Stratified Estimator for Recommendation Systems with Implicit Feedback

ANONYMOUS AUTHOR(S)

The evaluation of recommendation systems is an important and active research area. Evaluation approaches are typically categorized as offline or online. Offline evaluation is popular due to its lower cost and accessibility, but it lacks the statistical guarantees of an online A/B test. To address this issue, a novel estimator, called the Stratified estimator, was recently proposed. This estimator uses an item set stratification based on item popularity to improve offline evaluation accuracy. While the underlying idea seems promising, the problem is that this estimator makes active use of explicit feedback (i.e., ratings given by users). However, it is widely known that explicit feedback data is hard to obtain and that the vast majority of offline evaluation research nowadays currently focuses on implicit feedback (e.g., a user clicks on an item). For this reason, the goal of this paper is to generalize the Stratified estimator to the wider case of implicit feedback. In order to do so, in this paper, we propose a novel theoretical formulation of the Stratified estimator, which is suitable for the implicit feedback scenario. We thoroughly analyze the theoretical aspects of our Generalized Stratified estimator, showing how (i) it is strictly connected to a widely known estimator for unbiased evaluation called Inverse Propensity Scoring (our estimator generalizes IPS); (ii) the new generalized stratified estimator induces a bias-variance trade-off on the estimation that can be provably preferable to the one of IPS. Finally, we conduct an experimental evaluation on implicit feedback datasets, showing that the proposed estimator is suitable in this context and it is competitive with traditional baselines.

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

Additional Key Words and Phrases: Recommendation Systems, Offline Evaluation, Unbiased Evaluation, Propensity, Importance Sampling

ACM Reference Format:
Anonymous Author(s). 2018. Stratified Estimator for Recommendation Systems with Implicit Feedback. In . ACM, New York, NY, USA, 18 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Recommender systems have become ubiquitous in our daily lives, playing a significant role in shaping the information we consume. In the research field, evaluating these systems is of utmost importance, as the evaluation phase of a recommendation algorithm can greatly impact the final recommendations a user receives [5, 13]. It is crucial to have a thorough evaluation protocol that aligns with the goals of the entity deploying the recommender system. An unreliable evaluation process can result in selecting a suboptimal algorithm, which can be a costly mistake for a platform.

In order to evaluate a recommendation system, one can adopt either an *online* or an *offline* approach. Online evaluation refers to an evaluation procedure where the recommendation algorithm we want to evaluate is directly deployed on the online platform and can interact with users. A typical example is A/B testing [9], which consists of deploying a new recommendation policy on a real system and randomly assigning it to a portion of users while the remaining users continue to use the

Conference acronym 'XX, June 03–05, 2018, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/18/06...$15.00
https://doi.org/XXXXXXX.XXXXXXX

old policy. The random assignment provides statistical validity and allows for a comparison of the metrics between the old and new policies, leading to a conclusion of which policy is better. Despite its theoretical advantages, online evaluation has several drawbacks [8, 34]. For instance, if the tested algorithm is ineffective, this may result in a potential loss of revenue and user abandonment [32].

On the other hand, offline evaluation consists of two steps: first, collect the data from a deployed recommender; then, use the offline dataset just collected to evaluate the performance of other recommendation algorithms. In this way, the tested algorithms will not interact with real users, so the risk of offline evaluation is significantly lower than the one of online evaluation. This is why offline evaluation of recommender systems is widely employed by both researchers and practitioners, and it is a very active research area [7]. Unfortunately, evaluation with offline datasets does not carry all the desirable statistical guarantees of an online A/B test. For example, if we think about the data collection phase, it is reasonable to assume that the deployed recommender may have influenced users, i.e., user interactions will be guided by what is recommended to them, thus introducing some bias in the data [33].

