pairs without a click are missing. Generally, the missing mechanism of the conversion data is **biased**, i.e., the distributions of the clicked and unclicked data are significantly different, as shown in Figure 1 (a). It is essential to address this bias when using conversion data for the evaluation of algorithmic recommendations.

Previous Work. To evaluate a recommender with reliable post-click conversions, one might naively use only observed post-click conversions for the evaluation. However, this type of naive estimation ignores the distributional difference between clicked and unclicked events and can lead to a problematic bias in the evaluation of a recommender. To alleviate this bias, the *inverse propensity score* (IPS) estimator has been proposed in a fully implicit feedback setting and can easily be extended to our post-click conversion setting [31]. The IPS estimator provides an unbiased estimation of the ground-truth performance; however, several previous studies indicate that the variance of this estimator can be significant [5, 18]. Moreover, this instability problem is critical in recommendation settings because of the severe data sparsity problem; only a small fraction of post-click conversions are observable among all user-item pairs [15, 29, 32].

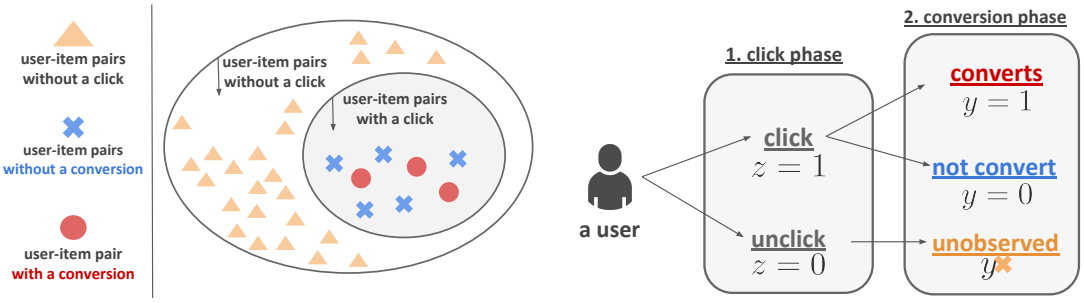
Contributions. To address the shortcomings of the previous unbiased estimator, we develop a *doubly robust* (DR) estimator for the ranking performance of a recommender using biased post-click conversions. The DR estimator is used to evaluate contextual bandit policies offline [5, 6, 10] and is considered desirable in these domains. This is because it improves the stability of the IPS estimator by combining outcome regression and propensity weighting in a theoretically sophisticated manner. However, applications of the DR method for the estimation of ranking performance in offline evaluations of algorithmic recommendations have not yet been investigated. In a theoretical analysis, we prove that the proposed DR estimator satisfies the unbiasedness for the ground-truth metric and improves the variance and estimation error tail bound over the IPS estimator under reasonable assumptions.

Finally, we perform comprehensive empirical comparisons using semi-synthetic and real-world datasets. The results demonstrate that the proposed estimator is more efficient compared with existing estimators in the offline model evaluation task, especially in a situation when observed conversions are highly sparse and biased.

Our contributions are summarized as follows.

- We formulate the evaluation of a recommender with post-click conversions as a statistical estimation problem and theoretically demonstrate that the IPS estimator for the ranking metrics [31] can be affected by the variance and instability problems.
- We propose a DR estimator for the ground-truth ranking performance and present its advantages over previous estimators with a theoretical analysis.
- We conduct extensive experiments with semi-synthetic and real-world datasets. The results demonstrate that the proposed estimator significantly outperforms the other baseline metrics in terms of model evaluation, particularly when observed post-click conversions are highly sparse and biased.

These theoretical and empirical findings provide practitioners with guidelines on how to conduct model selection, model evaluation, and hyperparameter tuning of their recommender systems in an offline environment using biased post-click conversions.



(a) selection bias problem (b) user behavior pattern

Fig. 1. (a) Illustration of the *selection bias* problem of the post-click conversion. User–item pairs with a click ($z = 1$) are not a representative population of the entire inference space, which is composed of all user–item pairs. (b) The sequential pattern (*click* $\rightarrow$ *conversion*) of a user's behavior under the post-click conversion setting. When the user clicks an item ($z = 1$), then either the positive ($y = 1$) or negative ($y = 0$) conversion outcome is observed. In contrast, when he/she does not click an item, y is unobservable.

2 PROBLEM SETTING

In this section, we describe the problem setting and objective of this study.

2.1 Notation

Let $u \in \mathcal{U}$ be a user and $i \in \mathcal{I}$ be an item; $\mathcal{D} = \mathcal{U} \times \mathcal{I}$ is defined as all user–item pairs. Now, we introduce two types of binary random variables to formulate a recommendation with post-click conversions. First, $z_{u,i}$ is a click indicator that takes 1 if user u clicks item i and 0 otherwise. Moreover, $y_{u,i}$ is a conversion indicator observed after a click and is assigned 1 if user u takes a pre-defined **positive** action, such as the purchase of item i after clicking the item. In contrast, it takes 0 if user u does not perform that action after the click. We also use $p^{ctr}_{u,i} = \mathbb{E}[z_{u,i}] = \mathbb{P}(z_{u,i} = 1)$ and $p^{cvr}_{u,i} = \mathbb{E}[y_{u,i}] = \mathbb{P}(y_{u,i} = 1)$ as the expectations of $z_{u,i}$ and $y_{u,i}$, respectively.[1]

In this study, we make the following assumptions to characterize a post-click conversion setting.

Assumption 2.1. *Conversion indicator ($y_{u,i}$) for a user–item pair is observable if and only if the item is clicked by the user.*

$$z_{u,i} = 1 \Leftrightarrow y_{u,i} \text{ is observed}, \ \forall (u, i) \in \mathcal{D}$$

This assumption implies that a post-click conversion cannot be observed without a click. This is consistent with real-world e-commerce recommender systems, such as Amazon and Etsy. In these platforms, conversions of an item cannot be observed without a click on the web page of the corresponding item, as depicted in Figure 1 (b).

