Batch-Adaptive Doubly Robust Learning for Debiasing Post-Click Conversion Rate Prediction Under Sparse Data

HANG PAN, University of Science and Technology of China, Hefei, China
CHUNYUAN ZHENG, Peking University, Beijing, China
WENJIE WANG, JINGANG JIANG, and XUEYING LI, University of Science and Technology of China, Hefei, China
HAOXUAN LI, Peking University, Beijing, China
FULI FENG, University of Science and Technology of China, Hefei, China

Post-Click Conversion Rate (CVR) prediction aims to predict the probability of a conversion event occurring after a user clicks. Most CVR prediction methods use clicked events to train models and subsequently predict on both clicked and unclicked events, facing selection bias. To unbiasedly predict CVR, doubly robust (DR) learning incorporates propensity score reweighting and missing data error imputation, but with suboptimal performance under sparse click events. We theoretically demonstrate that existing DR methods face high or even unbounded bias, variance, and generalization error bound under small propensity scores from sparse click events. This motivates us to propose a new Batch-Adaptive DR (BADR) Learning method. In particular, we propose a BADR estimator, which adaptively adjusts the influence of each data batch during debiasing CVR prediction model training based on the propensity scores within that batch. We prove that the BADR estimator has bounded bias, variance, and generalization error bound, all of which are smaller than those of the DR estimator under small propensity scores, while maintaining asymptotic double robustness, i.e., achieving double robustness under a large sample size. Furthermore, we approximate the variance of the BADR estimator and derive a new batch-adaptive imputation model training loss compatible with the BADR estimator, which theoretically ensures further variance reduction during training. Our experiments on real-world datasets validate BADR's effectiveness and rationality.

CCS Concepts: • **Information systems** → **Recommender systems**;

Additional Key Words and Phrases: Recommender systems, Post-Click Conversion Rate, Debias, Doubly Robust Learning

This work was supported by the National Natural Science Foundation of China (62272437, 623B2002) and by the National Key Research and Development Program of China (2022YFB3104701).

Authors' Contact Information: Hang Pan, University of Science and Technology of China, Hefei, China; e-mail: hungpaan@mail.ustc.edu.cn; Chunyuan Zheng, Peking University, Beijing, China; e-mail: cyzheng@stu.pku.edu.cn; Wenjie Wang, University of Science and Technology of China, Hefei, China; e-mail: wenjiewang96@gmail.com; Jingang Jiang, University of Science and Technology of China, Hefei, China; e-mail: jjg1210@mail.ustc.edu.cn; Xueying Li, University of Science and Technology of China, Hefei, China; e-mail: lxying@mail.ustc.edu.cn; Haoxuan Li (corresponding author), Peking University, Beijing, China; e-mail: hxli@stu.pku.edu.cn; Fuli Feng (corresponding author), University of Science and Technology of China, Hefei, China; e-mail: fulifeng93@gmail.com.

ACM 1558-2868/2026/2-ART56
https://doi.org/10.1145/3785146

ACM Reference format:
Hang Pan, Chunyuan Zheng, Wenjie Wang, Jingang Jiang, Xueying Li, Haoxuan Li, and Fuli Feng. 2026. Batch-Adaptive Doubly Robust Learning for Debiasing Post-Click Conversion Rate Prediction Under Sparse Data. *ACM Trans. Inf. Syst.* 44, 3, Article 56 (February 2026), 29 pages.
https://doi.org/10.1145/3785146

1 Introduction

The **Post-Click Conversion Rate (CVR)** represents the probability of a conversion occurring after a click event [56, 67]. It serves as a strong signal of the commercial performance of recommender systems in various applications, such as e-commerce, online advertising, and streaming media. As such, the task of CVR prediction plays a critical role in exploring user interests and increasing platform revenues.

The CVR prediction task can be seen as a missing data problem because we aim to predict CVR for all events, but conversion feedback is only observable in clicked events and is missing in unclicked events. This missingness is not random, since users are free to choose which items to click. As a result, the distributions of clicked events and all events differ significantly, which has driven CVR prediction research to focus on addressing selection bias [38, 52, 56, 57].

Existing methods debias CVR prediction based on unbiased estimators that estimate the loss of the CVR prediction model on all events [47, 57]. Among them, **Doubly Robust (DR)** learning methods [22, 57] achieve state-of-the-art performance for CVR prediction. DR combines **Inverse Propensity Score (IPS)** reweighting [47] and **Error Imputation-Based (EIB)** strategy [51]. IPS models the probability of a click (*aka.* propensity score) and reweights the prediction errors on click events with the IPSs, while EIB models CVR prediction errors and imputes them for unclicked events. DR has double robustness property, i.e., it is unbiased if either the learned propensity scores or the imputed errors are accurate for all user–item pairs.

Despite their double robustness, our theoretical analysis reveals that existing DR methods encounter high or even unbounded bias, variance, and generalization error bound when propensity scores are small due to sparse data in the CVR-prediction task. This weakens the debiasing in the training of the CVR prediction model, thereby limiting the reduction of its loss across all events and leading to suboptimal convergence and degraded model performance. We find that the root cause of the above issues is the formulation of the DR estimator[1] in the existing DR methods. This motivates us to explore a new DR learning method with a more robust DR estimator under sparse data, instead of only adjusting the training loss of the propensity model or the imputation model [5, 10, 69], which can reduce bias or variance under certain conditions.

To this end, we propose a new **Batch-Adaptive DR (BADR)** Learning method. In particular, we propose a principled BADR estimator that adaptively adjusts the influence of each data batch on debiasing CVR prediction, based on the propensity scores within that batch. Our theoretical analysis reveals that, in the presence of extremely small propensity scores, this new estimator ensures bounded bias, variance, and generalization error bound, all of which are smaller than those of the DR estimator in existing methods, while maintaining asymptotic double robustness, i.e., achieving double robustness under a large sample size. We show in the first remark in Section 3.3 that asymptotically DR and DR estimators are roughly equivalent for debiasing in practical applications. To further enhance algorithmic robustness, we theoretically derive a new batch-adaptive imputation model training loss that is compatible with the BADR estimator. This is achieved by approximating

[1]We exclude StableDR [29] because it does not use imputation errors to train the CVR prediction model, which limits its debiasing performance, as validated by our experiments. We aim to analyze the widely used DR estimator in existing methods, retaining imputation errors during CVR model training to achieve better debiasing performance.

the variance of the BADR estimator using the delta method [48], thereby theoretically ensuring further variance reduction during training. Extensive experiments conducted on real-world datasets demonstrate the effectiveness and rationality of BADR.

The contributions of this article can be summarized as follows.

— We propose a principled BADR estimator that adjusts the influence of each data batch during debiasing CVR prediction model training based on its propensity score distribution, achieving asymptotically unbiased CVR prediction under sparse data.
— We theoretically show that the BADR estimator can significantly reduce bias, variance, and generalization error bound in the presence of small propensity scores, while maintaining asymptotic double robustness.
— We approximate the variance of the BADR estimator using the delta method and theoretically derive a new batch-adaptive imputation model training loss that is compatible with the BADR estimator, further enhancing algorithmic robustness during training.
— We conduct extensive offline experiments on three real-world datasets, as well as online experiments in a large-scale e-commerce marketing scenario, thereby validating the rationality and effectiveness of the proposed BADR.

2 Preliminaries

In this section, we first formulate the problem of debiasing CVR prediction. Then, we introduce three widely used unbiased estimators that estimate the ideal loss: IPS, EIB, and DR learning. Finally, we present four representative enhanced DR methods that account for bias, variance, or both: MRDR [10], DR-BIAS [5], DR-MSE [5], GPL [69], and then show these methods still have high or unbounded bias, variance, and generalization error bound.

2.1 Problem Formulation

Suppose we have a user set $\mathcal{U}$ and an item set $\mathcal{I}$. Let u and i denote a user and an item from $\mathcal{U}$ and $\mathcal{I}$, respectively, and let $\mathcal{D} = \mathcal{U} \times \mathcal{I}$ denote the impression space, i.e., the set of all user–item pairs. For each $(u, i) \in \mathcal{D}$, let $x_{u,i}$ denotes the features of user-item pair (u, i), such as user ID and item ID. Furthermore, let $r_{u,i} \in \{0, 1\}$ denotes the conversion label of (u, i). The goal of the CVR prediction task is to learn a CVR prediction model $f(x_{u,i}; \theta)$ that provides an accurate prediction $\hat{r}_{u,i}$ for each (u, i) in $\mathcal{D}$. To achieve this goal, the ideal loss is:

$$\mathcal{L}_{ideal}(\theta) = \frac{1}{|\mathcal{D}|} \sum_{(u,i) \in \mathcal{D}} e_{u,i}, \tag{1}$$

where $e_{u,i} = l(\hat{r}_{u,i}, r_{u,i})$ denotes the prediction error of $\hat{r}_{u,i}$, and $l(\cdot, \cdot)$ is the binary cross-entropy loss function.

However, the ideal loss is not computable in practice because most of the conversion labels in the impression space are missing. In particular, the user behavior chain in the CVR task is: impression $\longrightarrow$ click $\longrightarrow$ conversion, indicating that the conversion label can only be observed after the user clicks on the item. To formulate the scenario of missing data, let $o_{u,i} \in \{0, 1\}$ denotes whether $r_{u,i}$ is observed. Accordingly, we obtain observational data[2] and denote it as $\mathcal{O} = \{(u, i) | (u, i) \in \mathcal{D}, o_{u,i} = 1\}$. A straightforward way to train a CVR prediction model is to minimize the naive loss:

$$\mathcal{L}_{naive}(\theta) = \frac{1}{|\mathcal{D}|} \sum_{(u,i) \in \mathcal{D}} o_{u,i} e_{u,i}. \tag{2}$$

[2]In this article, we use the terms "click event" and "observational data" interchangeably.

Nonetheless, the observational data suffer from selection bias [3, 8, 45, 52]. One main reason is that users click on items in the impression space based on their preferences rather than randomly. This means the naive estimator is a biased estimator of the ideal loss, i.e., $\mathbb{E}[\mathcal{L}_{naive}(\theta)] \neq \mathcal{L}_{ideal}(\theta)$. Blindly minimizing it leads to a biased CVR prediction model with suboptimal performance. Therefore, it is essential to build unbiased estimators of the ideal loss.

2.2 Existing Estimators

2.2.1 IPS Estimator. The propensity score represents the probability of a click (i.e., a user–item pair's conversion label being observed) and is denoted as $p_{u,i} = \mathbb{P}(o_{u,i} = 1 | x_{u,i})$. The IPS estimator reweights prediction errors on the observational data using IPSs:

$$\mathcal{L}_{IPS}(\theta) = \frac{1}{|\mathcal{D}|} \sum_{(u,i)\in\mathcal{D}} \frac{o_{u,i} e_{u,i}}{\hat{p}_{u,i}}, \tag{3}$$

where $\hat{p}_{u,i}$ is an estimation of $p_{u,i}$, and can be obtained from a propensity model $f(x_{u,i};\psi)$ trained by minimizing the propensity model training loss as follows:

$$\mathcal{L}_p(\psi) = \frac{1}{|\mathcal{D}|} \sum_{(u,i)\in\mathcal{D}} l(\hat{p}_{u,i}, o_{u,i}). \tag{4}$$

If the learned propensity scores are accurate, then the IPS estimator is unbiased, i.e., if $\hat{p}_{u,i} = p_{u,i}$ for all $(u,i) \in \mathcal{D}$, then $\mathbb{E}[\mathcal{L}_{IPS}(\theta)] = \mathcal{L}_{ideal}(\theta)$.

2.2.2 EIB Estimator. The EIB estimator imputes errors for missing data and estimates the ideal loss as follows:

$$\mathcal{L}_{EIB}(\theta) = \frac{1}{|\mathcal{D}|} \sum_{(u,i)\in\mathcal{D}} o_{u,i} e_{u,i} + (1 - o_{u,i})\hat{e}_{u,i}, \tag{5}$$

where $\hat{e}_{u,i}$ is the imputed error for user-item pair (u,i), which can be obtained from an imputation model $f(x_{u,i};\phi)$ trained by minimizing the following imputation model training loss:

$$\mathcal{L}_e(\phi) = \frac{1}{|\mathcal{D}|} \sum_{(u,i)\in\mathcal{D}} \frac{o_{u,i} l(\hat{e}_{u,i}, e_{u,i})}{\hat{p}_{u,i}}. \tag{6}$$

Although the goal of the imputation model is to accurately impute errors over all user–item pairs, we can only observe $e_{u,i}$ when $o_{u,i} = 1$. Therefore, the imputation model training loss on the observational data is reweighted using the propensity scores $\hat{p}_{u,i}$. If the imputed errors are accurate, then the EIB estimator is unbiased, i.e., if $\hat{e}_{u,i} = e_{u,i}$ for all $(u,i) \in \mathcal{D}$, then $\mathbb{E}[\mathcal{L}_{EIB}(\theta)] = \mathcal{L}_{ideal}(\theta)$.

2.2.3 DR Estimator. The DR estimator combines the IPS and the EIB estimators to estimate the ideal loss as follows:

$$\mathcal{L}_{DR}(\theta) = \frac{1}{|\mathcal{D}|} \sum_{(u,i)\in\mathcal{D}} \left(\hat{e}_{u,i} + \frac{o_{u,i}(e_{u,i} - \hat{e}_{u,i})}{\hat{p}_{u,i}} \right). \tag{7}$$

The DR estimator achieves double robustness, which means that it is unbiased if either the imputed errors or the learned propensity scores are accurate, i.e., if $\hat{e}_{u,i} = e_{u,i}$ or $\hat{p}_{u,i} = p_{u,i}$ for all $(u,i) \in \mathcal{D}$, then $\mathbb{E}[\mathcal{L}_{DR}(\theta)] = \mathcal{L}_{ideal}(\theta)$. This estimator formulation is widely used in existing DR methods [5, 10, 19, 22, 57].

2.3 Enhanced DR Methods for Bias and Variance

In this section, we first introduce some recent debiasing methods for controlling the bias and variance of the DR estimator, then show these methods still have high or unbounded bias, variance, and generalization error bound under small propensity scores.

2.3.1 MRDR. MRDR proposes a new imputation model training loss to obtain an imputation model that reduces the variance of the DR estimator, as shown below:

$$\mathcal{L}_{MRDR-e}(\phi) = \sum_{(u,i)\in\mathcal{D}} \frac{o_{u,i}(1-\hat{p}_{u,i})(\hat{e}_{u,i}-e_{u,i})^2}{\hat{p}_{u,i}^2}. \tag{8}$$

2.3.2 DR-MSE. DR-MSE estimates the bias of the DR estimator and proposes an imputation model training loss to control this bias, defined as follows:

$$\mathcal{L}_{DR-BIAS-e}(\phi) = \sum_{(u,i)\in\mathcal{D}} \frac{o_{u,i}(o_{u,i}-\hat{p}_{u,i})^2(\hat{e}_{u,i}-e_{u,i})^2}{\hat{p}_{u,i}^3}. \tag{9}$$

Then, DR-MSE combines the imputation model training losses of DR-BIAS and MRDR as follows:

$$\mathcal{L}_{\text{DR-MSE-e}}(\phi) = \alpha\mathcal{L}_{\text{DR-BIAS-e}}(\phi) + (1-\alpha)\mathcal{L}_{\text{MRDR-e}}(\phi), \tag{10}$$

where α is a hyperparameter that controls the bias-variance tradeoff.

2.3.3 GPL. GPL proposes to add the estimated bias and variance of the DR estimator into the propensity model training loss. The modified propensity model training loss is shown below:

$$\mathcal{L}_{GPL-p}(\psi) = \frac{1}{|\mathcal{D}|}\sum_{(u,i)\in\mathcal{D}} l(\hat{p}_{u,i}, o_{u,i}) + \lambda\left[\beta\widehat{\text{Bias}}(\mathcal{L}_{DR}) + (1-\beta)\widehat{\text{Var}}(\mathcal{L}_{DR})\right], \tag{11}$$

where λ and β are hyperparameters controlling the contribution of each component, and $\widehat{\text{Bias}}(\mathcal{L}_{DR})$ and $\widehat{\text{Var}}(\mathcal{L}_{DR})$ are:

$$\begin{aligned}
\widehat{\text{Bias}}(\mathcal{L}_{DR}) &= \frac{1}{|\mathcal{D}|}\sum_{(u,i)\in\mathcal{D}} \frac{o_{u,i}-2o_{u,i}\hat{p}_{u,i}+\hat{p}_{u,i}^2}{\hat{p}_{u,i}^2}(e_{u,i}-\hat{e}_{u,i})^2,\\
\widehat{\text{Var}}(\mathcal{L}_{DR}) &= \frac{1}{|\mathcal{D}|}\sum_{(u,i)\in\mathcal{D}} \frac{o_{u,i}}{\hat{p}_{u,i}^2}(e_{u,i}-\hat{e}_{u,i})^2.
\end{aligned} \tag{12}$$

However, these methods control the bias and variance by directly combining them into the propensity model training loss or the imputation model training loss, instead of proposing a new estimator or learning algorithm to achieve bounded bias and variance. Therefore, under small propensity scores arising from sparse click events, these methods still suffer from high or unbounded bias, variance, and generalization error bound. We will elaborate on these issues in detail in the next section.