This is the reason why there is a long line of research on how to estimate the performance of recommendation algorithms on offline datasets while maintaining the statistical properties of online evaluation. A simple and naive way to estimate the performance of a recommender is the following: evaluate simply by taking the average performance over all observed user feedback. This is what we call the *Naive* estimator. The Naive estimator can be highly biased due to the nature of the offline data collection. To address this issue, Jadidinejad et al. [12] recently proposed a new estimator method called the *Stratified* estimator. This estimator is based on partitioning the item set and weighting each partition by the estimated probability that a user will interact with an item inside that partition. The computation of this estimated probability is a crucial design choice for the proposed estimator. For this computation, Jadidinejad et al. [12] make use of explicit feedback (e.g., explicit ratings that users give to items). However, it is widely known that explicit feedback is hard to get and that the vast majority of offline evaluation nowadays is done on implicit feedback (e.g., a user clicks on an item) [10, 24, 27, 41]. Therefore, the goal of this paper is to generalize the Stratified estimator to the wider case of implicit feedback. Generalizing the Stratified estimator beyond the considered experimental setting with explicit feedback is not straightforward, because if we simply apply the Stratified estimator as originally described, in the context of implicit feedback dataset, we prove that the Stratified estimator reduces to the biased Naive one. For this reason, we propose a novel theoretical formulation of the Stratified estimator which relies only on implicit feedback, by starting from the original one of Jadidinejad et al. [12]. This novel formulation enables a two-fold generalization: on the one hand, the *Generalized Stratified* (GS) estimator can also be used in the presence of only implicit feedback, which is actually the most relevant experimental setting in modern recommendation systems; on the other hand, we theoretically show how the novel formulation of the Stratified estimator is strictly connected with the Inverse Propensity Scoring (IPS) technique [26], which is an estimator widely used in the field. In particular, we prove that our GS estimator is a generalization of IPS, i.e., with a specific choice of the partition of the item, we can reduce the GS estimator to be equivalent to IPS. Our contributions can be summarized as follows:

- We provide a theoretical analysis of the original Stratified estimator [12] and we show that, with implicit feedback, such estimator reduces to the Naive one;
- We propose a novel estimator inspired by the original Stratified estimator, which we call Generalized Stratified (GS) estimator. We show that this novel estimator generalizes the

original Stratified estimator to the more relevant experimental setting of implicit feedback data;

- We theoretically show that the GS estimator represents a generalization also of the Inverse Propensity Scoring (IPS) estimator, giving a mathematical proof of the fact that, under a particular partitioning of the item set, GS is equivalent to IPS;
- We derive high-probability bounds on the bias-variance trade-off of GS, which provide a theoretical justification of the proposed estimator: at the cost of some bias, GS has a provably lower variance term compared to the one of IPS;
- We conduct an experimental evaluation on implicit feedback datasets, showing that the proposed estimator is suitable in this context and it is competitive with traditional baselines.

2 UNBIASED EVALUATION WITH IMPLICIT FEEDBACK

In this section, we will introduce the necessary background on unbiased evaluation of recommender systems with implicit feedback. The mathematical framework we consider is mainly based on the widely known one proposed by Yang et al. [41].

Let us define the set of all users as $\mathcal{U}$ and the set of all items as $\mathcal{I}$. For each user $u \in \mathcal{U}$, there is a set of items that are relevant for the user, which we call $\mathcal{S}_u \subseteq \mathcal{I}$. Let us call $n_u = |\mathcal{S}_u|$. A recommendation algorithm will provide a ranking of items $\hat{Z}_u$ for any user u: $\hat{Z}_{u,i}$ is the rank given by the recommendation system $\hat{Z}$ to item i for user u. Ideally, we would like to compute the following reward for a given user u:

$$R_u(\hat{Z}) = \frac{1}{n_u} \sum_{i \in \mathcal{S}_u} \lambda(\hat{Z}_{u,i}) \ ,$$

where λ is a generic scoring function for the predicted ranking. By choosing the appropriate λ, we can retrieve many state-of-the-art accuracy functions. For instance, the Recall@K metric can be retrieved by setting $\lambda(\hat{Z}_{u,i}) = \mathbf{1}(\hat{Z}_{u,i} \leq K)$. Other possible choices are showed in [41]. In the rest of the paper, we will assume without loss of generality that $\lambda(\hat{Z}_{u,i}) \in [0, 1]$, as it holds for virtually any accuracy metric. In the end, the final reward of $\hat{Z}$ is simply the average of R_u across all users:

$$R(\hat{Z}) = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} R_u(\hat{Z}) \ .$$

Unfortunately, it is impossible to compute exactly $R(\hat{Z})$ from implicit feedback data because we do not know the set of relevant items $\mathcal{S}_u$ for each u. Instead, we observe a smaller set $\mathcal{S}_u^* \subseteq \mathcal{S}_u$, which is the set of items for which we observed a click from user u. In order to observe a click on an item, it is necessary that the item is shown to the user. We denote this event with the binary random variable $O_{u,i}$: $O_{u,i} = 1$ if the item i was shown to user u during the logging phase, $O_{u,i} = 0$ otherwise. We denote the expected value of $O_{u,i}$ as $P_{u,i}$ ($O_{u,i} \sim \text{Bern}(P_{u,i})$), and that for each user-item pair for which we observe a click ($u, i : i \in \mathcal{S}_u^*$), the item must have been shown to the user ($O_{u,i} = 1$) and the item is relevant to the user ($i \in \mathcal{S}_u$). We call $\left|\mathcal{S}_u^*\right| = n_u^*$.