Assumption 2.2. *The probability of observing a click is not uniform among user–item pairs, i.e.,*

$$p^{ctr}_{u,i} \neq p^{ctr}_{u',i'}, \ \exists (u, i), (u', i') \in \mathcal{D}$$

This assumption introduces differences in the distributions of clicked and unclicked events, i.e., the *selection bias* depicted in Figure 1 (a). This type of bias is observed in many public recommendation datasets, for example, in the form of *popularity bias*, where the number of observed interactions of items is significantly different [19, 31].

[1]CTR denotes *click through rate*, and CVR represents the *conversion rate*.

The following technical assumptions have a connection with the standard assumptions in causal inference [9, 16, 17].

Assumption 2.3. *Click ($z_{u,i}$) and conversion ($y_{u,i}$) indicators of any user–item pair depend only on the pair, i.e., $z_{u,i} \perp z_{u',i'},\ \ y_{u,i} \perp y_{u',i'},\ \forall (u,i) \neq (u',i')$. Moreover, click and conversion are independent conditionals for a user–item pair, i.e., $z_{u,i} \perp y_{u,i},\ \forall (u,i) \neq (u',i')$.*

Assumption 2.4. *For any user–item pair, the probability of observing a click is non-zero, i.e., $p^{ctr}_{u,i} \in (0,1],\ \forall (u,i) \in \mathcal{D}$.*

2.2 Ground-truth Performance of a Recommender (Estimation Target)

Here, we describe the objective of this study. In collaborative filtering with implicit feedback, additive ranking metrics, such as *mean average precision*, *discounted cumulative gain* (DCG), and *recall* are often used to evaluate a recommender. However, in our setting, the ranking metrics are preferably defined using post-click conversions. This is because the objective of making recommendations is to recommend lists of items with high preference levels for each user with the goal of maximizing user satisfaction and revenue, and conversions are reliable signals for them.

Therefore, we define the ground-truth performance of a given recommender in a post-click conversion setting as follows.

Definition 2.1. (Ground-truth Ranking Performance) The ground-truth ranking performance of a recommender in a post-click conversion setting is defined as

$$\mathcal{R}_{GT}(\widehat{Z}) = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}} \underbrace{p^{cvr}_{u,i}}_{\text{preference level}} \cdot c(\hat{Z}_{u,i}) \tag{1}$$

where $\hat{\boldsymbol{Z}} = \{Z_{u,i}\}_{(u,i) \in \mathcal{D}}$ is a set of the rankings of items for users, which is the output of a recommender to be evaluated. $c(\cdot)$ is a function that characterizes the ranking metrics. For example, DCG and recall can be represented as

$$DCG@K = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}} p^{cvr}_{u,i} \cdot \frac{\mathbb{I}\{\widehat{Z}_{u,i} \leq K\}}{\log_2(\widehat{Z}_{u,i} + 1)}, \quad Recall@K = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}} p^{cvr}_{u,i} \cdot \mathbb{I}\{\widehat{Z}_{u,i} \leq K\}.$$

The problem here is that we can observe the post-click conversions only for clicked events and not for unclicked events. Therefore, we cannot calculate the ground-truth ranking performance of a recommender directly. **It is essential to estimate it using only the observable information to accurately evaluate the performance of algorithmic recommenders offline.**

3 PREVIOUS ESTIMATORS

In this section, we summarize the previous estimators for the ground-truth performance and analyze their limitations.

3.1 Naive Estimator

The naive estimator is the most basic estimator for the ground-truth performance with post-click conversions and is often used to evaluate recommendation algorithms in experiments (for example, the post-click aware metric used in [28] has the same structure as the naive estimator).

Definition 3.1. (Naive Estimator) The naive estimator for the ground-truth performance of a recommender is defined as follows.

$$\widehat{\mathcal{R}}_{Naive}(\widehat{Z}) = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}} z_{u,i} y_{u,i} c(\hat{Z}_{u,i}) = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}: z_{u,i}=1} y_{u,i} c(\hat{Z}_{u,i}) \tag{2}$$

where $\hat{Z}$ is the output of a recommender to be evaluated.

The naive estimator is the averaged value of the ranking function among the clicked events (the summation is taken over the data with a click, i.e., $z_{u,i} = 1$). This estimator is intuitive and easily usable. However, as shown in [31], this naive estimator is biased against the ground-truth ranking metrics in Eq. (1), i.e., for some given $\widehat{Z}$,

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

The bias of this naive estimator is owing to the fact that it ignores the distributional shift between the clicked ($z_{u,i} = 1$) and unclicked ($z_{u,i} = 0$) data. Therefore, using the naive estimator for the evaluation of a recommender might lead to a sub-optimal model selection or hyperparameter tuning; one should rely on the estimators addressing this bias as an alternative to using the naive one.

3.2 Inverse Propensity Score (IPS) Estimator

To address the bias of the naive estimator, [31] applied the IPS estimator, a well-established estimator in off-policy evaluation of bandit policies and causal inference [5, 9, 10, 16], to the estimation of the ground-truth ranking performance in a fully implicit feedback setting. This estimator can be extended to our post-click conversion setting in a straightforward manner and is defined as follows.

Definition 3.2. (IPS Estimator) When the set of true CTRs and scoring set $\hat{Z}$ are given, the IPS estimator for the ground-truth performance of a recommender is defined as follows.

$$\widehat{\mathcal{R}}_{IPS}(\widehat{Z}) = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}} \frac{z_{u,i} y_{u,i}}{p^{ctr}_{u,i}} c(\hat{Z}_{u,i}) = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}: z_{u,i}=1} \frac{y_{u,i}}{p^{ctr}_{u,i}} c(\hat{Z}_{u,i}) \tag{3}$$

This IPS estimator is proven to be statistically unbiased against the ground-truth performance, i.e., for any given $\hat{Z}$,

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

The unbiasedness of the IPS estimator is desirable to evaluate the performance of recommenders offline. However, the variance of the IPS estimator in the recommendation setting has not yet been analyzed, despite the fact that this type of inverse propensity weighting estimator is often subject to the severe variance issue [5, 18, 19]. Thus, we derive the variance of the IPS estimator as follows.

Theorem 3.3. *(Variance of the IPS estimator) When the set of true CTRs and scoring set $\hat{Z}$ are given, the variance of the IPS estimator is*

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

We also derive the estimation error tail bound of the IPS estimator, which is an important statistical property of an estimator that characterizes its stability.