3 Method

In this section, we elaborate on our BADR method. First, we clarify our theoretical motivation. Next, we propose a BADR estimator, followed by an analysis of its bias, variance, and generalization error bound. We then propose a new batch-adaptive imputation model training loss. Finally, we present the overall algorithm of BADR.

Table 1. Comparison of the Bias, Variance, and Generalization Error Bound of the IPS, DR, and Our Proposed BADR Estimators

	IPS	DR	BADR
Bias [69]	$\lvert\mathcal{D}\rvert^{-1}\lvert\sum_{u,i\in\mathcal{D}}(\hat{p}_{u,i}-p_{u,i})e_{u,i}/\hat{p}_{u,i}\rvert$	$\lvert\mathcal{D}\rvert^{-1}\lvert\sum_{u,i\in\mathcal{D}}(\hat{p}_{u,i}-p_{u,i})(e_{u,i}-\hat{e}_{u,i})/\hat{p}_{u,i}\rvert$	See Theorem 3.1
Robust to small $\hat{p}_{u,i}$	No, Bias $(\mathcal{L}_{\mathrm{IPS}})\to\infty$ when $\hat{p}_{u,i}\to 0$	No, Bias $(\mathcal{L}_{\mathrm{DR}})\to\infty$ when $\hat{p}_{u,i}\to 0$	Yes
Variance [69]	$\lvert\mathcal{D}\rvert^{-2}\sum_{u,i\in\mathcal{D}}p_{u,i}(1-p_{u,i})e_{u,i}^2/\hat{p}_{u,i}^2$	$\lvert\mathcal{D}\rvert^{-2}\sum_{u,i\in\mathcal{D}}p_{u,i}(1-p_{u,i})(e_{u,i}-\hat{e}_{u,i})^2/\hat{p}_{u,i}^2$	See Theorem 3.2
Robust to small $\hat{p}_{u,i}$	No, Var $(\mathcal{L}_{\mathrm{IPS}})\to\infty$ when $\hat{p}_{u,i}\to 0$	No, Var $(\mathcal{L}_{\mathrm{DR}})\to\infty$ when $\hat{p}_{u,i}\to 0$	Yes
Generalization error bound	$\lvert\mathcal{L}_{\mathrm{IPS}}-\mathbb{E}_O[\mathcal{L}_{\mathrm{IPS}}]\rvert\le\sqrt{\frac{\log\left(\frac{2}{\eta}\right)}{2\lvert\mathcal{D}\rvert^2}\sum_{u,i\in\mathcal{D}}\left(\frac{e_{u,i}}{\hat{p}_{u,i}}\right)^2}$	$\lvert\mathcal{L}_{\mathrm{DR}}-\mathbb{E}_O[\mathcal{L}_{\mathrm{DR}}]\rvert\le\sqrt{\frac{\log\left(\frac{2}{\eta}\right)}{2\lvert\mathcal{D}\rvert^2}\sum_{u,i\in\mathcal{D}}\left(\frac{e_{u,i}-\hat{e}_{u,i}}{\hat{p}_{u,i}}\right)^2}$	Combining Above
Robust to small $\hat{p}_{u,i}$	No, the error bound of IPS $\to\infty$ when $\hat{p}_{u,i}\to 0$	No, the error bound of DR $\to\infty$ when $\hat{p}_{u,i}\to 0$	Yes

3.1 Motivation

A major challenge in debiasing CVR prediction by the DR estimator is sparse observational data [35, 56], i.e., very few items are clicked by users in real-world scenarios. Given the definition of propensity scores in Section 2.2.1, the sparsity implies that for many user–item pairs, the propensity scores are small. Table 1 demonstrates that the bias, variance, and generalization error bound of the IPS and DR estimators are highly sensitive to data with small propensity scores. Particularly, the variance and generalization error bound of the DR estimator depend on the square of the learned propensity scores in the denominator and diverge to infinity when the learned propensity scores approach zero, regardless of the accuracy of the learned propensity scores. Moreover, since propensity scores cannot be learned with complete accuracy, bias also faces the problem of diverging to infinity when the learned propensity scores approach zero. Going further, the generalization error bound is a combination of bias and variance, which implies that it is high or even unbounded under small propensity scores. Consequently, the consistency between the descent of the ideal loss and the estimated loss cannot be guaranteed under sparse data, leading to suboptimal convergence and degraded model performance. The above analysis motivates us to propose a new BADR method with a more robust estimator.

3.2 BADR Estimator

We redesign the formulation of the DR estimator for batch training and propose a principled BADR estimator. This estimator adaptively adjusts the influence of each data batch on debiasing CVR prediction, based on the learned propensity scores within that batch. Formally, suppose we have K batches $\{\mathcal{D}^{(1)},\ldots,\mathcal{D}^{(K)}\}$ with batch size $\lvert\mathcal{D}^{(k)}\rvert=\lvert\mathcal{D}\rvert/K$ for $k=1,\ldots,K$. The BADR estimator is:

$$\begin{aligned}\mathcal{L}_{BADR-c}(\theta)&=\frac{1}{K}\sum_{k=1}^{K}\frac{1}{\sum_{(u,i)\in\mathcal{D}^{(k)}}\frac{o_{u,i}}{\hat{p}_{u,i}}}\sum_{(u,i)\in\mathcal{D}^{(k)}}\left(\hat{e}_{u,i}+\frac{o_{u,i}(e_{u,i}-\hat{e}_{u,i})}{\hat{p}_{u,i}}\right)\\&:=\frac{1}{K}\sum_{k=1}^{K}\mathcal{L}_{BADR-c}^{(k)}.\end{aligned}\tag{13}$$

The influence of each data batch, i.e., $1/\sum_{(u,i)\in\mathcal{D}^{(k)}}\{o_{u,i}/\hat{p}_{u,i}\}$, on training the CVR prediction model adapts across batches:

$$\begin{cases}\frac{1}{\sum_{(u,i)\in\mathcal{D}^{(k)}}\frac{o_{u,i}}{\hat{p}_{u,i}}}\le\frac{1}{\frac{1}{\hat{p}_{u',i'}}}<\frac{1}{\lvert\mathcal{D}^{(k)}\rvert}, & \text{if there exists } (u',i')\in\mathcal{D}_{o=1}^{(k)},\ \hat{p}_{u',i'}<\frac{1}{\lvert\mathcal{D}^{(k)}\rvert}\\ \frac{1}{\sum_{(u,i)\in\mathcal{D}^{(k)}}\frac{o_{u,i}}{\hat{p}_{u,i}}}>\frac{1}{\lvert\mathcal{D}_{o=1}^{(k)}\rvert\frac{\lvert\mathcal{D}^{(k)}\rvert}{\lvert\mathcal{D}_{o=1}^{(k)}\rvert}}=\frac{1}{\lvert\mathcal{D}^{(k)}\rvert}, & \text{if } \hat{p}_{u,i}>\frac{\lvert\mathcal{D}_{o=1}^{(k)}\rvert}{\lvert\mathcal{D}^{(k)}\rvert}\quad\text{for all } (u,i)\in\mathcal{D}_{o=1}^{(k)}\end{cases},\tag{14}$$

where $\mathcal{D}_{o=1}^{(k)}$ denotes $\{(u,i)\mid(u,i)\in\mathcal{D}^{(k)},o_{u,i}=1\}$.

Before further elaborating on the BADR estimator, we present the influence of each data batch in the DR estimator as follows:

$$\begin{aligned}\mathcal{L}_{DR}(\theta) &= \frac{1}{|\mathcal{D}|}\sum_{(u,i)\in\mathcal{D}}\left(\hat{e}_{u,i}+\frac{o_{u,i}(e_{u,i}-\hat{e}_{u,i})}{\hat{p}_{u,i}}\right)\\ &= \frac{1}{K}\sum_{k=1}^{K}\frac{1}{|\mathcal{D}^{(k)}|}\sum_{(u,i)\in\mathcal{D}^{(k)}}\left(\hat{e}_{u,i}+\frac{o_{u,i}(e_{u,i}-\hat{e}_{u,i})}{\hat{p}_{u,i}}\right),\end{aligned} \tag{15}$$

where the influence of each data batch on training the CVR prediction model, i.e., $1/|\mathcal{D}^{(k)}| = K/|\mathcal{D}|$, is constant across batches, neglecting the learned propensity scores within each batch.

From Equations (13), (14), and (15), we can observe that, compared to the DR estimator, the BADR estimator adaptively reduces the influence of data batches with small learned propensity scores while increasing the influence of those with large ones. It is worth noting that the key feature of the BADR estimator lies in the adaptive adjustment of each data batch's influence, rather than its absolute value necessarily being greater or smaller than $1/|\mathcal{D}^k|$. The BADR estimator mitigates the negative impact of data with extremely small learned propensity scores in debiasing CVR prediction by adjusting the influence of each data batch. This leads to a more robust training process and ultimately results in improved unbiased CVR prediction. The intuition behind this is that the reciprocals of these small propensity scores are used as data weights, which can cause certain data points to be assigned abnormally large weights. Such large weights may amplify the estimation error of the imputed errors $(e_{u,i} - \hat{e}_{u,i})$, thereby increasing the bias of the estimated loss for that data batch and having a substantial negative impact on model updates. At the same time, this leads to a highly imbalanced distribution of IPS weights, and such an imbalance increases the variance of the model updates from each data batch. Therefore, adaptively adjusting the influence of each batch on model training based on its propensity score helps to reduce the aforementioned bias and variance.

3.3 Theoretical Analysis

We formally derive the bias and variance of the BADR estimator to show that it can effectively overcome high or even unbounded bias, variance, and generalization error bound under sparse data. The conclusion is summarized in Table 1. For the DR estimator, the observation indicator $o_{u,i}$ only appears in the numerator. Thus, they can derive the bias and variance by simply taking the expectation operation. Note that we cannot simply take the expectation of the numerator and denominator of the BADR estimator. Thus, we use the Taylor expansion to derive the bias and variance, which are shown in the following two theorems.

Theorem 3.1 (Bias of the BADR Estimator). *Given the imputed errors $\hat{e}_{u,i}$ and the learned propensity scores $\hat{p}_{u,i} > 0$ for all user–item pairs, the bias of $\mathcal{L}_{BADR-c}$ is:*

$$\left|\frac{1}{|\mathcal{D}|}\sum_{k=1}^{K}\sum_{(u,i)\in\mathcal{D}^{(k)}}\left(e_{u,i}-\hat{e}_{u,i}-\frac{\sum_{(u,i)\in\mathcal{D}^{(k)}}(e_{u,i}-\hat{e}_{u,i})\frac{p_{u,i}}{\hat{p}_{u,i}}}{\sum_{(u,i)\in\mathcal{D}^{(k)}}\frac{p_{u,i}}{\hat{p}_{u,i}}}\right)\right| + O(|\mathcal{D}^{(k)}|^{-1}). \tag{16}$$

Proof. Let $\delta_{u,i} = e_{u,i} - \hat{e}_{u,i}$ and $\hat{\mathcal{E}}^{(k)} = \sum_{(u,i)\in\mathcal{D}^{(k)}} \hat{e}_{u,i}/\sum_{(u,i)\in\mathcal{D}^{(k)}}\{o_{u,i}/\hat{p}_{u,i}\}$, $\mathcal{L}_{BADR-c}^{(k)}$ can be expressed as:

$$\mathcal{L}_{BADR-c}^{(k)} = \frac{\frac{1}{|\mathcal{D}^{(k)}|}\sum_{(u,i)\in\mathcal{D}^{(k)}}\frac{o_{u,i}(\delta_{u,i}+\hat{\mathcal{E}}^{(k)})}{\hat{p}_{u,i}}}{\frac{1}{|\mathcal{D}^{(k)}|}\sum_{(u,i)\in\mathcal{D}^{(k)}}\frac{o_{u,i}}{\hat{p}_{u,i}}}. \tag{17}$$

For simplicity, let $w_{u,i} \triangleq o_{u,i}/\hat{p}_{u,i}, v_{u,i}^{(k)} \triangleq o_{u,i}(\delta_{u,i} + \hat{\mathcal{E}}^{(k)})/\hat{p}_{u,i}$, then $\mathcal{L}_{BADR-c}^{(k)}$ can be written as a ratio statistic:

$$\mathcal{L}_{BADR-c}^{(k)} = \frac{\frac{1}{|\mathcal{D}^{(k)}|}\sum_{(u,i)\in\mathcal{D}^{(k)}} v_{u,i}^{(k)}}{\frac{1}{|\mathcal{D}^{(k)}|}\sum_{(u,i)\in\mathcal{D}^{(k)}} w_{u,i}} \triangleq f(\bar{v}^{(k)}, \bar{w}^{(k)}), \tag{18}$$

where $\bar{v}^{(k)} = \sum_{(u,i)\in\mathcal{D}^{(k)}} v_{u,i}^{(k)}/|\mathcal{D}^{(k)}|$, $\bar{w}^{(k)} = \sum_{(u,i)\in\mathcal{D}^{(k)}} w_{u,i}/|\mathcal{D}^{(k)}|$, and $f(v,w) = v/w$. Applying the Taylor expansion around $(\mu_v^{(k)}, \mu_w^{(k)}) \triangleq (\mathbb{E}[\bar{v}^{(k)}], \mathbb{E}[\bar{w}^{(k)}])$, we have:

$$\begin{aligned} f(\bar{v}^{(k)}, \bar{w}^{(k)}) &= f(\mu_v^{(k)}, \mu_w^{(k)}) + f_v'(\mu_v^{(k)}, \mu_w^{(k)})(\bar{v}^{(k)} - \mu_v^{(k)}) + f_w'(\mu_v^{(k)}, \mu_w^{(k)})(\bar{w}^{(k)} - \mu_w^{(k)}) \\ &\quad + \frac{1}{2}\Big\{ f_{vv}''(\mu_v^{(k)}, \mu_w^{(k)})(\bar{v}^{(k)} - \mu_v^{(k)})^2 + 2 f_{vw}''(\mu_v^{(k)}, \mu_w^{(k)})(\bar{v}^{(k)} - \mu_v^{(k)})(\bar{w}^{(k)} - \mu_w^{(k)}) \\ &\quad + f_{ww}''(\bar{w}^{(k)} - \mu_w^{(k)})^2 + R(\tilde{v}^{(k)}, \tilde{w}^{(k)})\Big\}, \end{aligned} \tag{19}$$

where $R(\tilde{v}^{(k)}, \tilde{w}^{(k)})$ is the remainder term.