2.1 Naive Estimator

A very simple estimator for R_u is the *Naive* estimator, which consists of averaging the values for which we observe a feedback:

$$\hat{R}_u^{\text{naive}}(\hat{Z}) = \frac{1}{n_u^*} \sum_{i \in \mathcal{S}_u} \lambda(\hat{Z}_{u,i}) O_{u,i} = \frac{1}{n_u^*} \sum_{i \in \mathcal{S}_u^*} \lambda(\hat{Z}_{u,i}) \ . \tag{1}$$

Despite its simplicity, several previous studies show how this estimator exhibits bias in general when we have an implicit feedback dataset (i.e., $\mathbb{E}_O[\hat{R}_u^{\text{naive}}(\hat{Z})] \neq R_u(\hat{Z})$) [12, 30, 36, 41]. This means that the Naive estimator should be avoided in the implicit feedback setting since it does not converge to the true value of $R_u(\hat{Z})$.

2.2 Inverse Propensity Score Estimator

In order to reduce the bias in the estimation, one of the most used estimators is *Inverse Propensity Score* (IPS). This estimator is widely used for recommender evaluation [14, 23, 33, 41], but also in other research fields such as *causal inference* [11, 25] and *reinforcement learning* [19, 26, 37]. The IPS estimator tries to adjust for the fact that different items may have different probabilities of being observed by the user. These probabilities are also called *propensities*. The IPS estimator weights each collected feedback by the inverse of the corresponding propensity. It is defined as follows:

$$\hat{R}_u^{\mathrm{IPS}}(\hat{Z}) = \frac{1}{n_u} \sum_{i \in \mathcal{S}_u^*} \frac{\lambda(\hat{Z}_{u,i})}{P_{u,i}} \ . \tag{2}$$

The advantage of such an estimator is that it is unbiased (under mild assumptions):

Proposition 2.1. *Given a user u, assuming that $P_{u,i} > 0$ for all $i \in \mathcal{S}_u$, we have that*

$$\mathop{\mathbb{E}}_{O}[\hat{R}_u^{IPS}(\hat{Z})] = R_u(\hat{Z}) \ .$$

Proof. Given in [41]. □

Notice that, with the current formulation, not even $\hat{R}_u^{\mathrm{IPS}}$ can be computed from implicit feedback since we do not know the number of relevant items n_u. However, we will show a simple solution to solve this issue in Section 3.1. In particular, we show that it is easy to get an estimation for the number n_u of relevant items for user u using only the observed feedback and apply a re-weighting using the propensities. Therefore, for the moment we assume that we can access n_u.

3 STRATIFIED ESTIMATOR

Recently, Jadidinejad et al. [12] proposed an alternative and novel estimator for reducing the bias in recommender evaluation. This estimator is based on a *stratification* of the item set. The first step for the computation of this estimator is the definition of a *partition* $\mathcal{V} = \{v_1, \ldots, v_K\}$ of the item set $\mathcal{I}$. We call *stratum* each $v \in \mathcal{V}$, and we define $P(v)$ the probability that an item inside v is observed by a user. The computation of this probability is a crucial design choice for the proposed estimator. Since for each item i there exists exactly one stratum v such that $i \in v$, with a slight abuse of notation, we will indicate such stratum as $v(i)$. For each user u, we call $\mathcal{S}_{u,v}$ the set of relevant items for u inside stratum v (i.e., $\mathcal{S}_{u,v} = \mathcal{S}_u \cap v$), and we call $\mathcal{S}_{u,v}^*$ the set of relevant items for u inside stratum v for which we observed feedback (i.e., $\mathcal{S}_{u,v}^* = \mathcal{S}_u^* \cap v$). Accordingly, we call $|\mathcal{S}_{u,v}| = n_{u,v}$ and $|\mathcal{S}_{u,v}^*| = n_{u,v}^*$. Finally, the *Stratified* estimator is defined in [12] as:

$$\hat{R}_u^{\mathrm{Str}}(\hat{Z}) = \sum_{v \in \mathcal{V}} \hat{R}_u^{\mathrm{Str}}(\hat{Z}|v) P(v) \ , \tag{3}$$

where $\hat{R}_u^{\mathrm{Str}}(\cdot|v)$ is defined as:

$$\hat{R}_u^{\mathrm{Str}}(\hat{Z}|v) = \frac{1}{n_{u,v}^*} \sum_{i \in \mathcal{S}_{u,v}^*} \lambda(\hat{Z}_{u,i}) \ .$$

In their work, Jadidinejad et al. [12] propose a stratification based on the popularity of items and compute the probability $P(v)$ using *explicit* feedback. $P(v)$ should represent the likelihood that a rating for an item in stratum v will be observed. To compute $P(v)$, they take the ratio of the number of explicit ratings for items within stratum v to the total number of explicit ratings in the dataset.

This heuristic choice seems reasonable because explicit ratings include both positive and negative interactions, providing a proxy for the popularity of a given stratum. However, this approach is not suitable for the implicit feedback scenario, as we will show in the following.

Since we are dealing with implicit feedback, for each user u, we have only access to the implicit clicks (i.e., positive feedback) provided for relevant items, which compose $\mathcal{S}_u^*$. Hence, if we want to follow the original proposal, we compute $P(v)$ as

$$P(v) = \frac{n_{u,v}^*}{n_u^*} \; .$$

Unfortunately, using this definition with implicit feedback, the Stratified estimator corresponds to the biased Naive one, as we show in the following Proposition.

Proposition 3.1. *Given $\hat{R}_u^{Str}$ defined as in Eq.* (3), *with a stratification $\mathcal{V}$ and $P(v) = \frac{n_{u,v}^*}{n_u^*}$, we have that $\hat{R}_u^{Str}(\hat{Z}) = \hat{R}_u^{naive}(\hat{Z})$.*

Proof. First, we re-write the expression of $\hat{R}_u^{\text{Str}}(\hat{Z})$ as follows:

$$\begin{aligned}
\hat{R}_u^{\text{Str}}(\hat{Z}) &= \sum_{v\in\mathcal{V}} \hat{R}_u^{\text{Str}}(\hat{Z}|v)P(v) \\
&= \sum_{v\in\mathcal{V}} P(v)\frac{1}{n_{u,v}^*}\sum_{i\in\mathcal{S}_{u,v}^*}\lambda(\hat{Z}_{u,i}) \quad \text{(from the definition of } \hat{R}_u^{\text{Str}}(\hat{Z}|v)) \\
&= \sum_{v\in\mathcal{V}} \frac{n_{u,v}^*}{n_u^*}\frac{1}{n_{u,v}^*}\sum_{i\in\mathcal{S}_{u,v}^*}\lambda(\hat{Z}_{u,i}) \quad \text{(from the definition of } P(v)) \\
&= \frac{1}{n_u^*}\sum_{v\in\mathcal{V}}\sum_{i\in\mathcal{S}_{u,v}^*}\lambda(\hat{Z}_{u,i}) \; .
\end{aligned}$$

Now, we notice that $\sum_{v\in\mathcal{V}}\sum_{i\in\mathcal{S}_{u,v}^*}\lambda(\hat{Z}_{u,i}) = \sum_{i\in\mathcal{S}_u^*}\lambda(\hat{Z}_{u,i})$. This is because the left term is a sum of all the $\lambda(\hat{Z}_{u,i})$ inside a given partition, for each partition, but since each item in $\mathcal{S}_u^*$ appears exactly once in each partition, it is equivalent to summing all the $\lambda(\hat{Z}_{u,i})$ for $i \in \mathcal{S}_u^*$. Therefore:

$$\begin{aligned}
\hat{R}_u^{\text{Str}}(\hat{Z}) &= \frac{1}{n_u^*}\sum_{v\in\mathcal{V}}\sum_{i\in\mathcal{S}_{u,v}^*}\lambda(\hat{Z}_{u,i}) \\
&= \frac{1}{n_u^*}\sum_{i\in\mathcal{S}_u^*}\lambda(\hat{Z}_{u,i}) \\
&= \hat{R}_u^{\text{naive}}(\hat{Z}) \; .
\end{aligned}$$

□

3.1 Stratified Estimator with Implicit Feedback

In this Section, we propose an alternative way to define $P(v)$ which is more suitable for the implicit feedback scenario.