Proposition 3.4. *(Estimation Error Tail Bound of the IPS estimator) When the set of true CTRs and scoring set $\hat{Z}$ are given, the following inequality holds with a probability of at least $1-\delta(\delta \in (0,1))$*

$$\left|\widehat{\mathcal{R}}_{IPS}(\widehat{Z}) - \mathcal{R}_{GT}(\widehat{Z})\right| \leq \frac{1}{|\mathcal{U}|}\sqrt{\frac{\log(2/\delta)}{2}}\sqrt{\sum_{u\in\mathcal{U}}\sum_{i\in\mathcal{I}}\left(\frac{c(\hat{Z}_{u,i})}{p^{ctr}_{u,i}}\right)^2}$$

As shown in Theorem 3.3 and Proposition 3.4, both the variance and the estimation error tail bound of the IPS estimator depend on the inverse of the CTR. In the recommendation setting, the observed feedback data is often highly sparse, and the CTR is expected to be considerably small [15, 26, 29, 31, 32]. Therefore, the variance and instability of the IPS estimator can be critical in the evaluation of recommendation algorithms. These problems emerge, because the IPS estimator uses only the information of user–item pairs with a click (i.e., $z_{u,i} = 1$) and discards the information of user–item pairs without a click ($z_{u,i} = 0$). It is essential to develop an estimator that addresses the variance and instability problems of the IPS estimator while retaining its unbiasedness.

4 PROPOSED ESTIMATOR

In this section, we propose a novel estimator for the ground-truth ranking performance and show its advantages over existing estimators. Note that we omit the proofs owing to space constraints and present them in a complete version.[2]

4.1 Doubly Robust (DR) Estimator

We propose a DR estimator for the ground-truth ranking performance of a recommender. This estimator builds on the idea of using an estimator of a CVR for missing data (i.e., $z_{u,i} = 0$). Therefore, the estimator is more stable compared with the IPS estimator, which does not use any information of user-item pairs without a click. Moreover, our estimator is robust to the performance of the CVR estimator in use, and it is more desirable than the IPS estimator, even if the estimated CVR is not accurate. We define the DR estimator below.

Definition 4.1. (DR Estimator) Given the set of true CTRs, the DR estimator for the ground-truth performance of a recommender is defined as follows.

$$\widehat{\mathcal{R}}_{DR}(\widehat{Z}) = \frac{1}{|\mathcal{U}|}\sum_{u\in\mathcal{U}}\sum_{i\in\mathcal{I}}\left(\frac{z_{u,i}}{p^{ctr}_{u,i}}\left(y_{u,i} - \hat{p}^{cvr}_{u,i}\right) + \hat{p}^{cvr}_{u,i}\right)c(\hat{Z}_{u,i}) \tag{4}$$

where $\hat{p}^{cvr}_{u,i}$ is the estimated value for $p^{cvr}_{u,i}$. We will later discuss the method used for this estimation.

Initially, we show that the DR estimator is unbiased against the ground-truth performance.

Proposition 4.2. *(Unbiasedness of the DR estimator) Given the set of true CTRs, the DR estimator is statistically unbiased against the ground-truth performance, i.e., for any given $\hat{Z}$*

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

where the expectation is taken over the $z_{u,i}$ and $y_{u,i}$.

Furthermore, we derive the variance of the DR estimator.

[2]The complete version of this paper is available at ***https://usaito.github.io/files/RecSys2020_DRMetric.pdf***

Theorem 4.3. *(Variance of the DR estimator) Under the same conditions as in Theorem 3.3, the variance of the DR estimator is given by*

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

Note that the second term of the variance of the DR estimator can be negative. We also derive the estimation error tail bound of the DR estimator.

Proposition 4.4. *(Estimation Error Tail Bound of the DR Estimator) Given the set of true CTRs, realized conversions, and a scoring set $\hat{Z}$, the following inequality holds with a probability of at least $1-\delta(\delta \in (0,1))$*

$$\left|\widehat{\mathcal{R}}_{DR}(\widehat{Z}) - \mathcal{R}_{GT}(\widehat{Z})\right| \leq \frac{1}{|\mathcal{U}|}\sqrt{\frac{\log(2/\delta)}{2}}\sqrt{\sum_{u\in\mathcal{U}}\sum_{i\in\mathcal{I}}\left(\frac{\max\{1-\hat{p}^{cvr}_{u,i}, \hat{p}^{cvr}_{u,i}\}}{p^{ctr}_{u,i}}c(\hat{Z}_{u,i})\right)^2}$$

Using these derivations, we show that the DR estimator improves the variance of the IPS estimator under suitable conditions and consistently improves the estimation error bound compared with the IPS estimator.

Corollary 4.5. *(Smaller Variance) Suppose the following condition is satisfied for all user–item pairs,*

$$0 \leq \hat{p}^{cvr}_{u,i} \leq 2p^{cvr}_{u,i} \tag{5}$$

Then, the variance of the DR estimator is smaller than that of the IPS estimator, i.e., $\mathbb{V}\big(\widehat{\mathcal{R}}_{DR}(\widehat{Z})\big) \leq \mathbb{V}\big(\widehat{\mathcal{R}}_{IPS}(\widehat{Z})\big)$.

Corollary 4.6. *(Tighter Estimation Error Tail Bound) Given a fixed confidence delta ($\delta \in (0,1)$), the DR estimator achieves a tighter estimation error tail bound than the IPS estimator.*

We can rewrite the condition for the variance reduction of the DR estimator in Eq. (5) as

$$|\hat{p}^{cvr}_{u,i} - p^{cvr}_{u,i}| \leq p^{cvr}_{u,i}$$

Therefore, the condition only requires that the estimated error of $\hat{p}^{cvr}_{u,i}$ should be smaller than the original value of $p^{cvr}_{u,i}$ itself. For example, if $p^{cvr}_{u,i} \geq 0.5$, any estimated value $\hat{p}^{cvr}_{u,i}$ in $[0,1]$ satisfies this condition. Therefore, it is easy to satisfy this condition for the variance reduction.