Recall that the definition $\mu_v^{(k)} \triangleq \mathbb{E}[\bar{v}^{(k)}]$, thus $\mathbb{E}(\bar{v}^{(k)} - \mu_v^{(k)}) = \mathbb{E}[\bar{v}^{(k)}] - \mathbb{E}[\bar{v}^{(k)}] = 0$. The same thing also holds for $\mathbb{E}(\bar{w}^{(k)} - \mu_w^{(k)}) = 0$. Thus, both first-order terms are equal to zero. In addition, note that $f_{vv}''(\mu_v^{(k)}, \mu_w^{(k)}) = 0$, $f_{vw}''(\mu_v^{(k)}, \mu_w^{(k)}) = -1/(\mu_w^{(k)})^2$, and $f_{ww}''(\mu_v^{(k)}, \mu_w^{(k)}) = 2\mu_v^{(k)}/(\mu_w^{(k)})^3$, then taking an expectation on both sides of the Taylor expansion leads to:

$$\mathbb{E}(\bar{v}^{(k)}/\bar{w}^{(k)}) = \frac{\mu_v^{(k)}}{\mu_w^{(k)}} - \frac{\mathrm{Cov}(\bar{v}^{(k)}, \bar{w}^{(k)})}{(\mu_w^{(k)})^2} + \frac{\mathrm{Var}(\bar{w}^{(k)})\mu_v^{(k)}}{(\mu_w^{(k)})^3} + \mathbb{E}[R(\tilde{v}^{(k)}, \tilde{w}^{(k)})]. \tag{20}$$

By some calculations, we have $\mathrm{Cov}(\bar{v}^{(k)}, \bar{w}^{(k)}) = O(|\mathcal{D}^{(k)}|^{-1})$, $\mathrm{Var}(\bar{w}^{(k)}) = O(|\mathcal{D}^{(k)}|^{-1})$, and $\mathbb{E}[R(\tilde{v}^{(k)}, \tilde{w}^{(k)})] = o(|\mathcal{D}^{(k)}|^{-1})$. We illustrate the first equation in detail, and the second equation follows the same thing. Specifically:

$$\begin{aligned} \mathrm{Cov}(\bar{v}^{(k)}, \bar{w}^{(k)}) &= \mathrm{Cov}\left(\sum_{(u,i)\in\mathcal{D}^{(k)}} \frac{v_{u,i}^{(k)}}{|\mathcal{D}^{(k)}|}, \sum_{(u,i)\in\mathcal{D}^{(k)}} \frac{w_{u,i}}{|\mathcal{D}^{(k)}|}\right) \\ &= |\mathcal{D}^{(k)}|^{-2} \sum_{(u,i)\in\mathcal{D}^{(k)}} \sum_{(u',i')\in\mathcal{D}^{(k)}} \mathrm{Cov}(v_{u,i}^{(k)}, w_{u',i'}). \end{aligned} \tag{21}$$

Note that

$$\mathrm{Cov}(v_{u,i}^{(k)}, w_{u',i'}) = \mathrm{Cov}\left(\frac{o_{u,i}}{\hat{p}_{u,i}}, \frac{o_{u',i'}(\delta_{u',i'} + \hat{\mathcal{E}}^{(k)})}{\hat{p}_{u',i'}}\right) = \frac{\delta_{u',i'} + \hat{\mathcal{E}}^{(k)}}{\hat{p}_{u,i}\hat{p}_{u',i'}} \mathrm{Cov}(o_{u,i}, o_{u',i'}), \tag{22}$$

since $\delta_{u',i'} + \hat{\mathcal{E}}^{(k)}$, $\hat{p}_{u,i}$, and $\hat{p}_{u',i'}$ can be treated as constants. Note that $\mathrm{Cov}(o_{u,i}, o_{u',i'}) = 0$ if $(u,i) \neq (u',i')$ and $\mathrm{Cov}(o_{u,i}, o_{u',i'}) = 1$ if $(u,i) = (u',i')$, thus:

$$\sum_{(u,i)\in\mathcal{D}^{(k)}} \sum_{(u',i')\in\mathcal{D}^{(k)}} \mathrm{Cov}(v_{u,i}^{(k)}, w_{u',i'}) = |\mathcal{D}^{(k)}|. \tag{23}$$

Thus $\mathrm{Cov}(\bar{v}^{(k)}, \bar{w}^{(k)}) = O(|\mathcal{D}^{(k)}|^{-1})$. In addition, $\mathbb{E}[R(\tilde{v}^{(k)}, \tilde{w}^{(k)})] = o(|\mathcal{D}^{(k)}|^{-1})$ is due to the $R(\tilde{v}^{(k)}\tilde{w}^{(k)})$ is the remainder term. Therefore, the bias of $\mathcal{L}_{BADR-c}^{(k)}$ is:

$$\left|\frac{1}{|\mathcal{D}^{(k)}|} \sum_{(u,i)\in\mathcal{D}^{(k)}} \left(\delta_{u,i} - \frac{\sum_{(u,i)\in\mathcal{D}^{(k)}} \delta_{u,i} p_{u,i}/\hat{p}_{u,i}}{\sum_{(u,i)\in\mathcal{D}^{(k)}} p_{u,i}/\hat{p}_{u,i}}\right)\right| + O(|\mathcal{D}^{(k)}|^{-1}). \tag{24}$$

Then, by the linearity of expectation, Theorem 3.1 is proved. □

Next, we derive the variance of the BADR estimator in the following theorem.

Theorem 3.2 (Variance of the BADR Estimator). *Given the imputed errors $\hat{e}_{u,i}$ and the learned propensity scores $\hat{p}_{u,i} > 0$ for all user–item pairs, the variance of $\mathcal{L}_{BADR-c}$ is:*

$$\frac{1}{K^2}\sum_{k=1}^{K}\frac{\sum_{(u,i)\in\mathcal{D}^{(k)}} p_{u,i}(1-p_{u,i})(h_{u,i}^{(k)})^2/\hat{p}_{u,i}^2}{\left(\sum_{(u,i)\in\mathcal{D}^{(k)}}\frac{p_{u,i}}{\hat{p}_{u,i}}\right)^2} + O(|\mathcal{D}^{(k)}|^{-2}), \tag{25}$$

where $h_{u,i}^{(k)}$ is a finite difference between the error deviation $e_{u,i} - \hat{e}_{u,i}$ and its weighted average, which is defined as:

$$h_{u,i}^{(k)} = e_{u,i} - \hat{e}_{u,i} - \frac{\sum_{(u,i)\in\mathcal{D}^{(k)}}\frac{p_{u,i}}{\hat{p}_{u,i}}(e_{u,i}-\hat{e}_{u,i})}{\sum_{(u,i)\in\mathcal{D}^{(k)}}\frac{p_{u,i}}{\hat{p}_{u,i}}}. \tag{26}$$

Proof. By the proof of Theorem 3.1, we have:

$$\mathbb{E}[\bar{v}^{(k)}/\bar{w}^{(k)}] - \mu_v^{(k)}/\mu_w^{(k)} = O(|\mathcal{D}^{(k)}|^{-1}). \tag{27}$$

Then the variance of $\mathcal{L}_{BADR-c}^{(k)}$ can be decomposed into as:

$$\begin{aligned}
\mathrm{Var}(\mathcal{L}_{BADR-c}^{(k)}) &= \mathrm{Var}(\bar{v}^{(k)}/\bar{w}^{(k)})\\
&= \mathbb{E}\left[\left\{\bar{v}^{(k)}/\bar{w}^{(k)} - \mathbb{E}(\bar{v}^{(k)}/\bar{w}^{(k)})\right\}^2\right]\\
&= \mathbb{E}\Bigg[\left\{\bar{v}^{(k)}/\bar{w}^{(k)} - \mu_v^{(k)}/\mu_w^{(k)}\right\}^2 - 2O(|\mathcal{D}^{(k)}|^{-1})\\
&\qquad \cdot\left\{\bar{v}^{(k)}/\bar{w}^{(k)} - \mu_v^{(k)}/\mu_w^{(k)}\right\} + O(|\mathcal{D}^{(k)}|^{-2})\Bigg]\\
&= \mathcal{V}_1^{(k)} + \mathcal{V}_2^{(k)} + O(|\mathcal{D}^{(k)}|^{-2}),
\end{aligned} \tag{28}$$

where $\mathcal{V}_1^{(k)} \triangleq \mathbb{E}[\{\bar{v}^{(k)}/\bar{w}^{(k)} - \mu_v^{(k)}/\mu_w^{(k)}\}^2]$, $\mathcal{V}_2^{(k)} \triangleq -2O(|\mathcal{D}^{(k)}|^{-1})\cdot[\mathbb{E}(\bar{v}^{(k)}/\bar{w}^{(k)}) - \mu_v^{(k)}/\mu_w^{(k)}]$. Equation (27) implies that $\mathcal{V}_2^{(k)} = O(|\mathcal{D}^{(k)}|^{-2})$. Let $f(v,w) = v/w$, and apply delta method around $(\mu_v^{(k)}, \mu_w^{(k)})$ to calculate $\mathcal{V}_1^{(k)}$:

$$\begin{aligned}
\mathcal{V}_1^{(k)} &= \mathbb{E}\left\{\left[f_v'(\mu_v^{(k)},\mu_w^{(k)})(\bar{v}^{(k)}-\mu_v^{(k)}) + f_w'(\mu_v^{(k)},\mu_w^{(k)})(\bar{w}^{(k)}-\mu_w^{(k)}) + O_p(|\mathcal{D}^{(k)}|^{-1})\right]^2\right\}\\
&= f_v'^2(\mu_v^{(k)},\mu_w^{(k)})\,\mathrm{Var}(\bar{v}^{(k)}) + 2f_v'(\mu_v^{(k)},\mu_w^{(k)})f_w'(\mu_v^{(k)},\mu_w^{(k)})\,\mathrm{Cov}(\bar{v}^{(k)},\bar{w}^{(k)})\\
&\quad + f_w'^2(\mu_v^{(k)},\mu_w^{(k)})\,\mathrm{Var}(\bar{w}^{(k)}) + O(|\mathcal{D}^{(k)}|^{-2}).
\end{aligned} \tag{29}$$

Note that $f'_v(\mu_v^{(k)}, \mu_w^{(k)}) = 1/\mu_w^{(k)}$ and $f'_w(\mu_v^{(k)}, \mu_w^{(k)}) = -\mu_v^{(k)}/(\mu_w^{(k)})^2$, we have:

$$
\begin{aligned}
\mathcal{V}_1^{(k)} &= \frac{1}{\left(\mu_w^{(k)}\right)^2} \operatorname{Var}(\bar{v}^{(k)}) + 2\frac{-\mu_v^{(k)}}{(\mu_w^{(k)})^3} \operatorname{Cov}(\bar{v}^{(k)}, \bar{w}^{(k)}) + \frac{(\mu_v^{(k)})^2}{(\mu_w^{(k)})^4} \operatorname{Var}(\bar{w}^{(k)}) + O(|\mathcal{D}^{(k)}|^{-2}) \\
&= \frac{(\mu_v^{(k)})^2}{(\mu_w^{(k)})^2} \left[\frac{\operatorname{Var}(\bar{v}^{(k)})}{(\mu_v^{(k)})^2} - 2\frac{\operatorname{Cov}(\bar{v}^{(k)}, \bar{w}^{(k)})}{\mu_v^{(k)}\mu_w^{(k)}} + \frac{\operatorname{Var}(\bar{w}^{(k)})}{(\mu_w^{(k)})^2} \right] + O(|\mathcal{D}^{(k)}|^{-2}) \\
&= \mathbb{E}\left(\bar{v}^{(k)} - \frac{\mu_v^{(k)}}{\mu_w^{(k)}} \bar{w}^{(k)} \right)^2 \Big/ (\mu_w^{(k)})^2 + O(|\mathcal{D}^{(k)}|^{-2}) \\
&= \frac{\sum_{(u,i)\in\mathcal{D}^{(k)}} p_{u,i}(1-p_{u,i})(h_{u,i}^{(k)})^2/\hat{p}_{u,i}^2}{\left(\sum_{(u,i)\in\mathcal{D}^{(k)}} p_{u,i}/\hat{p}_{u,i}\right)^2} + O(|\mathcal{D}^{(k)}|^{-2}),
\end{aligned}
\tag{30}
$$

where $h_{u,i}^{(k)} = \delta_{u,i} - \sum_{(u,i)\in\mathcal{D}^{(k)}} \{p_{u,i}\delta_{u,i}/\hat{p}_{u,i}\} / \sum_{(u,i)\in\mathcal{D}^{(k)}} \{p_{u,i}/\hat{p}_{u,i}\}$ is a bounded difference between $\delta_{u,i}$ and its weighted average. By the independence of batches, Theorem 3.2 is proved. □

Remark. Theorem 3.1 reveals that $\mathcal{L}_{BADR-c}$ is an *asymptotic DR unbiased estimator* of the ideal loss, which is asymptotically unbiased if either $\hat{e}_{u,i}$ or $\hat{p}_{u,i}$ is accurate for each user–item pair. Asymptotic unbiased means that when $|\mathcal{D}^{(k)}| \to \infty$, the bias of the BADR estimator converges to zero. In practical applications, asymptotic double robustness and double robustness are similar in terms of debiasing. For each batch, when the learned propensity scores or the imputed errors are accurate (i.e., the conditions for unbiased loss estimation in double robustness and asymptotic double robustness), the expectation of the BADR (i.e., $\mathbb{E}[1/\sum_{(u,i)\in\mathcal{D}^{(k)}}\{o_{u,i}/\hat{p}_{u,i}\}] \sum_{(u,i)\in\mathcal{D}^{(k)}} e_{u,i}$) differs from the ideal loss for the batch (i.e., $1/|\mathcal{D}^{(k)}| \sum_{(u,i)\in\mathcal{D}^{(k)}} e_{u,i}$) by a constant multiplier when updating the prediction model, implying that the update direction is unbiased. A potential consideration is that these constants may vary across batches. However, given the widespread use of adaptive optimizers like Adam in practical model training, the DR estimator also exhibits a constant difference from the ideal loss for each batch, where the constant varies across batches.

Corollary 3.1 (Bias and Variance Comparison). *Suppose $|e_{u,i} - \hat{e}_{u,i}| \le M, \ \forall (u,i)$, the BADR estimator has bounded bias and variance and exhibits smaller bias and variance than the DR estimator in the presence of small learned propensity scores. Specifically, $\exists C > 0$, such that if $\hat{p}_{u,i} < C$ for some $(u,i) \in \mathcal{D}$, then* $\operatorname{Bias}(\mathcal{L}_{BADR-c}) < \operatorname{Bias}(\mathcal{L}_{DR})$ *and* $\operatorname{Var}(\mathcal{L}_{BADR-c}) < \operatorname{Var}(\mathcal{L}_{DR})$.

Proof. If $|e_{u,i} - \hat{e}_{u,i}| \le M$, then $\sum_{(u,i)\in\mathcal{D}^{(k)}}\{p_{u,i}(e_{u,i} - \hat{e}_{u,i})/\hat{p}_{u,i}\} / \sum_{(u,i)\in\mathcal{D}^{(k)}}\{p_{u,i}/\hat{p}_{u,i}\} \le M, \ \forall k$. The bias of the BADR estimator is shown in Theorem 3.1, which is bounded by:

$$
\left| \frac{1}{|\mathcal{D}|} \sum_{k=1}^{K} \sum_{(u,i)\in\mathcal{D}^{(k)}} \Big(M + M\Big) \right| + O(|\mathcal{D}^{(k)}|^{-1}) = 2M + O(|\mathcal{D}^{(k)}|^{-1}). \tag{31}
$$

The variance of the BADR estimator is shown in Theorem 3.2. Because $h_{u,i}^{(k)} \le 2M$, then both the numerator and denominator of Theorem 3.2 are quadratic functions of $1/\hat{p}_{u,i}$, leading to a bounded variance. From Table 1, the bias and variance of the DR estimator are infinite when $\hat{p}_{u,i} \to 0$. Thus, there exists a constant C to support the conclusion. □

Corollary 3.2 (Generalization Error Bound Comparison). *The generalization error bound of the BADR is the combination of bias and variance [37], which is also bounded with arbitrarily small learned propensity scores. Similarly, the BADR also exhibits a smaller bound with $\hat{p}_{u,i} < C$ for some $C > 0$ and $(u, i) \in \mathcal{D}$ compared to the DR estimator.*

Remark. Corollary 3.3 and Corollary 3.4 reveal that: (1) the BADR estimator has bounded bias, variance, and generalization error bound for arbitrarily small values of $\hat{p}_{u,i}$; (2) the BADR estimator achieves smaller bias, variance, and generalization error bound compared to the DR estimator in the presence of small propensity scores; (3) the BADR estimator facilitates superior optimization of the ideal loss during model training on the estimated loss, thereby improving debiasing performance. The excellent theoretical properties of the BADR estimator demonstrate that it is a more robust estimator compared to the DR estimator used by the existing DR methods, which provides the foundation for the strong debiasing performance of the BADR method on sparse data.

3.4 Batch-Adaptive Imputation Model Training Loss

To further enhance the algorithmic robustness, we propose a novel variance-reduction imputation model training loss based on the approximation of the variance of the BADR estimator using the delta method [48] and further optimize it. Note that the variance of the BADR estimator in Theorem 3.2 cannot be optimized directly due to the requirement of the groundtruth propensity scores and prediction errors for all user–item pairs. The reason for proposing a new variance-reduction imputation model training loss, instead of using the existing loss, is that existing variance-reduction imputation model training losses (i.e., MRDR loss) cannot theoretically guarantee further variance reduction for the BADR estimator. This mismatch between the training loss and the estimator limits the potential for variance reduction, which is crucial for improving the performance and reliability of the estimator in practical applications.