Recalling that the ideal reward we would like to estimate is $R_u(\hat{Z}) = \frac{1}{n_u}\sum_{i\in\mathcal{S}_u}\lambda(\hat{Z}_{u,i})$, we notice that it is a uniform average across all the items relevant to the user $\mathcal{S}_u$. Instead, the Stratified estimator is defined as $\hat{R}_u^{\text{Str}}(\hat{Z}) = \sum_{v\in\mathcal{V}}\hat{R}_u^{\text{Str}}(\hat{Z}|v)P(v)$, which is a weighted average of the estimated reward for each stratum, where the weight of each stratum is $P(v)$. This means that, if the weight of a single item is $\frac{1}{n_u}$ (as in the ideal reward), it is reasonable to define the weight of a single stratum as

$$P(v) = \frac{n_{u,v}}{n_u} \ ,$$

i.e., the number of relevant items inside the stratum, over the total number of relevant items. With this definition, the Stratified estimator becomes:

$$\begin{aligned}\hat{R}_u^{\text{Str}}(\hat{Z}) &= \sum_{v \in \mathcal{V}} \hat{R}_u^{\text{Str}}(\hat{Z}|v)P(v) \\ &= \sum_{v \in \mathcal{V}} \frac{n_{u,v}}{n_u} \frac{1}{n^*_{u,v}} \sum_{i \in \mathcal{S}^*_{u,v}} \lambda(\hat{Z}_{u,i}) \\ &= \frac{1}{n_u} \sum_{v \in \mathcal{V}} \frac{n_{u,v}}{n^*_{u,v}} \sum_{i \in \mathcal{S}^*_{u,v}} \lambda(\hat{Z}_{u,i}) \ ,\end{aligned}$$

which is the sum, for each stratum v, of all the clicks found in v weighted by $\frac{n_{u,v}}{n^*_{u,v}}$, all multiplied by $\frac{1}{n_u}$. We can re-write the double summation as a single summation of all the observed clicks, where the weight of each click depends only on the stratum of the item:

$$\begin{aligned}\hat{R}_u^{\text{Str}}(\hat{Z}) &= \frac{1}{n_u} \sum_{v \in \mathcal{V}} \frac{n_{u,v}}{n^*_{u,v}} \sum_{i \in \mathcal{S}^*_{u,v}} \lambda(\hat{Z}_{u,i}) \\ &= \frac{1}{n_u} \sum_{i \in \mathcal{S}^*_u} \frac{n_{u,v(i)}}{n^*_{u,v(i)}} \lambda(\hat{Z}_{u,i}) \ .\end{aligned}$$

Now, we notice that this novel formulation of the Stratified estimator resembles the IPS one, where the weight of each click is $\frac{n_{u,v(i)}}{n^*_{u,v(i)}}$ instead of $\frac{1}{P_{u,i}}$. The problem is that we do not know the quantity $n_{u,v(i)}$, which is the number of items inside $v(i)$ that are relevant to user u. To solve this issue, we resort to the *Self-Normalization* control variate technique, as illustrated in [41]. They show how we can approximate $n_{u,v}$ by noticing that:

$$n_{u,v} \approx \sum_{i \in \mathcal{S}^*_{u,v}} \frac{1}{P_{u,i}} \ ,$$

because those two quantities are equivalent in expectation:

$$\begin{aligned}\mathop{\mathbb{E}}_O\left[\sum_{i \in \mathcal{S}^*_{u,v}} \frac{1}{P_{u,i}}\right] &= \mathop{\mathbb{E}}_O\left[\sum_{i \in \mathcal{S}_{u,v}} \frac{1}{P_{u,i}} O_{u,i}\right] \\ &= \sum_{i \in \mathcal{S}_{u,v}} \frac{1}{P_{u,i}} \mathop{\mathbb{E}}_O\left[O_{u,i}\right] \\ &= \sum_{i \in \mathcal{S}_{u,v}} \frac{P_{u,i}}{P_{u,i}} \\ &= n_{u,v} \ .\end{aligned} \tag{4}$$