Corollary 4.5 and Corollary 4.6 suggest that the DR estimator reduces the variance and increases the stability using the estimated CVR of user-item pairs with missing conversions. Note that Eq. (5) is a **sufficient** condition for a variance reduction of the DR estimator. We show that the DR estimator outperforms the Naive and IPS estimators, even if the condition for the variance reduction is not perfectly satisfied in the experiments.

4.2 Model Evaluation Procedure with the DR Estimator

We describe methods to derive the estimations of the CTR and CVR necessary to develop the proposed DR estimator in practice.

First, we estimate CTR by minimizing the following binary cross-entropy loss.

$$\sum_{u\in\mathcal{U}}\sum_{i\in\mathcal{I}} z_{u,i}\log(\hat{p}^{ctr}_{u,i}) + (1-z_{u,i})\log(1-\hat{p}^{ctr}_{u,i}) \tag{6}$$

Algorithm 1 Model Evaluation Procedure with DR Estimator

Input: observed feedback data $\mathcal{D}_{CT} = \{z_{u,i}\}_{(u,i)\in\mathcal{D}}$, $\mathcal{D}_{CV} = \{y_{u,i} \mid z_{u,i} = 1\}_{(u,i)\in\mathcal{D}}$, an arbitrary recommendation algorithm that is to be evaluated $\mathcal{A}$, a function characterizing a ranking metric $c(\cdot)$.
Output: Estimated performance of a given recommender $\mathcal{A}$.
1: Divide the feedback data $\mathcal{D}_{CT}$ and $\mathcal{D}_{CV}$ into training and validation sets.
2: Train the given recommender $\mathcal{A}$ using the training data
3: Predict the ranking scores ($\hat{Z}$) for the validation data by $\mathcal{A}$
4: Estimate CTR by minimizing Eq. (6) using the validation data
5: Estimate CVR by minimizing Eq. (8) using the validation data
6: Estimate the performance of a given recommender by the DR estimator in Eq. (4)
7: **return** $\widehat{\mathcal{R}}_{DR}(\widehat{Z})$

The minimization of this loss function is feasible, as we can observe click indicators $z_{u,i}$ for all user–item pairs. Second, the estimation of the CVR parameter should ideally be conducted by minimizing the following binary cross-entropy loss.

$$\sum_{u\in\mathcal{U}}\sum_{i\in\mathcal{I}} y_{u,i}\log(\hat{p}^{cvr}_{u,i}) + (1-y_{u,i})\log(1-\hat{p}^{cvr}_{u,i}) \tag{7}$$

However, the minimization of this ideal loss function is infeasible, as we cannot observe post-click feedback labels for unclicked data. Therefore, we suggest minimizing the following IPS estimator as an alternative solution.

$$\sum_{u\in\mathcal{U}}\sum_{i\in\mathcal{I}} \frac{z_{u,i}}{p^{ctr}_{u,i}}\left(y_{u,i}\log(\hat{p}^{cvr}_{u,i}) + (1-y_{u,i})\log(1-\hat{p}^{cvr}_{u,i})\right) \tag{8}$$

The minimization of this IPS estimator is feasible using only the observable data and satisfies the unbiasedness for the ideal loss function [20]. We have $\mathbb{E}[Eq.\ (8)] = Eq.\ (7)$, for any given $\hat{p}^{cvr}_{u,i}$.

It should be noted that this loss function can also suffer from a large variance caused by the data sparsity issue. However, the variance issue in the CVR estimation is not critical compared with that of the IPS estimator discussed in Section 3.2. This is because we use the estimated CVRs to construct the DR estimator in Eq. (4) and not to estimate the ground-truth metric in Eq. (1). Moreover, the proposed DR estimator is robust with regards to variance and instability issues and the CVR estimation is required to satisfy a mild condition in Eq. (5). Therefore, even if the CVR estimation does not perform well here, the resulting DR estimator is expected to be more accurate than the IPS estimator. We demonstrate the connection between the performance of the DR estimator and the accuracy of CVR estimations in the next section.

Algorithm 1 describes the whole model evaluation procedure with the proposed DR estimator.

5 EMPIRICAL EVALUATIONS

In this section, we empirically demonstrate the advantages of the proposed DR estimator.[3]

5.1 Experimental Setup

Datasets and preprocessing. We used the following three public datasets.

[3]Our code is available at ***https://github.com/usaito/dr-ranking-metric***

- **MovieLens (ML) 100K**[4]: This dataset contains 100,000 five-star movie ratings from 943 users and 1,682 movies collected on a movie recommendation service, and the ratings are missing-not-at-random. We used this dataset for the semi-synthetic experiment.
- **Coat**[5]: This dataset contains 6,500 five-star user-coat ratings from 290 Amazon mechanical turk workers on an inventory of 300 coats. The training data have been collected via self-selections by the workers. In contrast, the test data have been collected by asking the workers to rate 16 randomly selected coats.
- **Yahoo! R3**[6]: The training data contain approximately 300,000 five-star ratings from 15,400 users and 1,000 songs, and the test data have been collected by asking a subset of 5,400 users to rate 10 randomly selected songs. Following a previous work [31], we filtered the users who have at least one positive and one negative song in the test set and those with two positive songs in the training set (2,296 users satisfy these conditions). We also held out a validation set from the training set by randomly sampling 30 songs for each user.

For **ML 100K**, we employed the following preprocessing procedure to create a biased post-click conversion setting.

(1) Predict the ratings and the probability of a rating being observed for user—item pairs by using standard matrix factorization [13]. We denoted the prediction for the rating of (u, i) as $\hat{R}_{u,i}$ and that for the probability of a rating of (u, i) being observed as $\hat{O}_{u,i}$. This allows ground-truth evaluation against fully known CTR and CVR matrices.
(2) Transform predicted ratings $\hat{R}$ into CVR using the methodology used in the information retrieval domain [1, 2], as follows.

$$p_{u,i}^{cvr} = \epsilon + (1-\epsilon)\frac{2^{\hat{R}_{u,i}} - 1}{2^{\hat{R}_{max}} - 1},\ \forall (u,i) \in \mathcal{D}$$

where $\hat{R}_{u,i}$ is a predicted rating for (u, i) and $\hat{R}_{max}$ is the maximum predicted rating, which was 5 in our case. $\epsilon \in [0, 1]$ controls the noise level, and we set $\epsilon = 0.1$.
(3) Transform predicted observation probabilities $\hat{O}$ into CTR using the following procedure.

$$p_{u,i}^{ctr} = \left(\hat{O}_{u,i}\right)^{power},\ \forall (u,i) \in \mathcal{D}$$

where $power \geq 0$ controls the level of data sparsity; a larger value of $power$ introduces a severe sparsity in the semi-synthetic data. We used different values of $power$ in the experiment.
(4) Sample binary click and conversion variables by the following Bernoulli sampling

$$z_{u,i} \sim Bern(p_{u,i}^{ctr}),\ y_{u,i} \sim Bern(p_{u,i}^{cvr}),\ \forall (u,i) \in \mathcal{D}$$

where $Bern(\cdot)$ is the Bernoulli distribution.
(5) Derive the observed post-click conversions using the above variables as $\{(u, i, y_{u,i}) \mid z_{u,i} = 1\}$.