Theorem 3.5 (Delta Method Variance Approximation). *Given the imputed errors $\hat{e}_{u,i}$ and the learned propensity scores $\hat{p}_{u,i} > 0$ for all user–item pairs, the variance approximation of $\mathcal{L}^{(k)}_{BADR-c}$ given by delta method is:*

$$\widehat{\mathrm{Var}}\left(\mathcal{L}^{(k)}_{BADR-c}\right) = \frac{\sum_{(u,i)\in\mathcal{D}^{(k)}} w_{u,i}^2 \left(e_{u,i} - \hat{e}_{u,i} + \hat{\mathcal{E}}^{(k)} - \mathcal{L}^{(k)}_{BADR-c}\right)^2}{\left(\sum_{(u,i)\in\mathcal{D}^{(k)}} w_{u,i}\right)^2}, \tag{32}$$

where $w_{u,i} = o_{u,i}/\hat{p}_{u,i}$ and $\hat{\mathcal{E}}^{(k)} = \sum_{(u,i)\in\mathcal{D}^{(k)}} \hat{e}_{u,i} / \sum_{(u,i)\in\mathcal{D}^{(k)}} \{o_{u,i}/\hat{p}_{u,i}\}$.

Proof. Recall that $(\mu_v^{(k)}, \mu_w^{(k)}) \triangleq \left(\mathbb{E}[\bar{v}^{(k)}], \mathbb{E}[\bar{w}^{(k)}]\right)$, the first-order approximation of $\mathbb{E}(f(v_{u,i}^{(k)}, w_{u,i}^{(k)})$ is $\mathbb{E}(f(v_{u,i}^{(k)}, w_{u,i}^{(k)}) \approx f(\mu_v^{(k)}, \mu_w^{(k)})$, where $\bar{v}^{(k)} = \sum_{(u,i)\in\mathcal{D}^{(k)}} v_{u,i}^{(k)}/|\mathcal{D}^{(k)}|$, $\bar{w}^{(k)} = \sum_{(u,i)\in\mathcal{D}^{(k)}} w_{u,i}/|\mathcal{D}^{(k)}|$, $v_{u,i}^{(k)} = o_{u,i}\left(\delta_{u,i} + \hat{\mathcal{E}}^{(k)}\right)/\hat{p}_{u,i}$, $w_{u,i} = o_{u,i}/\hat{p}_{u,i}$, and $f(v, w) = v/w$. Thus, by the definition of variance, we have:

$$\begin{aligned}\mathrm{Var}(f(v_{u,i}^{(k)}, w_{u,i}^{(k)})) &= \mathbb{E}\left[(f(v_{u,i}^{(k)}, w_{u,i}^{(k)}) - \mathbb{E}(f(v_{u,i}^{(k)}, w_{u,i}^{(k)})))^2\right] \\ &\approx \mathbb{E}\left[(f(v_{u,i}^{(k)}, w_{u,i}^{(k)}) - f(\mu_v^{(k)}, \mu_w^{(k)}))^2\right].\end{aligned} \tag{33}$$

Thus, applying Taylor expansion around $(\mu_v^{(k)}, \mu_w^{(k)})$ gives that:

$$\begin{aligned}
\operatorname{Var}(f(v_{u,i}^{(k)}, w_{u,i}^{(k)})) &\approx \mathbb{E}\Big[\big(f(\mu_v^{(k)}, \mu_w^{(k)}) + f_v'(\mu_v^{(k)}, \mu_w^{(k)})(v_{u,i}^{(k)} - \mu_v^{(k)}) \\
&\quad + f_w'(\mu_v^{(k)}, \mu_w^{(k)})(w_{u,i}^{(k)} - \mu_w^{(k)}) - f(\mu_v^{(k)}, \mu_w^{(k)})\big)^2\Big] \\
&= \mathbb{E}\Big[\big(f_v'(\mu_v^{(k)}, \mu_w^{(k)})(v_{u,i}^{(k)} - \mu_v^{(k)}) + f_w'(\mu_v^{(k)}, \mu_w^{(k)})(w_{u,i}^{(k)} - \mu_w^{(k)})\big)^2\Big] \\
&= f_v'^2(\mu_v^{(k)}, \mu_w^{(k)}) \operatorname{Var}(v_{u,i}^{(k)}) + f_w'^2(\mu_v^{(k)}, \mu_w^{(k)}) \operatorname{Var}(w_{u,i}^{(k)}) \\
&\quad + 2 f_v'(\mu_v^{(k)}, \mu_w^{(k)}) f_w'(\mu_v^{(k)}, \mu_w^{(k)}) \operatorname{Cov}(v_{u,i}^{(k)}, w_{u,i}^{(k)}).
\end{aligned} \tag{34}$$

Note that $f_v'(\mu_v^{(k)}, \mu_w^{(k)}) = 1/\mu_w^{(k)}$ and $f_w'(\mu_v^{(k)}, \mu_w^{(k)}) = -\mu_v^{(k)}/(\mu_w^{(k)})^2$, then the approximation of $\operatorname{Var}(f(v_{u,i}^{(k)}, w_{u,i}^{(k)}))$ is given as:

$$\begin{aligned}
\operatorname{Var}(v_{u,i}^{(k)}/w_{u,i}^{(k)}) &\equiv \operatorname{Var}(f(v_{u,i}^{(k)}, w_{u,i}^{(k)})) \\
&\approx \frac{1}{(\mu_w^{(k)})^2} \operatorname{Var}(v_{u,i}^{(k)}) + 2\frac{-\mu_v^{(k)}}{(\mu_w^{(k)})^3} \operatorname{Cov}(v_{u,i}^{(k)}, w_{u,i}^{(k)}) + \frac{(\mu_v^{(k)})^2}{(\mu_w^{(k)})^4} \operatorname{Var}(w_{u,i}^{(k)}) \\
&= \frac{(\mu_v^{(k)})^2}{(\mu_w^{(k)})^2} \left[\frac{\operatorname{Var}(v_{u,i}^{(k)})}{(\mu_v^{(k)})^2} - 2\frac{\operatorname{Cov}(v_{u,i}^{(k)}, w_{u,i}^{(k)})}{\mu_v^{(k)} \mu_w^{(k)}} + \frac{\operatorname{Var}(w_{u,i}^{(k)})}{(\mu_w^{(k)})^2} \right] \\
&= \frac{\mathbb{E}\left(v_{u,i}^{(k)} - \frac{\mu_v^{(k)}}{\mu_w^{(k)}} w_{u,i}^{(k)} \right)^2}{(\mu_w^{(k)})^2}.
\end{aligned} \tag{35}$$

After plugging in Equation (35) with the estimates of $\mu_w^{(k)}$ and $\mu_v^{(k)}$, and other unknowns, the variance approximation of the BADR estimator is:

$$\begin{aligned}
\widehat{\operatorname{Var}}\left(\mathcal{L}_{BADR-c}^{(k)}\right) &= \frac{1}{|\mathcal{D}^{(k)}|} \frac{\frac{1}{|\mathcal{D}^{(k)}|} \sum_{(u,i)\in\mathcal{D}^{(k)}} \left(v_{u,i}^{(k)} - w_{u,i} \mathcal{L}_{BADR^{(k)}-c} \right)^2}{\left(\frac{1}{|\mathcal{D}^{(k)}|} \sum_{(u,i)\in\mathcal{D}^{(k)}} w_{u,i} \right)^2} \\
&= \frac{\sum_{(u,i)\in\mathcal{D}^{(k)}} w_{u,i}^2 \left(e_{u,i} - \hat{e}_{u,i} + \hat{\mathcal{E}}^{(k)} - \mathcal{L}_{BADR-c}^{(k)} \right)^2}{\left(\sum_{(u,i)\in\mathcal{D}^{(k)}} w_{u,i} \right)^2},
\end{aligned} \tag{36}$$

where $w_{u,i} = o_{u,i}/\hat{p}_{u,i}$. □

Theorem 3.3 demonstrates that the approximated variance of the BADR estimator depends heavily on the imputed errors. Therefore, we derive a batch-adaptive imputation model training loss $\mathcal{L}_{BADR-e}^{k}(\phi) := \widehat{\operatorname{Var}}\left(\mathcal{L}_{BADR-c}^{(k)}\right)$ in each batch, which further enhances the reduction of variance in CVR model training and improves algorithmic robustness theoretically. Intuitively, the unique batch-adaptive property and additional constraint in the BADR imputation model training loss, enable it to force the imputation model better, which, in turn, benefits the training of the CVR prediction model using the BADR estimator. Particularly, the BADR imputation model training loss features a unified adaptive influence of each data batch with the BADR estimator, which adapts the influence of each data batch on the training of the imputation model with $\left(\sum_{(u,i)\in\mathcal{D}^{(k)}} w_{u,i}\right)^2$. Furthermore,

Algorithm 1: The Extended Joint Learning Algorithm for BADR

Input: Observed user-item pairs $\mathcal{O}$, all user-item pairs $\mathcal{D}$, and step size η;
Initialize the parameters ψ, ϕ, θ;
while stopping criteria is not satisfied **do**
  Sample a batch of user-item pairs from $\mathcal{D}$;
  Update $\psi \leftarrow \psi - \eta \nabla_{\psi} \mathcal{L}_p(\psi)$;
end while
while stopping criteria is not satisfied **do**
  Sample a batch with size $|\mathcal{D}^{(k)}|$ of user-item pairs from $\mathcal{D}$;
  Update $\phi \leftarrow \phi - \eta \nabla_{\phi} \mathcal{L}^{(k)}_{BADR-e}(\phi)$;
  Update $\theta \leftarrow \theta - \eta \nabla_{\theta} \mathcal{L}^{(k)}_{BADR-c}(\theta)$;
end while

the BADR imputation model training loss introduces an additional constraint $\hat{\mathcal{E}}^{(k)} - \mathcal{L}^{(k)}_{BADR-c}$ for $\hat{e}$, which leverages information from the entire data distribution and the BADR estimator.

3.5 Batch-Adaptive Joint Learning

In practice, given the learned propensity scores for all user–item pairs, we extend the widely used joint learning framework [57] to optimize the CVR prediction model and the imputation model jointly. Specifically, in each batch $\mathcal{D}^{(k)}$, we train imputation model $f(x_{u,i};\phi)$ by minimizing $\mathcal{L}^{(k)}_{BADR-e}(\phi)$. And we use $\mathcal{L}^{(k)}_{BADR-c}(\theta)$ to train CVR prediction model $f(x_{u,i};\theta)$. We alternately train the CVR prediction model and the imputation model within each batch until the early stopping criterion is met. The pseudo-code of the extended joint learning is shown in Algorithm 1. Since the proposed algorithm focuses on modifying the loss function, it does not increase the model inference time.

To the best of our knowledge, our method is the first to address recommendation debiasing from the batch training perspective. While batch training is a standard algorithm, existing methods are primarily designed to handle large-scale data efficiently, focusing on approximating the gradient of the overall loss function using batch-level samples. In contrast, BADR leverages batch-level samples to mitigate high bias and high variance under small propensity scores in recommendation debiasing. Specifically, by adaptively adjusting the influence of each data batch on model updates based on the propensity scores within the batch, BADR effectively reduces both bias and variance in cases where propensity scores are small, while maintaining asymptotic double robustness.

It is worth noting that, like other DR learning methods, BADR aims to estimate the loss over the entire data space, making it broadly applicable to scenarios with selection bias or missing-not-at-random data, such as rating prediction [43] and off-policy evaluation [17]. For instance, in rating prediction, only a small fraction of items receive ratings from users relative to the entire user–item space, which leads to small propensity scores and provides an opportunity for BADR to achieve better performance compared to existing DR methods.

3.6 Comparison between BADR and Self-Normalized IPS (SNIPS)

The SNIPS estimator [53] weights the IPS by the sum of IPSs, as shown below:

$$\mathcal{L}_{\text{SNIPS}}(\theta) = \frac{\sum_{(u,i)\in\mathcal{D}} \frac{o_{u,i} e_{u,i}}{\hat{p}_{u,i}}}{\sum_{(u,i)\in\mathcal{D}} \frac{o_{u,i}}{\hat{p}_{u,i}}}. \tag{37}$$

SNIPS can be viewed as a special case of BADR with $\hat{e}_{u,i} = 0$ for all $(u,i) \in \mathcal{D}$. This means SNIPS does not have an imputation model, thus does not have an imputation loss, joint learning algorithm, asymptotic double robustness/double robustness. As a result, the self-normalized technique cannot be used on the DR estimator directly. In summary, the contributions of the proposed BADR compared to SNIPS lie in the asymptotic double robustness property of BADR, the proposed variance reduction imputation loss derived based on the delta method, as well as the variance reduction learning algorithm.

3.7 Discussion on the Implementation of BADR under a Multi-Task Framework

BADR provides an improvement on the imputation loss function. To integrate BADR into a multi-task framework, the propensity prediction (i.e., **Click-Through Rate (CTR)** prediction), CVR prediction, and imputation prediction can be treated as three separate tasks. The overall training loss consists of the losses proposed by BADR for the propensity model, prediction model, and imputation model. A general design is as follows: a shared embedding table generates sample embeddings, which are then fed into a multi-task framework (e.g., MMOE [34]) to produce task-specific embeddings for each task. The task-specific embeddings are finally passed through each task's prediction network (e.g., an MLP) to predict propensity scores, CVR, and imputation errors. If there are additional tasks, such as **Click-Through Conversion Rate (CTCVR)** prediction [35], they can be incorporated into the framework in the same manner. The loss used for model training is computed as a weighted sum of BADR's propensity model training loss, prediction model training loss, imputation model training loss, and the losses of any additional tasks, enabling collaborative learning across all tasks. Since this is not the main focus of this work, we do not elaborate further.

4 Experiments

In this section, we conduct experiments to answer the following **Research Questions (RQ)**:

- *RQ1*: How does BADR perform compared to state-of-the-art methods *w.r.t.* debiasing performance?
- *RQ2*: How do different modules in BADR affect its debiasing performance?
- *RQ3*: How does BADR perform under varying levels of data sparsity?
- *RQ4*: Does BADR, as expected, enable the ideal loss to be better optimized compared to state-of-the-art debiasing methods?
- *RQ5*: How does batch size influence BADR's performance?
- *RQ6*: What are the convergence behaviors of the propensity model, prediction model, and imputation model in BADR?
- *RQ7*: How does BADR perform on high-propensity-score datasets?
- *RQ8*: What is the online performance of BADR?

4.1 Experimental Settings

4.1.1 Datasets. We conduct experiments on three datasets: Yahoo! R3 [36], Coat[3] [47], and KuaiRec[4] [7]. All three datasets contain biased data suffering from selection bias and unbiased data collected through random exposure policies. Table 2 summarizes the statistics of these datasets. The details are as follows.

- Yahoo! R3 is a classic dataset in the field of music recommendation, released by Yahoo. It contains explicit ratings (1–5) from 15,400 users on 1,000 songs. The dataset consists of a biased subset with 311,704 records and an unbiased subset with 54,000 records. Following the processing procedure in existing works [19, 22, 29], we binarize the ratings for Yahoo! R3,

[3] https://www.cs.cornell.edu/~schnabts/mnar/.
[4] https://kuairec.com/.

Table 2. The Statistics of Yahoo! R3, Coat, and KuaiRec

Dataset	#User	#Item	#Biased Data	#Unbiased Data	Sparsity Rate	Average CTR	Average CVR
Yahoo! R3	15,400	1,000	311,704	54,000	97.98%	0.02	0.56
Coat	290	300	6,960	4,640	92.00%	0.08	0.50
KuaiRec	1,411	3,327	201,171	117,113	95.71%	0.04	0.26

setting ratings below 3 as negative conversion labels (0) and ratings of 3 or higher as positive conversion labels (1). Accordingly, the rating behavior can be interpreted as a click.