Therefore, we define $\hat{w}_{u,i}$ to approximate $\frac{n_{u,v(i)}}{n^*_{u,v(i)}}$ as

$$\hat{w}_{u,i} = \frac{1}{n^*_{u,v(i)}} \sum_{j \in \mathcal{S}^*_{u,v(i)}} \frac{1}{P_{u,j}} \ , \tag{5}$$

which corresponds to the average inverse propensity on the observed feedback of a given stratum, and can be computed using implicit feedback. In the end, the final formulation of the *Generalized Stratified* (GS) estimator is the following:

$$\hat{R}_u^{\mathrm{GS}}(\hat{Z}) = \frac{1}{n_u} \sum_{i \in \mathcal{S}_u^*} \hat{w}_{u,i} \lambda(\hat{Z}_{u,i}) \ , \tag{6}$$

where the weights $\hat{w}_{u,i}$ are defined as in Eq. (5).

4 THEORETICAL ANALYSIS

In this Section, we provide a theoretical analysis of the Genedalized Stratified estimator, in order to further understand its theoretical properties. First, we show how our GS estimator formulation is strictly connected with the IPS estimator (it can be seen as a generalization of IPS); then, we analyze the bias-variance trade-off through bounds that hold with high probability.

4.1 Relationship with IPS

In Section 3.1, we have shown how our novel formulation of the Generalized Stratified estimator (Eq. (6)) resembles one of the IPS. In the following Proposition, we show that, for a particular choice of the stratification $\mathcal{V}$, the GS estimator is equivalent to the IPS one.

Proposition 4.1. *Consider the formulation of $\hat{R}_u^{GS}$ given in Eq.* (6)*. If we select a stratification $\mathcal{V}$ such that $n_{u,v}^* = 1$ for each $v \in \mathcal{V}$, we have that $\hat{R}_u^{GS}(\hat{Z}) = \hat{R}_u^{IPS}(\hat{Z})$.*

Proof.

$$\begin{aligned}
\hat{R}_u^{\mathrm{GS}}(\hat{Z}) &= \frac{1}{n_u} \sum_{i \in \mathcal{S}_u^*} \lambda(\hat{Z}_{u,i}) \hat{w}_{u,v(i)} \\
&= \frac{1}{n_u} \sum_{i \in \mathcal{S}_u^*} \lambda(\hat{Z}_{u,i}) \frac{1}{n_{u,v(i)}^*} \sum_{j \in \mathcal{S}_{u,v(i)}^*} \frac{1}{P_{u,j}} \\
&= \frac{1}{n_u} \sum_{i \in \mathcal{S}_u^*} \frac{\lambda(\hat{Z}_{u,i})}{P_{u,i}} = \hat{R}_u^{\mathrm{IPS}}(\hat{Z})
\end{aligned}$$

□

This means that our Generalized Stratified estimator is actually a generalization of IPS: for a particular choice of $\mathcal{V}$, we can resort to IPS, but we are not forced to select that particular stratification. Therefore, we may have better estimation quality with a different selection of the stratification of the items. We will show in the following how this selection influences the Bias-Variance trade-off of the estimation process.

4.2 Bias-Variance Trade-Off

In this Section, we investigate the bias-variance trade-off of the GS estimator, and we compare it with the IPS baseline. To do so, we analyze a generic estimator defined as:

$$\hat{R}_u(\hat{Z}) = \frac{1}{n_u} \sum_{i \in \mathcal{S}_u^*} w_{u,i} \lambda(\hat{Z}_{u,i}) \ , \tag{7}$$

where $w_{u,i}$ is a generic non-negative weight. We use this generic formulation because it is easy to recover the known estimators from here: if we set $w_{u,i} = 1/P_{u,i}$, we have $\hat{R}_u(\hat{Z}) = \hat{R}_u^{\mathrm{IPS}}(\hat{Z})$; while if we set $w_{u,i} = \hat{w}_{u,i}$ defined as in Eq. (5), we have $\hat{R}_u(\hat{Z}) = \hat{R}_u^{\mathrm{GS}}(\hat{Z})$. To analyze the bias-variance trade-off, we derive bounds that hold with high probability on the absolute distance between the estimator and the real value of the reward that we would like to estimate: $\left|\hat{R}_u(\hat{Z}) - R_u(\hat{Z})\right|$. These kinds of bounds are usually called *concentration inequalities* [2]. Concentration inequalities are powerful tools with desirable properties, as they offer probabilistic bounds that hold with high probability (allowing to select the desired confidence level). Also, unlike bounds that hold only in expectation, concentration inequalities hold even for finite sample sizes.