For **Coat and Yahoo! R3**, we employed the following procedure.

(1) Define conversion variable $y_{u,i}$ for a user–item pair as follows:

$$y_{u,i} = \begin{cases} 1 & \text{(if item } i \text{ is rated greater than or equal to 4 by user } u\text{)} \\ 0 & \text{(otherwise)} \end{cases}$$

[4]https://grouplens.org/datasets/movielens/100k/
[5]https://www.cs.cornell.edu/ schnabts/mnar/
[6]http://webscope.sandbox.yahoo.com/

(2) Define click variable $z_{u,i}$ for a user–item pair as follows:

$$z_{u,i} = \begin{cases} 1 & \text{(if item } i \text{ is rated by user } u\text{)} \\ 0 & \text{(otherwise)} \end{cases}$$

(3) Derive the observed post-click conversion data using the above variables as $\{(u, i, y_{u,i}) \mid z_{u,i} = 1\}$. Note that post-click conversions of user–item pairs without a click are unavailable in our formulation and cannot be used to evaluate recommenders.

We used Coat and Yahoo! R3 for the real-world experiment, as they contain training and test sets with different user-item distributions. Moreover, they are explicit feedback datasets and can consequently utilize ground-truth user preference information. Therefore, these two datasets are appropriate for the evaluation of recommenders with biased post-click conversions. To the best of our knowledge, they are the only datasets that satisfy these properties.

Table 1. Used values for each hyperparameter when constructing candidate recommendation models with LightFM.

hyperparameters	used values
no_components	$\{5, 20, 50, 100\}$
user & item alpha	$\{10^{-2}, 10^{-4}\}$
loss	{'logistic', 'bpr', 'warp', 'warp-kos'}
learning_rate	10^{-4} (fixed)
epochs	100 (fixed)

Candidate recommendation models. To develop a set of candidate recommenders, we used LightFM[7] and different values of hyperparameters for **no_components**, **user_alpha & item_alpha**, and **loss**, as described in Table 1. The number of candidate recommenders was thus $4 \times 2 \times 4 = 32$. The default values were used for the other hyperparameters.

Compared estimators. We compared the following three estimators for the ground-truth ranking metrics:

- **Naive estimator**: This estimator uses only clicked data for model evaluation and is defined in Eq. (2).
- **IPS estimator [31]**: This estimator estimates the CTR as the propensity score and weighs the values of the function $c(\cdot)$ by the inverse propensity score of the corresponding user-item pairs. This estimator is defined in Eq. (3).
- **DR estimator**: This is our proposed estimator. It first estimates the CTR and then estimates the CVR based on the observable data. Finally, it combines the estimated CTR and CVR, as described in Eq. (4). The procedure to evaluate a recommender with the DR estimator can be found in Section 4.2.

For IPS and DR, we used the true $p^{ctr}_{u,i}$ for ML 100K to evaluate the pure effect of the data sparsity issue controlled by *power*. In contrast, we estimated it by logistic matrix factorization [12] for Coat and Yahoo! R3 in order to imitate real-world situations.

Next, for DR, we used the following predicted CVR for ML 100K:

$$\hat{p}^{cvr}_{u,i} = \max\{\min\{p^{cvr}_{u,i} + Unif(-bound, bound), 0\}, 1\}, \quad \forall (u, i) \in \mathcal{D}$$

[7] https://lyst.github.io/lightfm/docs/home.html

where $Unif(\cdot, \cdot)$ is a uniform distribution and $bound \geq 0$ controls the accuracy of the CVR estimation; a large value of $bound$ introduces a large estimation bias of CVR. Note that max and min operators simply scale $\hat{p}_{u,i}^{cvr}$ in the interval of $[0, 1]$. In contrast, we estimated it by logistic matrix factorization for Coat and Yahoo! R3 in order to imitate real-world situations.

Performance measures for estimators. We compared the performance of the three estimators by the *relative root mean-squared-error* (relative-RMSE) defined as follows:

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

where $\mathcal{M}$ is a set of outputs by the candidate recommenders, and $\widehat{\mathcal{R}}(\hat{Z})$ is one of the compared estimators. The relative-RMSE evaluates an estimators' ability to accurately estimate the performance of candidate recommenders (model evaluation performance).

For the ground-truth ranking metrics (i.e., $\mathcal{R}_{GT}(\hat{Z})$ in Eq. (1)), we used DCG and Recall in all experiments. The formal definitions of these ranking metrics can be found in Section 2.2.

Experimental procedure. For all datasets, we first split the original training set into 70% training and 30% validation sets. Then, we trained all 32 candidate recommenders using the training set and evaluated the ground-truth performance of the recommender using the test set. The performance measured using the test set is defined as the ground-truth ranking performance. Besides, we estimated the performance of all candidate recommenders using validation sets with three estimators (i.e., Naive, IPS, and DR). Finally, we evaluated the performance of the estimators by comparing the estimated performance of the estimators with the ground-truth ranking performance by *relative-RMSE*.