— Coat is a dataset released via Amazon Mechanical Turk, simulating an online shopping scenario for coats. It contains explicit ratings (1–5) from 290 users on 300 coats. The dataset consists of a biased subset with 6,960 records and an unbiased subset with 4,640 records. Following the processing procedure in existing works [19, 22, 29], we binarize the ratings for Coat, setting ratings below 3 as negative conversion labels (0) and ratings of 3 or higher as positive conversion labels (1). Accordingly, the rating behavior can be interpreted as a click.
— KuaiRec is a dataset released by the Kuaishou short video platform. Users scroll through a recommended feed and watch short videos one by one, with each video displayed individually; users can switch at any time without waiting for the video to finish. We use the sampled version from [19, 29]. The dataset contains watch ratios of 1,411 users for 3,327 short videos, where the watch ratio is defined as the ratio of play duration to video duration. It consists of a biased subset with 201,171 records and an unbiased subset with 117,113 records. Following the official KuaiRec documentation, we binarize the watch ratio as a conversion label: a watch ratio greater than 2 is treated as a positive conversion label (1), while a watch ratio less than or equal to 2 is treated as a negative conversion label (0). Accordingly, the watch behavior can be interpreted as a click.

4.1.2 Baselines. We compare BADR with **Matrix Factorization (MF)** [18], the base model without debiasing, and the following debiasing methods in three categories: (1) IPS-based methods: *IPS* [47] and *SNIPS* [53]; (2) DR-based methods: *DRJL* [57], *MRDR* [10], *DR-BIAS* [5], *DR-MSE* [5], *StableDR* [29], *TDR-CL* [22], *DR-MSE-GPL* [69], *D-DR* [12], and *DCE-TDR* [19]; (3) propensity-free methods: *CVIB* [60], *DIB* [33], and *DAMF* [45]. Furthermore, we also compare debiasing methods that utilize the multi-task training: *Multi-IPS* [67], *ESCM2-IPS* [56], *Multi-DR* [67], *ESCM2-DR* [56], and *DDPO* [52]. The brief introductions to these debiasing methods are as follows:

— *IPS* [47] reweights prediction errors on observational data using IPSs.
— *SNIPS* [53] normalizes IPSs per batch for IPS.
— *DRJL* [57] incorporates IPS with EIB and jointly learns the propensity and imputation models.
— *MRDR* [10] adjusts the imputation model training loss in DR methods to reduce the variance of the DR estimator.
— *DR-BIAS* [5] adjusts the imputation model training loss in DR methods to reduce the bias of the DR estimator.
— *DR-MSE* [5] combines DR-BIAS and MRDR to reduce the bias and variance of the DR estimator.
— *StableDR* [29] enhances the robustness toward small propensity scores and reduces the prediction model's reliance on imputed errors to mitigate their negative impact.
— *TDR-CL* [22] enhances the robustness of DR methods caused by misspecified imputed errors via target learning.
— *DR-MSE-GPL* [69] adjusts the propensity model training loss to further balance the bias and variance of the DR estimator in DR-MSE.

— *D-DR* [12] adjusts the learned propensity scores to balance the bias and variance of the DR estimator.
— *DCE-TDR* [19] proposes a calibrator for the propensity and imputation models.
— *CVIB* [60] balances observational and missing data by extending the original information bottleneck [55].
— *DIB* [33] employs the information bottleneck [55] to decouple user and item representations into biased and unbiased components.
— *DAMF* [45] derives an upper bound for the prediction error on unbiased data and minimizes it through adversarial learning.
— *Multi-IPS* [67] directly updates the propensity model and prediction model in the IPS estimator.
— *ESCM2-IPS* [56] incorporates IPS into a multi-task framework.
— *Multi-DR* [67] directly updates the propensity model, imputation model, and prediction model in the DR estimator.
— *ESCM2-DR* [56] incorporates DR into a multi-task framework.
— *DDPO* [52] introduces a dual propensity network to optimize the CVR model using both clicked and unclicked data, while reinforcing the causal relationship between the networks through attention mechanisms.

4.1.3 Implementation Details. We implement our method in PyTorch [40].[5] Following [19], we implement MF as the backbone model, configuring embedding sizes to 64 for Yahoo! R3, 16 for Coat, and 64 for KuaiRec with corresponding batch sizes of 4,096, 128, and 4,096, respectively. The rest hyperparameters are optimized via Optuna [1], consistent with the original study's tuning framework. In particular, we tune the learning rate within the range of $[5e^{-3}, 1e^{-1}]$ and the weight decay within the range of $[1e^{-7}, 1]$. We use a small portion of the unbiased data for validation (10% for Yahoo! R3, 20% for Coat, and 10% for KuaiRec), with the rest reserved for testing. The difference in configuration for Coat compared to the other datasets is due to its smaller data size, with only 6,960 biased samples and 4,640 unbiased samples.

We run all baselines using their official code[6] and tune them based on the original hyperparameter ranges provided in their papers. For methods without official code, we use the implementation from [22]. For a fair comparison, we also tune all baselines using our hyperparameter ranges and report better results between those obtained with our range and the original range. Furthermore, following [19, 22], the early stopping criterion in our experiments for all methods is: if the loss on biased data increases over five epochs, training is stopped.

4.2 Performance Comparison (RQ1)

We compare all methods on Yahoo! R3, Coat, and KuaiRec. For each method, we run it five times using five different random seeds. Table 3 reports the mean and standard deviation of the results for each method. We observe that:

— The baseline DR method, DRJL, outperforms IPS and SNIPS, demonstrating the effectiveness of the DR estimator in debiasing CVR prediction. This result highlights the importance of advancing DR methods to enhance robustness against small propensity scores, rather than IPS. The superior performance stems from the DR estimator's double robustness, as discussed in Section 2.2, and aligns with findings in prior studies [10, 19].

[5] The code of BADR is available at https://github.com/HungPaan/DR-ALR/.
[6] We retain only the debias-related components in baselines that utilize the multi-task training setting. The reason is to isolate and evaluate the effectiveness of all debiasing methods without potential performance gains attributed to other auxiliary tasks.

Table 3. Performance Comparison between BADR and Baselines on Yahoo! R3, Coat, and KuaiRec

	Yahoo! R3			**Coat**			**KuaiRec**		
Method	**AUC**	**NDCG@5**	**Recall@5**	**AUC**	**NDCG@5**	**Recall@5**	**AUC**	**NDCG@50**	**Recall@50**
MF	$0.7096_{\pm 0.0001}$	$0.7089_{\pm 0.0010}$	$0.4660_{\pm 0.0018}$	$0.6983_{\pm 0.0088}$	$0.6658_{\pm 0.0055}$	$0.5151_{\pm 0.0093}$	$0.7530_{\pm 0.0012}$	$0.5714_{\pm 0.0026}$	$0.6252_{\pm 0.0055}$
IPS	$0.7097_{\pm 0.0005}$	$0.7106_{\pm 0.0021}$	$0.4674_{\pm 0.0012}$	$0.7371_{\pm 0.0043}$	$0.6959_{\pm 0.0031}$	$0.5381_{\pm 0.0076}$	$0.7789_{\pm 0.0007}$	$0.5761_{\pm 0.0027}$	$0.6470_{\pm 0.0057}$
SNIPS	$0.7108_{\pm 0.0006}$	$0.7122_{\pm 0.0015}$	$0.4685_{\pm 0.0017}$	$0.7439_{\pm 0.0020}$	$0.6868_{\pm 0.0043}$	$0.5296_{\pm 0.0057}$	$0.7844_{\pm 0.0013}$	$0.5887_{\pm 0.0020}$	$0.6472_{\pm 0.0049}$
DRJL	$0.7171_{\pm 0.0003}$	$0.7232_{\pm 0.0005}$	$0.4777_{\pm 0.0009}$	$0.7423_{\pm 0.0089}$	$0.7010_{\pm 0.0105}$	$0.5368_{\pm 0.0114}$	$0.7879_{\pm 0.0013}$	$0.5918_{\pm 0.0034}$	$0.6574_{\pm 0.0039}$
MRDR	$0.7186_{\pm 0.0003}$	$0.7248_{\pm 0.0007}$	$0.4780_{\pm 0.0008}$	$0.7515_{\pm 0.0038}$	$0.7063_{\pm 0.0056}$	$0.5458_{\pm 0.0071}$	$0.8075_{\pm 0.0034}$	$0.5803_{\pm 0.0105}$	$0.6774_{\pm 0.0036}$
DR-BIAS	$0.7195_{\pm 0.0003}$	$0.7265_{\pm 0.0007}$	$0.4803_{\pm 0.0015}$	$0.7540_{\pm 0.0038}$	$0.7042_{\pm 0.0052}$	$0.5388_{\pm 0.0049}$	$0.8058_{\pm 0.0025}$	$0.5916_{\pm 0.0122}$	$0.6803_{\pm 0.0032}$
DR-MSE	$0.7194_{\pm 0.0005}$	$0.7270_{\pm 0.0010}$	$0.4800_{\pm 0.0013}$	$0.7561_{\pm 0.0027}$	$0.7064_{\pm 0.0035}$	$0.5432_{\pm 0.0040}$	$0.8049_{\pm 0.0032}$	$0.5926_{\pm 0.0091}$	$0.6671_{\pm 0.0021}$
StableDR	$0.7120_{\pm 0.0006}$	$0.7195_{\pm 0.0013}$	$0.4750_{\pm 0.0012}$	$0.7538_{\pm 0.0038}$	$0.7058_{\pm 0.0049}$	$0.5442_{\pm 0.0062}$	$0.8049_{\pm 0.0007}$	$0.6098_{\pm 0.0008}$	$0.6816_{\pm 0.0025}$
TDR-CL	$0.7212_{\pm 0.0008}$	$0.7283_{\pm 0.0021}$	$0.4830_{\pm 0.0031}$	$0.7571_{\pm 0.0015}$	$0.7070_{\pm 0.0037}$	$0.5435_{\pm 0.0088}$	$0.8061_{\pm 0.0043}$	$0.5968_{\pm 0.0141}$	$0.6519_{\pm 0.0013}$
DR-MSE-GPL	$0.7229_{\pm 0.0007}$	$0.7319_{\pm 0.0015}$	$0.4844_{\pm 0.0022}$	$0.7565_{\pm 0.0009}$	$0.7090_{\pm 0.0008}$	$0.5417_{\pm 0.0038}$	$0.8126_{\pm 0.0009}$	$0.6151_{\pm 0.0008}$	$0.6822_{\pm 0.0019}$
D-DR	$0.7198_{\pm 0.0005}$	$0.7274_{\pm 0.0012}$	$0.4785_{\pm 0.0010}$	$0.7556_{\pm 0.0021}$	$0.7056_{\pm 0.0024}$	$0.5422_{\pm 0.0034}$	$0.7833_{\pm 0.0019}$	$0.5872_{\pm 0.0032}$	$0.6572_{\pm 0.0003}$
DCE-TDR	$\underline{0.7233_{\pm 0.0003}}$	$\underline{0.7600_{\pm 0.0013}}$	$\underline{0.5170_{\pm 0.0012}}$	$0.7608_{\pm 0.0023}$	$\underline{0.7162_{\pm 0.0047}}$	$\underline{0.5552_{\pm 0.0046}}$	$\underline{0.8269_{\pm 0.0019}}$	$\underline{0.6566_{\pm 0.0034}}$	$\underline{0.6991_{\pm 0.0052}}$
CVIB	$0.7103_{\pm 0.0006}$	$0.7249_{\pm 0.0016}$	$0.4912_{\pm 0.0012}$	$0.7579_{\pm 0.0045}$	$0.6793_{\pm 0.0064}$	$0.5302_{\pm 0.0051}$	$0.7701_{\pm 0.0018}$	$0.5839_{\pm 0.0031}$	$0.6312_{\pm 0.0036}$
DIB	$0.7133_{\pm 0.0007}$	$0.7073_{\pm 0.0014}$	$0.4654_{\pm 0.0029}$	$0.7556_{\pm 0.0025}$	$0.7014_{\pm 0.0049}$	$0.5409_{\pm 0.0052}$	$0.7702_{\pm 0.0012}$	$0.5836_{\pm 0.0028}$	$0.6304_{\pm 0.0012}$
DAMF	$0.7216_{\pm 0.0006}$	$0.7402_{\pm 0.0009}$	$0.4963_{\pm 0.0016}$	$0.7561_{\pm 0.0050}$	$0.7012_{\pm 0.0065}$	$0.5370_{\pm 0.0067}$	$0.8085_{\pm 0.0009}$	$0.6306_{\pm 0.0032}$	$0.6556_{\pm 0.0026}$
Multi-IPS	$0.7109_{\pm 0.0005}$	$0.7120_{\pm 0.0019}$	$0.4687_{\pm 0.0024}$	$0.7337_{\pm 0.0033}$	$0.6939_{\pm 0.0048}$	$0.5391_{\pm 0.0051}$	$0.7683_{\pm 0.0022}$	$0.5785_{\pm 0.0032}$	$0.6253_{\pm 0.0018}$
Multi-DR	$0.7168_{\pm 0.0005}$	$0.7212_{\pm 0.0009}$	$0.4766_{\pm 0.0022}$	$0.7556_{\pm 0.0026}$	$0.6921_{\pm 0.0040}$	$0.5252_{\pm 0.0056}$	$0.7767_{\pm 0.0011}$	$0.5811_{\pm 0.0036}$	$0.6309_{\pm 0.0063}$
ESCM2-IPS	$0.7106_{\pm 0.0008}$	$0.7113_{\pm 0.0018}$	$0.4676_{\pm 0.0020}$	$0.7392_{\pm 0.0026}$	$0.6872_{\pm 0.0040}$	$0.5294_{\pm 0.0056}$	$0.7766_{\pm 0.0030}$	$0.5817_{\pm 0.0056}$	$0.6467_{\pm 0.0039}$
ESCM2-DR	$0.7174_{\pm 0.0006}$	$0.7270_{\pm 0.0008}$	$0.4780_{\pm 0.0015}$	$0.7337_{\pm 0.0054}$	$0.6939_{\pm 0.0088}$	$0.5391_{\pm 0.0051}$	$0.7812_{\pm 0.0022}$	$0.5866_{\pm 0.0032}$	$0.6479_{\pm 0.0042}$
DDPO	$0.7204_{\pm 0.0004}$	$0.7266_{\pm 0.0012}$	$0.4788_{\pm 0.0010}$	$\underline{0.7612_{\pm 0.0038}}$	$0.7059_{\pm 0.0032}$	$0.5339_{\pm 0.0054}$	$0.8174_{\pm 0.0013}$	$0.6199_{\pm 0.0033}$	$0.6800_{\pm 0.0036}$
BADR	$\mathbf{0.7332}^{*}_{\pm 0.0009}$	$\mathbf{0.7831}^{*}_{\pm 0.0011}$	$\mathbf{0.5382}^{*}_{\pm 0.0012}$	$\mathbf{0.7694}_{\pm 0.0030}$	$\mathbf{0.7417}^{*}_{\pm 0.0009}$	$\mathbf{0.5750}^{*}_{\pm 0.0041}$	$\mathbf{0.8391}^{*}_{\pm 0.0011}$	$\mathbf{0.6689}^{*}_{\pm 0.0021}$	$\mathbf{0.7189}^{*}_{\pm 0.0061}$
Impv (%)	1.37	3.04	4.10	1.08	3.56	3.57	1.48	1.87	2.83

Bold and underlined fonts indicate the best and second-best methods, respectively. "Impv" indicates the percentage improvement of BADR over the best baseline.
*Indicates statistically significant results (p-value $\leq$ 0.05) based on the paired t-test.