From the triangle inequality, it is easy to see how the absolute distance between the estimator and the real value of the reward $\left|\hat{R}_u(\hat{Z}) - R_u(\hat{Z})\right|$ can be decomposed into a bias term and a concentration term (which is closely related to the variance):

$$\begin{aligned} \left|R_u(\hat{Z}) - \hat{R}_u(\hat{Z})\right| &= \left|R_u(\hat{Z}) - \mathbb{E}[\hat{R}_u(\hat{Z})] + \mathbb{E}[\hat{R}_u(\hat{Z})] - \hat{R}_u(\hat{Z})\right| \\ &\leq \underbrace{\left|R_u(\hat{Z}) - \mathbb{E}[\hat{R}_u(\hat{Z})]\right|}_{\text{bias}} + \underbrace{\left|\mathbb{E}[\hat{R}_u(\hat{Z})] - \hat{R}_u(\hat{Z})\right|}_{\text{concentration}} . \end{aligned}$$

In the following Proposition, we derive a bound for the bias of the generic estimator $\hat{R}_u(\hat{Z})$.

Proposition 4.2. *Let $\hat{R}_u(\hat{Z})$ be the estimator defined in Eq.* (7). *Then,*

$$\left|\mathbb{E}[\hat{R}_u(\hat{Z})] - R_u(\hat{Z})\right| \leq \frac{1}{n_u} \sum_{i \in \mathcal{S}_u} \left|w_{u,i} P_{u,i} - 1\right| .$$

Proof.

$$\begin{aligned} \left|\mathop{\mathbb{E}}_{O}[\hat{R}_u(\hat{Z})] - R_u(\hat{Z})\right| = \quad & \left|\mathop{\mathbb{E}}_{O}[\hat{R}_u(\hat{Z}) - \hat{R}_u^{\text{IPS}}(\hat{Z})]\right| \\ &= \frac{1}{n_u} \left| \sum_{i \in \mathcal{S}_u} \lambda(\hat{Z}_{u,i}) \mathop{\mathbb{E}}_{O} \left[w_{u,i} O_{u,i} - \frac{O_{u,i}}{P_{u,i}} \right] \right| \\ &= \frac{1}{n_u} \left| \sum_{i \in \mathcal{S}_u} \lambda(\hat{Z}_{u,i}) \left(w_{u,i} P_{u,i} - 1 \right) \right| \\ &\leq \frac{1}{n_u} \sum_{i \in \mathcal{S}_u} \left| \lambda(\hat{Z}_{u,i}) \left(w_{u,i} P_{u,i} - 1 \right) \right| \\ &\leq \frac{1}{n_u} \sum_{i \in \mathcal{S}_u} \left| \hat{w}_{u,i} P_{u,i} - 1 \right| , \end{aligned}$$

where the first equality derives from the fact that IPS is unbiased (Proposition 2.1), the first inequality is the triangle inequality, and the second inequality derives from the fact that $\lambda(\hat{Z}_{u,i}) \in [0, 1]$. □

Now, we are ready to introduce a high-probability bound for the generic estimator $\hat{R}_u(\hat{Z})$.

Theorem 4.1. *Let $\hat{R}_u(\hat{Z})$ be defined as in Eq.* (7)*, and let $\delta \in (0, 1/2)$. Then, the following inequality holds with probability at least $1 - 2\delta$:*

$$\begin{aligned} \left|\hat{R}_u(\hat{Z}) - R_u(\hat{Z})\right| \leq & \underbrace{\frac{1}{n_u} \sum_{i \in \mathcal{S}_u} \left|w_{u,i} P_{u,i} - 1\right|}_{\textit{bias}} \\ & + \underbrace{\sqrt{\frac{2 \log(2/\delta) \sum_{i \in \mathcal{S}_u^*} w_{u,i}^2}{n_u (n_u - 1)}} + w_u^{max} \frac{7 \log(2/\delta)}{3(n_u - 1)}}_{\textit{concentration}} , \end{aligned}$$

where $w_u^{max} := \max_j w_{u,j}$.