Table 2. Averaged and minimum CTRs with different values of ***power***

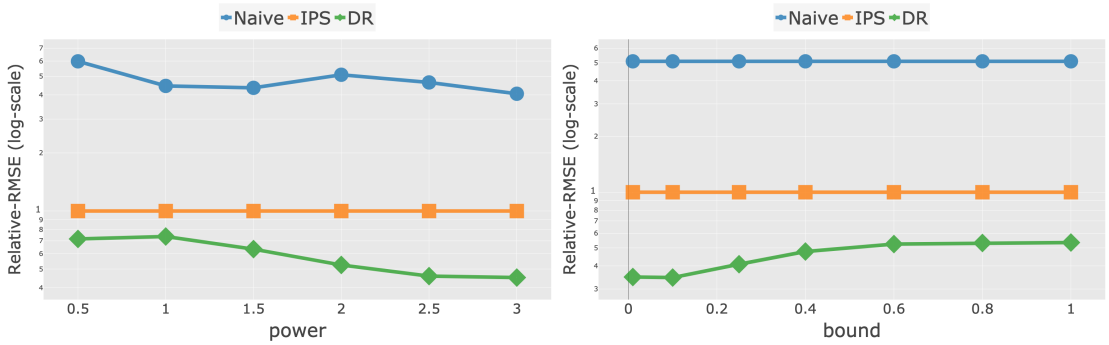
	Values of *power*						
	0.5	0.75	1.0	1.5	2.0	2.5	3.0
Average	0.218	0.113	0.063	0.023	0.010	5.024×10^{-3}	2.685×10^{-6}
Minimum	0.069	0.018	4.764×10^{-3}	3.288×10^{-4}	2.269×10^{-5}	1.566×10^{-6}	1.08×10^{-7}

Table 3. Percentage (PCT) of data satisfying the variance reduction condition in Eq. (5)

	Values of *bound*						
	0.01	0.1	0.25	0.4	0.6	0.8	1.0
PCT of data (%)	100.0	100.0	94.6	85.4	76.3	71.4	68.4

5.2 Results

Here we describe the results in the semi-synthetic and real-world experiments.



(a) Comparison of estimators with different *power* (b) Comparison of estimators with different *bound*

Fig. 2. Comparison of alternative estimators with ML 100K by *relative-RMSE* relative to that of IPS.

Note: The model evaluation performances of estimators with different values of *power* or *bound* averaged over 20 different random seeds are reported. (a) A larger *power* indicates a large selection bias. The result suggests that DR is effective, especially when data sparsity is severe. (b) A larger *bound* indicates a worse performance of the CVR estimator used in DR. The result demonstrates the robustness of DR with regards to the performance of the CVR estimator.

5.2.1 Results on the semi-synthetic experiment. First, we investigated how the significance of data sparsity affected the performance of the estimators. Figure 2 (a) shows the *relative-RMSE* of the estimators relative to that of the IPS estimator (log-scaled). We varied the value of *power* as follows: $\{0.5, 0.75, 1.0, 1.5, 2.0, 2.5, 3.0\}$, in contrast, $bound = 0.6$ here. A large value of *power* introduces severe data sparsity and variance problems, as shown in Table 2.

The result indicates that the proposed DR estimator significantly outperforms other estimators in all settings. Specifically, the performance gained by DR over IPS is 28.4% for $power = 0.5$; in contrast, it is 54.8% for $power = 5.0$. The result highlights the robustness of the proposed estimator with regard to the data sparsity issue.

Next, we investigated the effect of the accuracy of the CVR estimator on the performance of the DR estimator. Figure 2 (b) shows the *relative-RMSE* of estimators relative to that of the IPS estimator (log-scaled). We varied the value of *bound* as follows: $\{0.01, 0.1, 0.25, 0.4, 0.6, 0.8, 1.0\}$ in contrast, $power = 2.0$ here. A large value of *bound* leads to a worse performance of the CVR estimator used in DR (i.e., $\hat{p}^{cvr}_{u,i}$), as shown in Table 3. Note that the performances of naive and IPS are constant here, because they are independent of the values of *bound*.

The result indicates that the proposed DR estimator significantly outperforms the other estimators, even when the CVR estimator performs poorly. In particular, as suggested by our theoretical analysis, DR significantly outperforms IPS when the Eq. (5) is perfectly satisfied ($bound = 0.01$). In addition, DR still outperforms IPS by 46.5% when over 30% of the user-item pairs do not satisfy Eq. (5) ($bound = 1.0$). Thus, the performance of the DR estimator is robust with regards to the performance of the CVR estimator. Note that Eq. (5) is a **sufficient** condition for the variance reduction of DR, and the empirical result does not contradict our theoretical analysis.

5.2.2 Results on the real-world experiment. Finally, we compared the performance of the estimators using the real-world datasets. Table 4 presents the performances of estimators when DCG@K and Recall@K ($K \in \{5, 10, 50\}$) are used as the ground-truth ranking metrics.

The table shows that the proposed estimator outperforms the other baseline metrics in almost all cases. Specifically, it demonstrates the significant gains in the estimation of the ground-truth ranking performance for the Yahoo! R3 dataset.

Table 4. Comparison of *relative-RMSE* (model evaluation performances) of alternative estimators

Datasets	Estimators	DCG@K			Recall@K		
		$K = 5$	$K = 10$	$K = 50$	$K = 5$	$K = 10$	$K = 50$
Yahoo! R3	Naive	0.613 (± 0.070)	0.470 (± 0.057)	0.245 (± 0.027)	0.615 (± 0.067)	0.442 (± 0.047)	0.207 (± 0.017)
	IPS	0.767 (± 0.022)	0.780 (± 0.024)	0.850 (± 0.015)	0.473 (± 0.040)	0.308 (± 0.032)	0.158 (± 0.013)
	DR (ours)	**0.461** (± 0.053)	**0.316** (± 0.040)	**0.181** (± 0.022)	**0.397** (± 0.042)	**0.261** (± 0.029)	**0.101** (± 0.011)
Coat	Naive	0.666 (± 0.037)	0.430 (± 0.013)	0.208 (± 0.005)	0.617 (± 0.027)	0.387 (± 0.011)	0.184 (± 0.004)
	IPS	0.785 (± 0.020)	0.805 (± 0.010)	0.915 (± 0.004)	0.605 (± 0.028)	0.374 (± 0.011)	0.181 (± 0.004)
	DR (ours)	0.661 (± 0.066)	**0.359** (± 0.020)	**0.137** (± 0.004)	0.599 (± 0.050)	**0.318** (± 0.014)	**0.118** (± 0.003)

Note: The model evaluation performances averaged over 200 simulations for Coat and 30 simulations for Yahoo! R3. The standard errors (StdErr) are depicted in parentheses. The **bold font** is used for the best performance of each setting (only when the difference is significant given the standard errors.). The proposed DR estimator outperforms the other estimators in almost all cases.