— The DR methods that consider variance/bias control (MRDR, DR-BIAS, DR-MSE, GPL, and D-DR) outperform DRJL by reducing variance, bias, or both. This observation highlights the benefits of variance/bias control on debiasing, which achieves better optimization of the ideal loss than DRJL. This aligns with our theoretical analysis in Section 3.1, which indicates that the DR estimator used by these methods suffers from high bias, variance, and generalization error bound under sparse data. However, they do not optimize the estimator's formulation in batch training, leaving them still at risk of high or unbounded bias, variance, and generalization error bound.
— DCE-TDR achieves the best performance among baselines due to its calibrated propensity scores and imputation errors. However, solely refining propensity and imputation accuracy is insufficient for effectively debiasing CVR prediction under sparse data. DCE-TDR overlooks the problem of the sensitivity of the DR estimator to small propensity scores.
— Compared to TDR-CL and GPL, DAMF shows comparable performance and even surpasses them in NDCG@K and Recall@K on Yahoo! R3 and KuaiRec, demonstrating the effectiveness of propensity-free methods. One reason for its success is avoiding issues from propensity

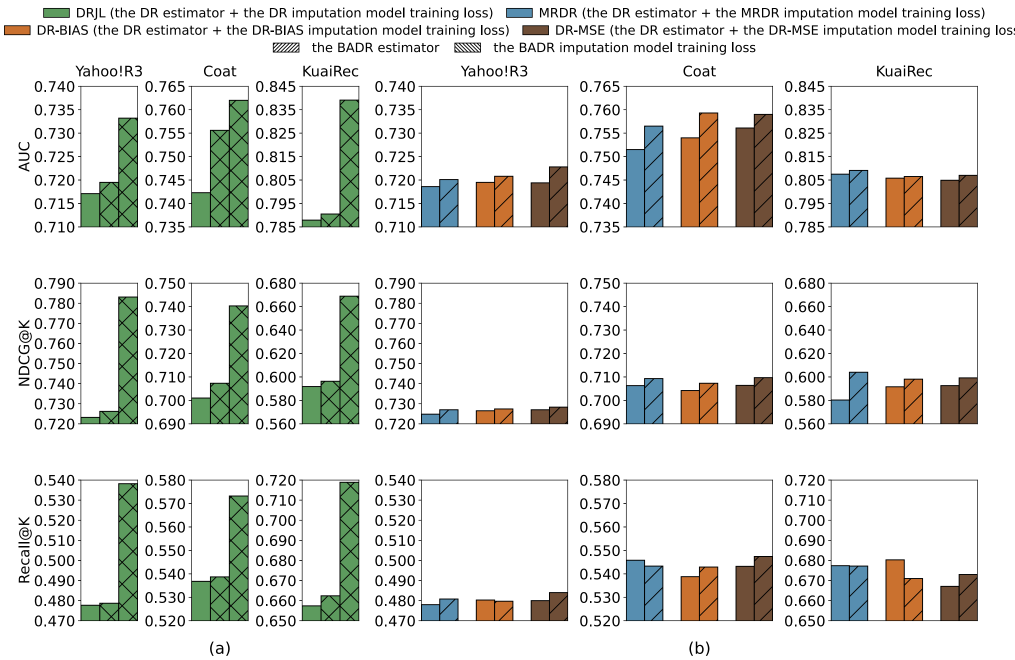

Fig. 1. Ablation study on Yahoo! R3, Coat, and KuaiRec. The bars filled with different colors and shading represent different combinations of modules. For example, a bar filled with blue and "/" represents the BADR estimator + the MRDR imputation model training loss.

and imputation. However, DAMF underperforms DCE-TDR, emphasizing the importance of addressing issues of DR for debiasing CVR prediction.

— BADR consistently outperforms all baselines across all datasets and metrics. Particularly in the top-K ranking metrics, NDCG@K and Recall@K, BADR demonstrates more significant improvements. This superiority highlights the effectiveness of BADR for debiasing. The success of BADR stems from its enhanced control over bias, variance, and generalization error bound under sparse data, enabling superior optimization of the ideal loss.

4.3 Ablation Study (RQ2)

To evaluate the impact of the BADR estimator and the BADR imputation model training loss, we develop two variants of BADR: (1) BADR w/o the BADR imputation model training loss (the BADR estimator + the DR imputation model training loss); (2) BADR w/o both the BADR estimator and the BADR imputation model training loss (DRJL). Figure 1(a) presents a performance comparison between BADR and its variants on Yahoo! R3, Coat, and KuaiRec, with results averaged over five runs. From the figure, we observe that:

— BADR w/o the BADR imputation model training loss outperforms DRJL across all datasets and metrics, validating the effectiveness of the BADR estimator for debiasing.

— BADR significantly outperforms its two variants, demonstrating that the DR imputation model training losses are incompatible with the BADR estimator, and the variance-reduction imputation model training loss is crucial for achieving optimal performance with the BADR estimator.

To further validate the necessity of designing a theoretically compatible variance-reduction imputation model training loss for the BADR estimator, we constructed three additional BADR variants by replacing the original BADR imputation model training loss with alternative variance/bias-reduction imputation model training losses: (1) the BADR estimator + the MRDR imputation model training loss (BADR-V); (2) the BADR estimator + the DR-BIAS imputation model training loss (BADR-B); (3) the BADR estimator + the DR-MSE imputation model training loss (BADR-M). The reasons for choosing these three baselines are as follows: MRDR, DR-BIAS, and DR-MSE are representative methods that control variance, bias, and both, respectively, by adjusting the imputation model training loss. From Figure 1(b), we observe:

— BADR-V and BADR-B outperform MRDR and DR-BIAS, respectively, in AUC and NDCG@K, but sometimes underperform in Recall@K. Meanwhile, BADR-M outperforms DR-MSE across all datasets and metrics. These results reconfirm that the BADR estimator can improve debiasing performance. The reason for the occasional poor performance is that the imputation model training losses of MRDR and DR-BIAS are incompatible with the BADR estimator.
— The performance of BADR-V, BADR-B, and BADR-M significantly lags behind that of BADR, once again suggesting that the imputation model training losses of MRDR, DR-BIAS, and DR-MSE are not well-suited for the BADR estimator. These results highlight the need for the BADR imputation model training loss for BADR's algorithmic robustness, as derived from our theory in Section 3.4, which is theoretically compatible with the BADR estimator and fully activates its debiasing potential.

4.4 Sparsity Study (RQ3)

To investigate the performance of BADR on sparser data, we sample the datasets with varying sparsity levels for training. Specifically, we sample from Yahoo! R3 and KuaiRec at rates of 80%, 60%, 40%, and 20% to create sparser versions of each dataset. This yields: (1) Yahoo! R3 with sparsity rates of 98.38%, 98.79%, 99.19%, and 99.60% (for 80%, 60%, 40%, and 20% sampling rates, respectively); (2) KuaiRec with sparsity rates of 96.57%, 97.43%, 98.29%, and 99.14% (for 80%, 60%, 40%, and 20% sampling rates, respectively). Due to the limited amount of data in Coat, we do not conduct this sparsity study on it. For each method, we run it five times using five different random seeds. Tables 4 and 5 report the mean and standard deviation of the results for each method. From the tables, we observe that: (1) BADR outperforms the baselines across different sparsity levels on both Yahoo! R3 and KuaiRec for all metrics, demonstrating its robustness to varying sparsity levels; (2) Even under extreme sparsity (99.60% in Yahoo! R3 and 99.14% in KuaiRec), BADR shows significant improvements, indicating its effectiveness under extreme sparse data. These results across different sparsity levels further validate our theoretical analysis in Section 3.

4.5 The Study of Loss on Unbiased Data (RQ4)

The ranking performance metrics in recommendation (AUC, NDCG@K, and Recall@K) discussed in the experiments above do not directly align with the point-wise loss used for BADR. To validate whether BADR achieves superior optimization of the ideal loss compared to baselines (as claimed in Section 3.3), we present loss-epoch curves in Figure 2. It is worth noting that computing the ideal loss requires conversion labels for the entire user–item space, which is infeasible. Therefore, we use the loss on unbiased data as a proxy. The unbiased data can be regarded as a uniform sample from the entire user–item space with available conversion labels, providing an unbiased estimate of the ideal loss. In particular, we compute the average prediction error on unbiased data with the cross-entropy loss function. These curves are generated under each method's optimal hyperparameter configuration and early stopping criteria. From the figure, we observe that BADR

Table 4. Sparsity Study on the Data Sampled from Yahoo! R3 at Sampling Rates of 80%, 60%, 40%, and 20%

	80% (98.38%)			60% (98.79%)			40% (99.19%)			20% (99.60%)		
Method	AUC	NDCG@5	Recall@5	AUC	NDCG@5	Recall@5	AUC	NDCG@5	Recall@5	AUC	NDCG@5	Recall@5
MF	0.7046	0.7070	0.4639	0.6944	0.6937	0.4532	0.6700	0.6758	0.4381	0.6328	0.6495	0.4128
DRJL	0.7123	0.7210	0.4744	0.7021	0.7108	0.4680	0.6843	0.6930	0.4489	0.6392	0.6523	0.4137
MRDR	0.7146	0.7219	0.4758	0.7036	0.7110	0.4679	0.6839	0.6939	0.4508	0.6403	0.6515	0.4125
DR-BIAS	0.7139	0.7221	0.4769	0.7028	0.7110	0.4675	0.6847	0.6925	0.4499	0.6395	0.6521	0.4133
TDR-CL	0.7140	0.7242	0.4796	0.7024	0.7110	0.4691	0.6846	0.6919	0.4495	0.6470	0.6545	0.4135
GPL	0.7187	0.7307	0.4820	0.7055	0.7140	0.4697	0.6864	0.6953	0.4534	0.6467	0.6610	0.4227
DCE-TDR	0.7222	0.7635	0.5190	0.7095	0.7486	0.5036	0.6893	0.7334	0.4890	0.6422	0.6874	0.4485
	± 0.0002	± 0.0016	± 0.0023	± 0.0006	± 0.0023	± 0.0016	± 0.0012	± 0.0018	± 0.0024	± 0.0013	± 0.0018	± 0.0017
BADR	**0.7246***	**0.7805***	**0.5372***	**0.7119***	**0.7730***	**0.5314***	**0.6934***	**0.7551***	**0.5164***	**0.6484***	**0.7065***	**0.4707***
	± 0.0008	± 0.0005	± 0.0014	± 0.0005	± 0.0005	± 0.0008	± 0.0010	± 0.0007	± 0.0012	± 0.0018	± 0.0030	± 0.0018
Impv-MF (%)	2.83	10.40	15.80	2.52	11.43	17.26	3.49	11.73	17.87	2.47	8.78	14.03
Impv (%)	0.32	2.23	3.51	0.34	3.26	5.52	0.60	2.96	5.60	0.22	2.78	4.95

The values in parentheses are the corresponding data sparsity rates. bold and underlined fonts indicate the best and second-best methods, respectively. "Impv-MF" and "Impv" indicate the improvement of BADR over MF and the best baseline, respectively.
*Indicates statistically significant results (p-value $\leq$ 0.05) based on the paired t-test.

Table 5. Sparsity Study on the Data Sampled from KuaiRec at Sampling Rates of 80%, 60%, 40%, and 20%

	80% (96.57%)			60% (97.43%)			40% (98.29%)			20% (99.14%)		
Method	AUC	NDCG@50	Recall@50	AUC	NDCG@50	Recall@50	AUC	NDCG@50	Recall@50	AUC	NDCG@50	Recall@50
MF	0.7411	0.5656	0.6144	0.7154	0.5538	0.5914	0.7137	0.5461	0.5988	0.6753	0.5279	0.5734
DRJL	0.7788	0.5798	0.6538	0.7708	0.5747	0.6433	0.7419	0.5484	0.6263	0.7017	0.5318	0.6049
MRDR	0.7950	0.5899	0.6673	0.7743	0.5751	0.6466	0.7455	0.5521	0.6302	0.7048	0.5330	0.6075
DR-BIAS	0.7959	0.5943	0.6676	0.7744	0.5703	0.6461	0.7419	0.5539	0.6339	0.7062	0.5339	0.6118
TDR-CL	0.8006	0.5515	0.6711	0.7771	0.5492	0.6567	0.7498	0.5511	0.6207	0.6841	0.5327	0.5682
GPL	0.8039	0.5970	0.6615	0.7771	0.5775	0.6337	0.7524	0.5640	0.6284	0.7084	0.5292	0.6053
DCE-TDR	0.8204	0.6472	0.6749	0.8106	0.6429	0.6828	0.8053	0.6376	0.6667	0.7627	0.6048	0.6317
	± 0.0015	± 0.0031	± 0.0065	± 0.0014	± 0.0027	± 0.0050	± 0.0025	± 0.0034	± 0.0085	± 0.0037	± 0.0036	± 0.0051
BADR	**0.8371***	**0.6684***	**0.7184***	**0.8340***	**0.6651***	**0.7088***	**0.8291***	**0.6594***	**0.7057***	**0.8126***	**0.6438***	**0.6854***
	± 0.0009	± 0.0033	± 0.0018	± 0.0012	± 0.0016	± 0.0030	± 0.0007	± 0.0025	± 0.0070	± 0.0025	± 0.0014	± 0.0061
Impv-MF (%)	12.96	18.18	16.93	16.58	20.10	19.85	16.17	20.75	17.85	20.33	21.96	19.53
Impv (%)	2.04	3.28	6.45	2.89	3.45	3.81	2.96	3.42	5.85	6.54	6.45	8.50

The values in parentheses are the corresponding data sparsity rates. Bold and underlined fonts indicate the best and second-best methods, respectively. "Impv-MF" and "Impv" indicate the improvement of BADR over MF and the best baseline, respectively.
*Indicates statistically significant results (p-value $\leq$ 0.05) based on the paired t-test.

consistently attains lower ideal loss values at convergence than the strongest baseline DCE-TDR across all datasets and all sparsity conditions, demonstrating that BADR more effectively minimizes loss on unbiased data, as expected.

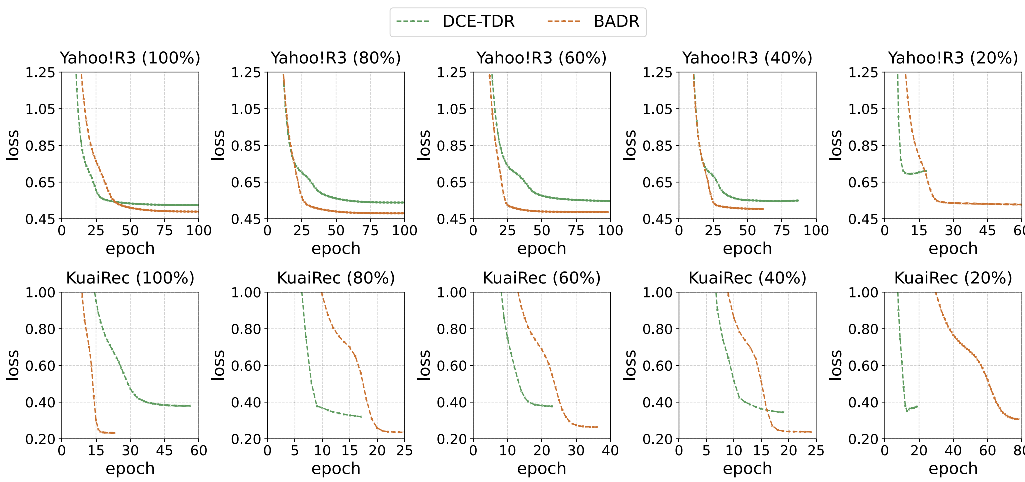

Fig. 2. Loss on unbiased data from Yahoo! R3 and KuaiRec.

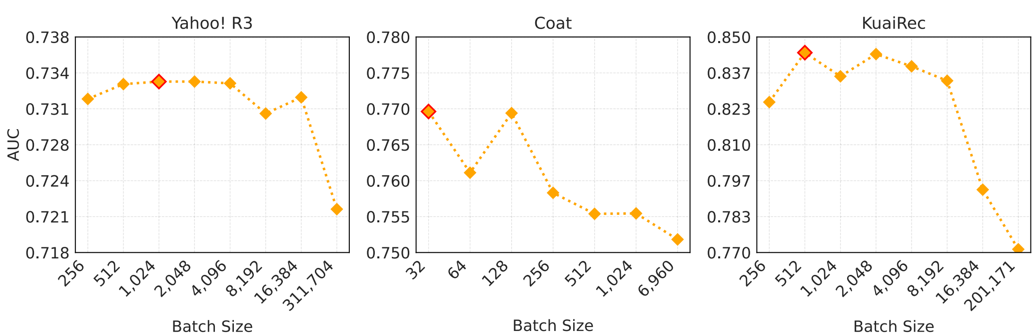

Fig. 3. Batch size analysis on the Yahoo! R3, Coat, and KuaiRec.

4.6 Batch Size Study (RQ5)

To evaluate the impact of batch size on BADR's debiasing performance, we compare the AUC across different batch sizes. The experimental settings are as follows: for Yahoo! R3, the batch sizes are [256, 512, 1,024, 2,048, 4,096, 8,192, 16,384, 311,704]; for KuaiRec, the batch sizes are [256, 512, 1,024, 2,048, 4,096, 8,192, 16,384, 201,171]; and for Coat, due to its relatively small scale, the batch sizes are [32, 64, 128, 256, 512, 1,024, 6,960]. Figure 3 presents the experimental results, and the main observations are as follows:

— On Yahoo! R3 and KuaiRec, BADR's AUC reaches its peak or near-peak values at medium-scale batch sizes (512–8,192). This indicates that medium batch sizes are more favorable for BADR debiasing than extremely small or extremely large ones. The reason is that medium batch sizes strike a good balance among sufficient gradient noise (which serves as regularization), stable gradient estimation, an appropriate number of model updates within each epoch, and the unbiasedness of BADR.