For Coat, there is no difference between DR and Naive when $K = 5$ given their standard errors, but it again outperforms the optimal baseline by large margins for the other values of K. The results empirically demonstrate that the proposed evaluation procedure provides more accurate and stable performance estimations of the candidate recommenders than the existing procedures.

Summary of Empirical Findings. In the semi-synthetic experiment with ML-100k data, the advantages of the proposed estimator are strengthened when the data sparsity issue is severe, which is consistent with the theoretical analysis. Moreover, it can be shown that DR is robust to the performance of the CVR estimator and outperforms IPS even when some data do not satisfy the variance reduction condition. This result suggests its wide applicability.

In the real-world experiment with Yahoo! R3 and Coat datasets, the DR estimator significantly outperforms the baselines in the model evaluation task. Therefore, the results verify that it can help with estimation of the accurate performance of recommenders in a real-world, offline setting.

6 RELATED WORK

In this section, we review related literature.

6.1 Post-Click Conversion Modeling

Post-click conversion modeling is an essential task for building e-commerce recommender systems, as it directly contributes to the final revenue. To model post-click conversions, the selection bias and data sparsity problems have to be addressed. [15] proposed an entire space multi-task model for predicting CVR, which remedies the data sparsity problem. Further, [29] leveraged supervisory signals from users' post-click behaviors other than conversions to further alleviate the data sparsity problem. Moreover, a pioneering work that utilizes post-click feedback to build a better ranking algorithm is [28]. They first empirically showed that half of the click events resulted in skipping the content by using two large scale datasets and indicated that user behaviors after a click (post-click feedback) capture important preference signals that are not captured by only implicit feedback. Further, they proposed a modified version of pointwise and pairwise loss functions to train recommenders with post-click conversions. They did not, however, address the selection bias caused by the distributional shift between the clicked and unclicked events. The work that has the most relevance to ours in post-click conversion modeling is [32]. This work utilized DR estimator to address the selection

bias problem and construct a debiased version of the pointwise loss function, satisfying unbiasedness. However, their work focused on the prediction methods, and a theoretical analysis on the variance and estimation error tail bound of the proposed estimator was not performed. Thus, our work is the first to address the evaluation problem with post-click conversion and show the desirable variance and tighter estimation error tail bound properties of the DR estimator in this setting.

6.2 Counterfactual Estimation

Offline evaluation of bandit policies or recommendation algorithms using counterfactual estimators has been extensively researched, as online A/B tests require time and cost money [3, 5–8]. Most existing off-policy performance estimators build on the Direct Method (DM) [5] or the IPS estimation technique [9, 11, 16, 17, 25]. The DM models and predicts the reward distribution using logged bandit feedback with machine learning algorithms and uses the predicted values to estimate the performance of a given policy offline. In contrast, the IPS estimation technique utilizes the propensity score, which is the probability of selecting each item according to a past policy, to address the distributional shift. However, it is widely acknowledged that the DM estimator is subject to a large bias, and that the IPS estimator is subject to the variance problem [5, 19, 22, 23]. The DR estimator combines the DM and IPS in a theoretically sophisticade manner to address the bias and variance problems [5, 10]. Moreover, beyond the DR estimator, several ways of combining the DM and IPS approaches have been proposed, including the more robust doubly robust [6], CAB estimator [21], or SWITCH estimator [27] for pursuing a better bias-variance trade-off. In this work, we have extended the DR estimator to the evaluation of the ranking performances with post-click conversions for the first time.

7 CONCLUSION

In this study, we explored a method for establishing a ranking performance metric using the reliable, but biased post-click conversions. First, we formulated the evaluation of a recommender with post-click conversions as a statistical estimation problem. Next, we discussed the limitations of the existing estimators from a theoretical perspective. Subsequently, we proposed a DR estimator for the ground-truth ranking performance metrics. We also showed its unbiasedness and desired statistical properties, such as a lower variance and tighter estimation error tail bound. The empirical evaluations show that our estimator significantly outperformed existing estimators in the offline model evaluation task, particularly for post-click conversions that are highly sparse and biased.

As future work, we plan to develop a listwise offline learning-to-rank method with biased post-click conversions. Our estimator provides an accurate estimation of additive ranking metrics such as DCG, and thus it can enable the learning of an efficient ranker in the offline post-click conversion setting. Moreover, our estimator and theoretical analysis depend on the assumption that the click of each item is independent of those of the other items (Assumption 2.3). This assumption might not hold, for example, in a slate recommendation setting where multiple items are recommended simultaneously to users [4, 24]. Thus, extending our estimator to more general and realistic settings is one of the essential future research directions.

REFERENCE

[1] Qingyao Ai, Keping Bi, Cheng Luo, Jiafeng Guo, and W Bruce Croft. 2018. Unbiased learning to rank with unbiased propensity estimation. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. ACM, 385–394.

[2] Olivier Chapelle, Donald Metlzer, Ya Zhang, and Pierre Grinspan. 2009. Expected reciprocal rank for graded relevance. In *Proceedings of the 18th ACM conference on Information and knowledge management*. 621–630.

[3] Minmin Chen, Alex Beutel, Paul Covington, Sagar Jain, Francois Belletti, and Ed H Chi. 2019. Top-k off-policy correction for a REINFORCE recommender system. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*. ACM, 456–464.

[4] Maria Dimakopoulou, Nikos Vlassis, and Tony Jebara. 2019. Marginal Posterior Sampling for Slate Bandits.. In *IJCAI*. 2223–2229.

[5] Miroslav Dudík, John Langford, and Lihong Li. 2011. Doubly Robust Policy Evaluation and Learning. *CoRR* abs/1103.4601 (2011). arXiv:1103.4601 http://arxiv.org/abs/1103.4601

[6] Mehrdad Farajtabar, Yinlam Chow, and Mohammad Ghavamzadeh. 2018. More Robust Doubly Robust Off-policy Evaluation. In *International Conference on Machine Learning*. 1446–1455.

[7] Alexandre Gilotte, Clément Calauzènes, Thomas Nedelec, Alexandre Abraham, and Simon Dollé. 2018. Offline a/b testing for recommender systems. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*. ACM, 198–206.

[8] Alois Gruson, Praveen Chandar, Christophe Charbuillet, James McInerney, Samantha Hansen, Damien Tardieu, and Ben Carterette. 2019. Offline Evaluation to Make Decisions About PlaylistRecommendation Algorithms. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*. ACM, 420–428.

[9] Guido W Imbens and Donald B Rubin. 2015. *Causal inference in statistics, social, and biomedical sciences*. Cambridge University Press.

[10] Nan Jiang and Lihong Li. 2016. Doubly Robust Off-policy Value Evaluation for Reinforcement Learning. In *International Conference on Machine Learning*. 652–661.

[11] Thorsten Joachims, Adith Swaminathan, and Tobias Schnabel. 2017. Unbiased learning-to-rank with biased feedback. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*. ACM, 781–789.

[12] Christopher C Johnson. 2014. Logistic matrix factorization for implicit feedback data. *Advances in Neural Information Processing Systems* 27 (2014).

[13] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 8 (2009), 30–37.

[14] Hongyu Lu, Min Zhang, and Shaoping Ma. 2018. Between Clicks and Satisfaction: Study on Multi-Phase User Preferences and Satisfaction for Online News Reading. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. ACM, 435–444.

[15] Xiao Ma, Liqin Zhao, Guan Huang, Zhi Wang, Zelin Hu, Xiaoqiang Zhu, and Kun Gai. 2018. Entire space multi-task model: An effective approach for estimating post-click conversion rate. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. ACM, 1137–1140.

[16] Paul R Rosenbaum and Donald B Rubin. 1983. The central role of the propensity score in observational studies for causal effects. *Biometrika* 70, 1 (1983), 41–55.

[17] Donald B Rubin. 1974. Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of educational Psychology* 66, 5 (1974), 688.

[18] Yuta Saito, Hayato Sakata, and Kazuhide Nakata. 2019. Doubly Robust Prediction and Evaluation Methods Improve Uplift Modeling for Observational Data. In *Proceedings of the 2019 SIAM International Conference on Data Mining*. SIAM, 468–476.

[19] Yuta Saito, Suguru Yaginuma, Yuta Nishino, Hayato Sakata, and Kazuhide Nakata. 2020. Unbiased Recommender Learning from Missing-Not-At-Random Implicit Feedback. In *Proceedings of the 13th International Conference on Web Search and Data Mining*. 501–509.

[20] Tobias Schnabel, Adith Swaminathan, Ashudeep Singh, Navin Chandak, and Thorsten Joachims. 2016. Recommendations as Treatments: Debiasing Learning and Evaluation. In *Proceedings of The 33rd International Conference on Machine Learning (Proceedings of Machine Learning Research)*, Maria Florina Balcan and Kilian Q. Weinberger (Eds.), Vol. 48. PMLR, New York, New York, USA, 1670–1679. http://proceedings.mlr.press/v48/schnabel16.html

[21] Yi Su, Lequn Wang, Michele Santacatterina, and Thorsten Joachims. 2019. CAB: Continuous Adaptive Blending for Policy Evaluation and Learning. In *International Conference on Machine Learning*. 6005–6014.

[22] Adith Swaminathan and Thorsten Joachims. 2015. Counterfactual risk minimization: Learning from logged bandit feedback. In *International Conference on Machine Learning*. 814–823.

[23] Adith Swaminathan and Thorsten Joachims. 2015. The self-normalized estimator for counterfactual learning. In *advances in neural information processing systems*. 3231–3239.

[24] Adith Swaminathan, Akshay Krishnamurthy, Alekh Agarwal, Miro Dudik, John Langford, Damien Jose, and Imed Zitouni. 2017. Off-policy evaluation for slate recommendation. In *Advances in Neural Information Processing Systems*. 3632–3642.

[25] Xuanhui Wang, Nadav Golbandi, Michael Bendersky, Donald Metzler, and Marc Najork. 2018. Position bias estimation for unbiased learning to rank in personal search. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*. ACM, 610–618.

[26] Xiaojie Wang, Rui Zhang, Yu Sun, and Jianzhong Qi. 2019. Doubly Robust Joint Learning for Recommendation on Data Missing Not at Random. In *International Conference on Machine Learning*. 6638–6647.

[27] Yu-Xiang Wang, Alekh Agarwal, and Miroslav Dudik. 2017. Optimal and adaptive off-policy evaluation in contextual bandits. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 3589–3597.

[28] Hongyi Wen, Longqi Yang, and Deborah Estrin. 2019. Leveraging Post-click Feedback for Content Recommendations. In *Proceedings of the 13th ACM Conference on Recommender Systems* (Copenhagen, Denmark) *(RecSys '19)*. ACM, New York, NY, USA, 278–286.

[29] Hong Wen, Jing Zhang, Yuan Wang, Wentian Bao, Quan Lin, and Keping Yang. 2019. Conversion Rate Prediction via Post-Click Behaviour Modeling. *arXiv preprint arXiv:1910.07099* (2019).

[30] Liang Wu, Diane Hu, Liangjie Hong, and Huan Liu. 2018. Turning clicks into purchases: Revenue optimization for product search in e-commerce. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. ACM, 365–374.

[31] Longqi Yang, Yin Cui, Yuan Xuan, Chenyang Wang, Serge Belongie, and Deborah Estrin. 2018. Unbiased Offline Recommender Evaluation for Missing-not-at-random Implicit Feedback. In *Proceedings of the 12th ACM Conference on Recommender Systems* (Vancouver, British Columbia, Canada) *(RecSys '18)*. ACM, New York, NY, USA, 279–287. https://doi.org/10.1145/3240323.3240355

[32] Wenhao Zhang, Wentian Bao, Xiao-Yang Liu, Keping Yang, Quan Lin, Hong Wen, and Ramin Ramezani. 2019. A Causal Perspective to Unbiased Conversion Rate Estimation on Data Missing Not at Random. *arXiv preprint arXiv:1910.09337* (2019).