— On Coat (a small dataset), smaller batches (32–128) perform better, while performance gradually degrades as batch size increases. This suggests that for small datasets, smaller batches provide stronger gradient stochasticity, which helps resist overfitting; at the same time, smaller batch sizes enable BADR to perform the required number of updates within an epoch.

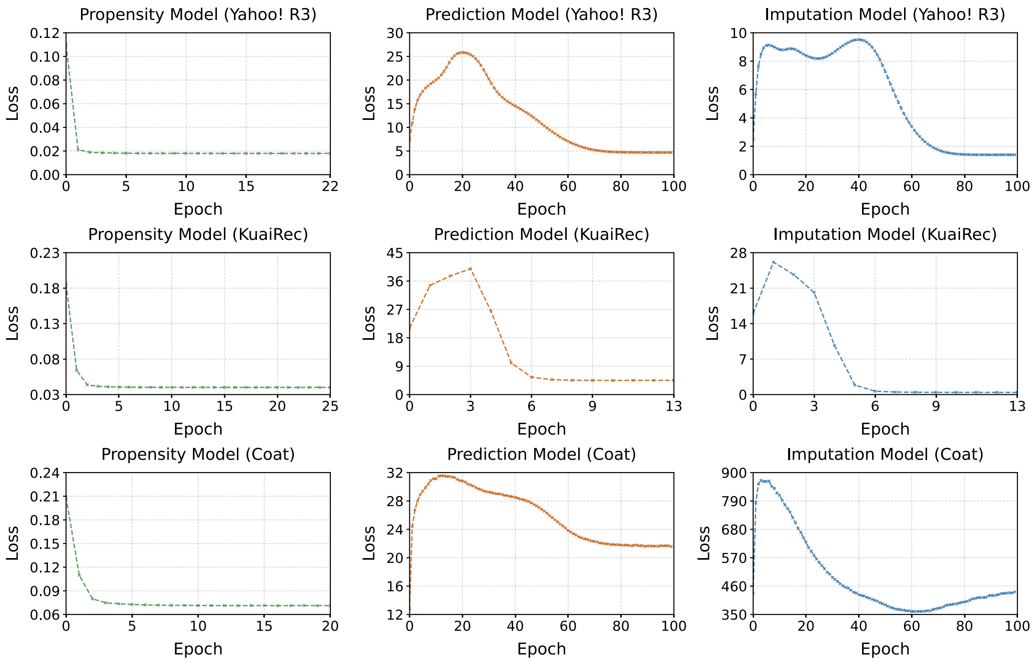

Fig. 4. Training loss curves of the propensity, prediction, and imputation models of BADR on the Yahoo! R3, Coat, and KuaiRec.

— When the batch size equals the dataset size, BADR exhibits the lowest debiasing performance across all three datasets. This suggests that it weaken BADR's debiasing ability: on the one hand, they remove the beneficial effect of gradient noise on generalization; on the other hand, with parameter updates occurring only once per epoch, BADR's batch adaptiveness cannot take effect. Nevertheless, even in such extreme cases, BADR retains some debiasing capability—for instance, on Yahoo! R3 it still outperforms several debiasing methods (e.g., DRJL and MRDR).

4.7 Convergence Behavior Analysis (RQ6)

To analyze the convergence behavior of the propensity, prediction, and imputation models in BADR, we plot their loss-epoch curves on Yahoo! R3, Coat, and KuaiRec (Figure 4). Our implementation is inspired by [19], but it does not strictly normalize the losses of the prediction and imputation models to the per-sample scale. As a result, the absolute loss values may appear large and are not directly comparable across models. Therefore, we primarily rely on the trends of the loss curves and the final evaluation metrics (e.g., AUC) to assess convergence. Based on Figure 4, we have the following observations:

— Across all datasets, the propensity model exhibits smoothly decreasing and convergent loss curves, suggesting that its training is highly stable. This is because the propensity model training loss is a single cross-entropy loss and does not involve complex model updates.
— Under joint training, the prediction and imputation models typically exhibit a "rise-then-fall" pattern in their training losses. This behavior likely arises because, in the early stages, both models rapidly adjust their parameters while being mutually dependent (i.e., the prediction model relies on the imputation model's outputs, whereas the imputation model depends on

Table 6. Performance Comparison on Simulated Datasets with Density Ratios of 10.00% and 15.11%

	Simulated ML-100K (10.00%)							Simulated ML-100K (15.11%)						
Method	AUC	NDCG@5	Recall@5	Precision@5	NDCG@50	Recall@50	Precision@50	AUC	NDCG@5	Recall@5	Precision@5	NDCG@50	Recall@50	Precision@50
MF	0.8680	0.8907	0.0323	0.8791	0.7401	0.2207	0.6927	0.8928	0.8976	0.0365	0.8861	0.7498	0.2516	0.6999
	± 0.0002	± 0.0014	± 0.0001	± 0.0019	± 0.0006	± 0.0003	± 0.0008	± 0.0005	± 0.0023	± 0.0003	± 0.0030	± 0.0009	± 0.0007	± 0.0007
DCE-TDR	0.8757	0.9010	0.0328	0.8912	0.7571	0.2263	0.7119	0.9008	0.9177	0.0370	0.9086	0.7820	0.2603	0.7353
	± 0.0007	± 0.0011	± 0.0001	± 0.0016	± 0.0013	± 0.0006	± 0.0014	± 0.0015	± 0.0029	± 0.0005	± 0.0031	± 0.0021	± 0.0023	± 0.0021
BADR	**0.8909***	**0.9098***	**0.0335***	**0.8988***	**0.7735***	**0.2321***	**0.7291***	**0.9088***	**0.9240***	**0.0378***	**0.9151***	**0.7896***	**0.2666***	**0.7426***
	± 0.0007	± 0.0021	± 0.0002	± 0.0022	± 0.0007	± 0.0005	± 0.0007	± 0.0005	± 0.0007	± 0.0002	± 0.0013	± 0.0014	± 0.0009	± 0.0016
Impv-MF (%)	2.64	2.14	3.72	2.24	4.51	5.17	5.25	1.79	2.94	3.56	3.27	5.31	5.96	6.10
Impv (%)	1.74	0.98	2.13	0.85	2.17	2.56	2.42	0.89	0.69	2.16	0.72	0.97	2.42	0.99

Bold and underlined fonts indicate the best and second-best methods, respectively. "Impv-MF" and "Impv" indicate the improvement of BADR over MF and the best baseline, respectively.
*Indicates statistically significant results (p-value $\leq$ 0.05) based on the paired t-test

the prediction model's outputs). Such dependency can cause temporary fluctuations during the first few epochs. As training progresses, the two models gradually reach compatible states, their outputs stabilize, and the losses decrease and converge.

— On Coat, however, the final losses of both the prediction and imputation models do not fall below their initial values. This is likely because the prediction model's loss is initially low due to the imputed errors at the start. Since both models are untrained at initialization, this initial low loss does not indicate full convergence. Therefore, we rely on the loss trends during training and the AUC achieved by the prediction model to assess convergence.

These observations highlight interesting phenomena in joint learning for debiasing recommendation, especially on small datasets like Coat. We plan to further investigate how to design more stable joint learning strategies to improve debiasing performance in future work.

4.8 Experiments on High-Propensity-Score Scenarios (RQ7)

To evaluate the performance of BADR under higher average propensity scores, we conduct experiments on simulated datasets. This is because, among the commonly used real-world datasets, Coat has the highest average propensity score of 0.08 [47] (i.e., data density), which cannot be used for investigating the performance of BADR under high-propensity-score scenarios. Thus, following the simulation approach in MRDR, we construct a simulated dataset based on ML-100K.[7] The minimum propensity scores in these datasets are 0.062 and 0.113, and the average propensity scores are 0.094 and 0.151, respectively. Table 6 reports the performance of MF, DCE-TDR, and BADR on these denser datasets. From the results, we observe: (1) Even under high-propensity-score scenarios, BADR still outperforms MF and DCE-TDR, highlighting its effectiveness. This is because, although the minimum propensity scores are higher than those in sparse datasets (e.g., Coat has a minimum propensity score of 0.0107), they may still lead to high bias and variance. (2) Compared to low-propensity-score datasets, the relative improvement of BADR over the strongest baseline, DCE-TDR, is smaller, suggesting that BADR is particularly advantageous in low-propensity-score scenarios (i.e., sparse data scenarios). (3) The relative improvement of BADR and DCE over MF is smaller in high-propensity-score datasets. The reason is that the bias patterns in the simulated data are simpler than those in real datasets, and the high-propensity-score datasets (i.e., more labeled samples can be observed) already provide sufficient information for CVR prediction.

[7] https://grouplens.org/datasets/movielens/100k/.

Table 7. Online Performance over 7 Days

	Clicks (p-value = 0.0001)		Conversions (p-value = 0.0002)	
Day	Baseline	BADR	Baseline	BADR
1	+0.00%	+0.39%	+0.00%	+0.28%
2	+0.00%	+0.83%	+0.00%	+0.71%
3	+0.00%	+0.73%	+0.00%	+0.52%
4	+0.00%	+0.88%	+0.00%	+0.42%
5	+0.00%	+1.23%	+0.00%	+0.91%
6	+0.00%	+0.62%	+0.00%	+0.76%
7	+0.00%	+1.01%	+0.00%	+0.62%

4.9 Online Experiment (RQ8)

To evaluate the performance of integrating BADR into a real-world industrial framework, we conduct online A/B experiments in a large-scale e-commerce marketing scenario. Our online setting uses a multi-task framework. Due to the company's confidentiality policies, we cannot disclose the specific framework and baseline we used. The integration of BADR into the multi-task framework follows the discussion in Section 3.7. Table 7 reports clicks and conversions over a seven-day period. Because of business requirements, we evaluate performance directly using clicks and conversions rather than normalized metrics such as CTR and CVR. These results show that BADR consistently outperforms the online baseline, highlighting BADR's effectiveness in real-world online scenarios.

5 Related Work

In this section, we introduce the literature on debiasing CVR prediction.

5.1 Debiasing in Recommendation

Selection bias is one of the most common biases in recommendation, leading to the training data distribution differing from the target distribution [31, 65]. One of the most popular research directions is based on estimators of the ideal loss, which are broadly classified into three categories: EIB [13, 43, 51], IPS [47, 53, 66], and DR learning [27, 28, 44, 57]. EIB imputes errors for missing data and trains a recommendation model using both the imputed and observational data. IPS reweights prediction errors on observational data using IPSs. Various strategies are used to estimate propensity scores: statistical values of observational data (e.g., user activity and item popularity) [43], naive Bayes [26, 47], logistic regression [10, 19], and boosting for balancing properties [25, 26]. DR combines the advantages of IPS and EIB to achieve double robustness. Extensive work has been proposed to enhance DR: achieving multiple robustness [21], removing hidden confounders [6], improving stability [29], reducing bias/variance [5, 10, 12, 69], enhancing robustness to inaccurate imputation [22, 50], and improving propensity/imputation estimation accuracy [19], among others. Unlike existing methods for reducing variance and bias, BADR redesigns the formulation of the DR estimator in batch training and derives a variance-reduction imputation model training loss that is compatible with it. Consequently, BADR theoretically has smaller bias, variance, and generalization bound than the DR estimator under small propensity scores, all of which are bounded.

Another line of research in debiasing recommendation focuses on debiasing without designing an ideal loss estimator. A variety of techniques are employed in this line, including information bottleneck [33, 60], domain adaptation [38, 45, 61], and balancing [63]. For example, [60] extends the original information bottleneck [55] to frame the debiasing recommendation problem and derive a contrastive loss, facilitating balanced learning between observational and missing data. [45] derive an upper bound for the error on unbiased data using unsupervised domain adaptation theory [15] and minimize it through adversarial learning. Our work follows the DR paradigm and aims to achieve greater robustness in batch training under sparse data.

There are other methods for debiasing in recommendation that operate in different settings from ours. Ren et al. [41] extended IPS to pairwise loss. Saito et al. [46] apply IPS in the implicit feedback setting. Previous works [2, 23, 24, 32, 58] leverage random exposure data to assist in model training. For example, with the assistance of a small amount of random exposure data, Liu et al. [32] distill knowledge from random exposure data into models trained on biased data, Chen et al. [2] propose a meta-learning framework to correct propensity and imputation model parameters, and Li et al. [23] propose a residual fitting task for better prediction.

5.2 CVR Prediction

Recently, several studies have focused on modeling and debiasing CVR in a multi-task setting [14, 23, 35, 52, 56, 67, 68, 70], where CTR prediction is incorporated to provide auxiliary information for CVR prediction. Multi-DR [67] updates the CTR, CVR, and an imputation model directly using the DR estimator. ESMM [35] introduces the entire space loss, which leverages the CTCVR to link CTR and CVR by modeling CTCVR as CTR × CVR. This entire space loss has become a foundational component and has been adopted in subsequent multi-task work. ESCM^2-DR [56] incorporates the DR estimator into ESMM to further debias the CVR estimation. UKD [62] generates pseudo-conversion labels via knowledge distillation and optimizes the CVR model similarly to EIB. AECM [68] combines task-invariant representation learning, space-invariant representation learning, and DR-MSE [5] to enhance the debiasing of CVR prediction. DDPO [52] introduces a click propensity network and a conversion propensity network to unbiasedly optimize the CVR model using both clicked and unclicked data. Additionally, it reinforces the causal relationship between networks via attention mechanisms. In this work, we deliberately avoid using a multi-task setting to isolate and evaluate the effectiveness of debiasing methods, excluding potential performance gains attributed to multi-task learning. Nevertheless, our method can be seamlessly integrated into a multi-task framework if desired.

5.3 Deep Learning Architecture for CVR Prediction

Besides addressing bias in CVR prediction, another important research direction is the design of deep learning architectures for CVR modeling. These architectures are often closely intertwined with CTR prediction models. Early studies primarily rely on traditional models such as logistic regression [59] and gradient boosting decision trees [59], which are limited in capturing complex feature interactions. FM [42] and its extensions, such as FFM [16], effectively model second-order feature interactions, achieving significant improvements in both CTR and CVR tasks. Subsequent deep learning models, including Wide & Deep [4], DeepFM [9], and xDeepFM [30], combine explicit linear features with deep nonlinear feature interactions, enabling automatic learning of high-order feature combinations. In addition, multi-task learning methods such as MMOE [34] and PLE [54] share underlying feature representations to collaboratively learn CTR and CVR tasks, further enhancing CVR prediction accuracy. To capture sequential and graph structures in the data, both sequence-based [49, 64] and graph-based [11, 20, 39] methods have been proposed. For example, TWIN-v2 [49] leverages attention mechanisms to model ultra-long user behavior sequences. Meanwhile, methods such as BADR provide a prototypical method for debiasing by focusing on improvements to the training loss. It is worth noting that bias correction in CVR prediction is orthogonal to these architecture-focused methods.

6 Conclusion

In this article, we addressed the challenge of debiasing CVR prediction when click events are sparse. We theoretically demonstrated that existing DR learning methods are highly sensitive to small propensity scores and consequently encounter high or even unbounded bias, variance, and generalization error bound. To this end, we proposed a novel BADR method. In particular, we proposed a principled BADR estimator that adjusts each data batch's influence on debiasing CVR prediction by considering the propensity scores within the batch. Our theoretical analysis revealed that, compared to the existing DR estimator, the proposed BADR estimator exhibits bounded and reduced bias, variance, and generalization error bound in the presence of small propensity scores. We then derived a new batch-adaptive imputation model training loss, compatible with the BADR estimator, to theoretically improve algorithmic robustness. Finally, we implemented BADR by extending the widely used joint learning framework for DR methods, which jointly optimizes the CVR prediction model and the error imputation model. Our experiments validated the effectiveness and rationality of BADR for debiasing CVR prediction under varying sparsity levels of click events.

Future work includes extending BADR with more accurate propensity score estimation and error imputation methods, as well as exploring its integration with more complex model architectures. Beyond these directions, BADR also holds potential in real-time inference scenarios, where fast and unbiased CVR predictions are critical. One possible application is to apply BADR to streaming user interactions, where the influence of each incoming data batch is adjusted adaptively based on the learned propensity scores. In this setup, the propensity and imputation models are collaboratively trained alongside the prediction model within a multi-task learning framework. Additionally, BADR could be extended to federated learning settings, enabling debiased CVR prediction across distributed user data while preserving privacy.

References

[1] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. 2019. Optuna: A next-generation hyperparameter optimization framework. In *KDD*, 2623–2631.

[2] Jiawei Chen, Hande Dong, Yang Qiu, Xiangnan He, Xin Xin, Liang Chen, Guli Lin, and Keping Yang. 2021. AutoDebias: Learning to debias for recommendation. In *SIGIR*, 21–30.

[3] Jiawei Chen, Hande Dong, Xiang Wang, Fuli Feng, Meng Wang, and Xiangnan He. 2023. Bias and Debias in recommender system: A survey and future directions. *ACM Transactions on Information Systems* 41, 3 (2023), 1–39.

[4] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispiret al. 2016. Wide & deep learning for recommender systems. In *RecSys WorkShop*, 7–10.

[5] Quanyu Dai, Haoxuan Li, Peng Wu, Zhenhua Dong, Xiao-Hua Zhou, Rui Zhang, Rui Zhang, and Jie Sun. 2022. A generalized doubly robust learning framework for debiasing post-click conversion rate prediction. In *KDD*, 252–262.

[6] Sihao Ding, Peng Wu, Fuli Feng, Yitong Wang, Xiangnan He, Yong Liao, and Yongdong Zhang. 2022. Addressing unmeasured confounder for recommendation with sensitivity analysis. In *KDD*, 305–315.

[7] Chongming Gao, Shijun Li, Wenqiang Lei, Jiawei Chen, Biao Li, Peng Jiang, Xiangnan He, Jiaxin Mao, and Tat-Seng Chua. 2022. KuaiRec: A fully-observed dataset and insights for evaluating recommender systems. In *CIKM*, 540–550.

[8] Chen Gao, Yu Zheng, Wenjie Wang, Fuli Feng, Xiangnan He, and Yong Li. 2024. Causal inference in recommender systems: A survey and future directions. *ACM Transactions on Information Systems* 42, 4 (2024), 1–32.

[9] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. DeepFM: A factorization-machine based neural network for CTR prediction. In *IJCAI*, 1725–1731.

[10] Siyuan Guo, Lixin Zou, Yiding Liu, Wenwen Ye, Suqi Cheng, Shuaiqiang Wang, Hechang Chen, Dawei Yin, and Yi Chang. 2021. Enhanced doubly robust learning for debiasing post-click conversion rate estimation. In *SIGIR*, 275–284.

[11] Wei Guo, Rong Su, Renhao Tan, Huifeng Guo, Yingxue Zhang, Zhirong Liu, Ruiming Tang, and Xiuqiang He. 2021. Dual graph enhanced embedding neural network for CTR prediction. In *KDD*, 496–504.

[12] Mingming Ha, Xuewen Tao, Wenfang Lin, Qionxu Ma, Wujiang Xu, and Linxun Chen. 2024. Fine-Grained dynamic framework for bias-variance joint optimization on data missing not at random. In *NeurIPS*, 104010–104034.

[13] José Miguel Hernández-Lobato, Neil Houlsby, and Zoubin Ghahramani. 2014. Probabilistic matrix factorization with non-random missing data. In *ICML*, 1512–1520.

[14] Jiahui Huang, Lan Zhang, Junhao Wang, Shanyang Jiang, Dongbo Huang, Cheng Ding, and Lan Xu. 2024. Utilizing non-click samples via semi-supervised learning for conversion rate prediction. In *RecSys*, 350–359.
[15] Junguang Jiang, Yang Shu, Jianmin Wang, and Mingsheng Long. 2022. Transferability in deep learning: A Survey. *CoRR* abs/2201.05867 (2022).
[16] Yu-Chin Juan, Yong Zhuang, Wei-Sheng Chin, and Chih-Jen Lin. 2016. Field-aware factorization machines for CTR prediction. In *RecSys*, 43–50.
[17] Haruka Kiyohara, Yuta Saito, Tatsuya Matsuhiro, Yusuke Narita, Nobuyuki Shimizu, and Yasuo Yamamoto. 2022. Doubly robust off-policy evaluation for ranking policies under the Cascade behavior model. In *ESDM*, 487–497.
[18] Yehuda Koren, Robert M. Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 42, 8 (2009), 30–37.
[19] Wonbin Kweon and Hwanjo Yu. 2024. Doubly calibrated estimator for recommendation on data missing not at random. In *WWW*, 3810–3820.
[20] Chaozhuo Li, Bochen Pang, Yuming Liu, Hao Sun, Zheng Liu, Xing Xie, Tianqi Yang, Yanling Cui, Liangjie Zhang, and Qi Zhang. 2021. AdsGNN: Behavior-graph augmented relevance modeling in sponsored search. In *SIGIR*, 223–232.
[21] Haoxuan Li, Quanyu Dai, Yuru Li, Yan Lyu, Zhenhua Dong, Xiao-Hua Zhou, and Peng Wu. 2023. Multiple robust learning for recommendation. In *AAAI*, 4417–4425.
[22] Haoxuan Li, Yan Lyu, Chunyuan Zheng, and Peng Wu. 2023. TDR-CL: Targeted doubly robust collaborative learning for debiased recommendations. In *ICLR*.
[23] Haoxuan Li, Kunhan Wu, Chunyuan Zheng, Yanghao Xiao, Hao Wang, Zhi Geng, Fuli Feng, Xiangnan He, and Peng Wu. 2023. Removing hidden confounding in recommendation: A unified multi-task learning approach. In *NeurIPS*, 54614–54626.
[24] Haoxuan Li, Yanghao Xiao, Chunyuan Zheng, and Peng Wu. 2023. Balancing unobserved confounding with a few unbiased ratings in debiased recommendations. In *WWW*, 1305–1313.
[25] Haoxuan Li, Yanghao Xiao, Chunyuan Zheng, Peng Wu, and Peng Cui. 2023. Propensity matters: Measuring and enhancing balancing for recommendation. In *ICML*, 20182–20194.
[26] Haoxuan Li, Yanghao Xiao, Chunyuan Zheng, Peng Wu, Zhi Geng, Xu Chen, and Peng Cui. 2024. Debiased collaborative filtering with kernel-based causal balancing. In *ICLR*.
[27] Haoxuan Li, Chunyuan Zheng, Sihao Ding, Peng Wu, Zhi Geng, Fuli Feng, and Xiangnan He. 2024. Be aware of the neighborhood effect: Modeling selection bias under interference for recommendation. In *ICLR*.
[28] Haoxuan Li, Chunyuan Zheng, Shuyi Wang, Kunhan Wu, Eric Wang, Peng Wu, Zhi Geng, Xu Chen, and Xiao-Hua Zhou. 2024. Relaxing the accurate imputation assumption in doubly robust learning for debiased collaborative filtering. In *ICML*, 29448–29460.
[29] Haoxuan Li, Chunyuan Zheng, and Peng Wu. 2023. StableDR: Stabilized doubly robust learning for recommendation on data missing not at random. In *ICLR*.
[30] Jianxun Lian, Xiaohuan Zhou, Fuzheng Zhang, Zhongxia Chen, Xing Xie, and Guangzhong Sun. 2018. xDeepFM: Combining explicit and implicit feature interactions for recommender systems. In *KDD*, 1754–1763.
[31] Siyi Lin, Sheng Zhou, Jiawei Chen, Yan Feng, Qihao Shi, Chun Chen, Ying Li, and Can Wang. 2024. ReCRec: Reasoning the causes of implicit feedback for debiased recommendation. *ACM Transactions on Information Systems* 42, 6 (2024), 1–26.
[32] Dugang Liu, Pengxiang Cheng, Zhenhua Dong, Xiuqiang He, Weike Pan, and Zhong Ming. 2020. A general knowledge distillation framework for counterfactual recommendation via uniform data. In *SIGIR*, 831–840.
[33] Dugang Liu, Pengxiang Cheng, Hong Zhu, Zhenhua Dong, Xiuqiang He, Weike Pan, and Zhong Ming. 2021. Mitigating confounding bias in recommendation via information bottleneck. In *RecSys*, 351–360.
[34] Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed H. Chi. 2018. Modeling task relationships in multi-task learning with multi-gate mixture-of-experts. In *KDD*, 1930–1939.
[35] 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 *SIGIR*, 1137–1140.
[36] Benjamin M. Marlin, Richard S. Zemel, Sam T. Roweis, and Malcolm Slaney. 2012. Collaborative filtering and the missing at random assumption. *CoRR* abs/1206.5267 (2012).
[37] Mehryar Mohri. 2018. *Foundations of Machine Learning*. MIT press.
[38] Hang Pan, Jiawei Chen, Fuli Feng, Wentao Shi, Junkang Wu, and Xiangnan He. 2023. Discriminative-invariant representation learning for unbiased recommendation. In *IJCAI*, 2270–2278.
[39] Bochen Pang, Chaozhuo Li, Yuming Liu, Jianxun Lian, Jianan Zhao, Hao Sun, Weiwei Deng, Xing Xie, and Qi Zhang. 2022. Improving relevance modeling via heterogeneous behavior graph learning in Bing Ads. In *KDD*, 3713–3721.
[40] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. PyTorch: An imperative style, high-performance deep learning library. In *NeurIPS*, 8024–8035.

[41] Yi Ren, Hongyan Tang, Jiangpeng Rong, and Siwen Zhu. 2023. Unbiased pairwise learning from implicit feedback for recommender systems without biased variance control. In *SIGIR*, 2461–2465.

[42] Steffen Rendle. 2010. Factorization machines. In *ICDM*, 995–1000.

[43] Yuta Saito. 2020. Asymmetric tri-training for debiasing Missing-Not-At-Random explicit feedback. In *SIGIR*, 309–318.

[44] Yuta Saito. 2020. Doubly robust estimator for ranking metrics with post-click conversions. In *RecSys*, 92–100.

[45] Yuta Saito and Masahiro Nomura. 2022. Towards resolving propensity contradiction in offline recommender learning. In *IJCAI*, 2211–2217.

[46] Yuta Saito, Suguru Yaginuma, Yuta Nishino, Hayato Sakata, and Kazuhide Nakata. 2020. Unbiased recommender learning from Missing-Not-At-Random implicit feedback. In *WSDM*, 501–509.

[47] Tobias Schnabel, Adith Swaminathan, Ashudeep Singh, Navin Chandak, and Thorsten Joachims. 2016. Recommendations as treatments: Debiasing learning and evaluation. In *ICML*, 1670–1679.

[48] Shao Jun. 1999. *Mathematical Statistics*. Springer.

[49] Zihua Si, Lin Guan, Zhongxiang Sun, Xiaoxue Zang, Jing Lu, Yiqun Hui, Xingchao Cao, Zeyu Yang, Yichen Zheng, Dewei Leng, et al. 2024. TWIN V2: Scaling ultra-long user behavior sequence modeling for enhanced CTR prediction at Kuaishou. In *CIKM*, 4890–4897.

[50] Zijie Song, Jiawei Chen, Sheng Zhou, Qihao Shi, Yan Feng, Chun Chen, and Can Wang. 2023. CDR: Conservative doubly robust learning for debiased recommendation. In *CIKM*, 2321–2330.

[51] Harald Steck. 2010. Training and testing of recommender systems on data missing not at random. In *KDD*, 713–722.

[52] Hongzu Su, Lichao Meng, Lei Zhu, Ke Lu, and Jingjing Li. 2024. DDPO: Direct dual propensity optimization for post-click conversion rate estimation. In *SIGIR*, 1179–1188.

[53] Adith Swaminathan and Thorsten Joachims. 2015. The self-normalized estimator for counterfactual learning. In *NeurIPS*, 3231–3239.

[54] Hongyan Tang, Junning Liu, Ming Zhao, and Xudong Gong. 2020. Progressive layered extraction (PLE): A novel multi-task learning (MTL) model for personalized recommendations. In *RecSys*, 269–278.

[55] Naftali Tishby and Noga Zaslavsky. 2015. Deep learning and the information bottleneck principle. In *ITW*, 1–5.

[56] Hao Wang, Tai-Wei Chang, Tianqiao Liu, Jianmin Huang, Zhichao Chen, Chao Yu, Ruopeng Li, and Wei Chu. 2022. ESCM2: Entire space counterfactual multi-task model for post-click conversion rate estimation. In *SIGIR*, 363–372.

[57] Xiaojie Wang, Rui Zhang, Yu Sun, and Jianzhong Qi. 2019. Doubly robust joint learning for recommendation on data missing not at random. In *ICML*, 6638–6647.

[58] Xiaojie Wang, Rui Zhang, Yu Sun, and Jianzhong Qi. 2021. Combating selection biases in recommender systems with a few unbiased ratings. In *WSDM*, 427–435.

[59] Yaozheng Wang, Dawei Feng, Dongsheng Li, Xinyuan Chen, Yunxiang Zhao, and Xin Niu. 2016. A mobile recommendation system based on logistic regression and gradient boosting decision trees. In *IJCNN*, 1896–1902.

[60] Zifeng Wang, Xi Chen, Rui Wen, Shao-Lun Huang, Ercan E. Kuruoglu, and Yefeng Zheng. 2020. Information theoretic counterfactual learning from Missing-Not-At-Random feedback. In *NeurIPS*, 1854–1864.

[61] Zimu Wang, Yue He, Jiashuo Liu, Wenchao Zou, Philip S. Yu, and Peng Cui. 2022. Invariant preference learning for general debiasing in recommendation. In *KDD*, 1969–1978.

[62] Zixuan Xu, Penghui Wei, Weimin Zhang, Shaoguo Liu, Liang Wang, and Bo Zheng. 2022. UKD: Debiasing conversion rate estimation via uncertainty-regularized knowledge distillation. In *WWW*, 2078–2087.

[63] Mengyue Yang, Guohao Cai, Furui Liu, Jiarui Jin, Zhenhua Dong, Xiuqiang He, Jianye Hao, Weiqi Shao, Jun Wang, and Xu Chen. 2023. Debiased recommendation with user feature balancing. *ACM Transactions on Information Systems* 41, 4 (2023), 1–25.

[64] Junda Ye, Zhongbao Zhang, Li Sun, Yang Yan, Feiyang Wang, and Fuxin Ren. 2023. SINCERE: Sequential interaction networks representation learning on co-evolving RiEmannian manifolds. In *WWW*, 360–371.

[65] An Zhang, Wenchang Ma, Jingnan Zheng, Xiang Wang, and Tat-Seng Chua. 2024. Robust collaborative filtering to popularity distribution shift. *ACM Transactions on Information Systems* 42, 3 (2024), 1–25.

[66] Honglei Zhang, Shuyi Wang, Haoxuan Li, Chunyuan Zheng, Xu Chen, Li Liu, Shanshan Luo, and Peng Wu. 2024. Uncovering the propensity identification problem in debiased recommendations. In *ICDE*, 653–666.

[67] Wenhao Zhang, Wentian Bao, Xiao Yang Liu, Keping Yang, Quan Lin, Hong Wen, and Ramin Ramezani. 2020. Large-scale causal approaches to debiasing post-click conversion rate estimation with multi-task learning. In *WWW*, 2775–2781.

[68] Xinyue Zhang, Cong Huang, Kun Zheng, Hongzu Su, Tianxu Ji, Wei Wang, Hongkai Qi, and Jingjing Li. 2024. Adversarial-enhanced causal multi-task framework for debiasing post-click conversion rate estimation. In *WWW*, 3287–3296.

[69] Yuqing Zhou, Tianshu Feng, Mingrui Liu, and Ziwei Zhu. 2023. A generalized propensity learning framework for unbiased post-click conversion rate estimation. In *CIKM*, 3554–3563.
[70] Feng Zhu, Mingjie Zhong, Xinxing Yang, Longfei Li, Lu Yu, Tiehua Zhang, Jun Zhou, Chaochao Chen, Fei Wu, Guanfeng Liu, et al. 2023. DCMT: A direct entire-space causal multi-task framework for post-click conversion estimation. In *ICDE*, 3113–3125.

Received 24 April 2025; revised 23 August 2025; accepted 5 November 2025