Proof. From Proposition 4.2, we have a bound for the bias term. Let us now focus on the concentration term. For this part of the bound, we follow the proof strategy used by Schnabel et al. [33], Proposition 3.1. Recall the definition of $\hat{R}_u(\hat{Z})$:

$$\hat{R}_u(\hat{Z}) = \frac{1}{n_u} \sum_{i \in \mathcal{S}_u^*} \lambda(\hat{Z}_{u,i}) w_{u,i} = \frac{1}{n_u} \sum_{i \in \mathcal{S}_u} \lambda(\hat{Z}_{u,i}) w_{u,i} O_{u,i}.$$

If we define the random variable $X_i := \lambda(\hat{Z}_{u,i}) w_{u,i} O_{u,i}$, we have that X_i is bounded between 0 and $w_u^{max} := \max_j w_{u,j}$ for any i. Let us define the sample mean $\bar{X} := \frac{1}{n_u} \sum_{i \in \mathcal{S}_u} X_i$, and the sample variance $\widehat{Var}(X) := \frac{1}{n_u - 1} \sum_{i \in \mathcal{S}_u} (X_i - \bar{X})^2$. Now, we can apply the *Empirical Bernstein Inequality* ([22], Theorem 4), obtaining that, with probability at least $1 - 2\delta$:

$$\left|\hat{R}_u(\hat{Z}) - \mathbb{E}[\hat{R}_u(\hat{Z})]\right| \leq \sqrt{\frac{2 \log(2/\delta) \widehat{Var}(X)}{n_u}} + w_u^{max} \frac{7 \log(2/\delta)}{3(n_u - 1)}. \tag{8}$$

This provides us with a bound on the distance between the estimator and its expected reward, i.e., on the concentration. We can also find a bound for the sample variance as follows:

$$\begin{aligned}
\widehat{Var}(X) &:= \frac{1}{n_u - 1} \sum_{i \in \mathcal{S}_u} (X_i - \bar{X})^2 \leq \frac{1}{n_u - 1} \sum_{i \in \mathcal{S}_u} X_i^2 \\
&= \frac{1}{n_u - 1} \sum_{i \in \mathcal{S}_u} (\lambda(\hat{Z}_{u,i}) w_{u,i} O_{u,i})^2 = \frac{1}{n_u - 1} \sum_{i \in \mathcal{S}_u^*} (\lambda(\hat{Z}_{u,i}) w_{u,i})^2 \\
&\leq \frac{1}{n_u - 1} \sum_{i \in \mathcal{S}_u^*} w_{u,i}^2.
\end{aligned}$$

This, together with Eq. (8), concludes the proof.

□

This bound allows us to characterize the theoretical features of the analyzed estimators rigorously. Notably, this bound unveils how changing the weights can strongly influence the bias-variance trade-off of the estimator. For instance, if we set $w_{u,i} = 1/P_{u,i}$, the bias part of the bound goes to 0. This result is expected since with $w_{u,i} = 1/P_{u,i}$ we recover the IPS estimator, and it is known that IPS is unbiased. Instead, for the GS estimator, we set $w_{u,i} = \hat{w}_{u,i}$. Therefore, looking at the bias term obtained in Theorem 4.1, we notice that there will be some bias in general (unless the stratification is such that $\hat{w}_{u,i} = 1/P_{u,i}$). However, the GS estimator *provably reduces* the concentration term of the bound compared to the IPS estimator. This finding gives a theoretical justification for the GS estimator: with the GS weights $\hat{w}_{u,i}$, we are decreasing the variance of the estimation at the cost of some bias, and this may lead to a decreased estimation error. This result is proved in the following Theorem.

Theorem 4.2. *Define the quantities*

$$bias(\hat{R}_u(\hat{Z})) = \frac{1}{n_u} \sum_{i \in \mathcal{S}_u} \left| w_{u,i} P_{u,i} - 1 \right|$$

and

$$conc(\hat{R}_u(\hat{Z})) = \sqrt{\frac{2 \log(2/\delta) \sum_{i \in \mathcal{S}_u^*} w_{u,i}^2}{n_u(n_u - 1)}} + w_u^{max} \frac{7 \log(2/\delta)}{3(n_u - 1)}$$

for a generic estimator $\hat{R}_u(\hat{Z})$ defined as in Eq. (7). Let $\hat{R}_u^{IPS}(\hat{Z})$ be defined as in Eq. (2) and $\hat{R}_u^{GS}(\hat{Z})$ be defined as in Eq. (6) for any stratification $\mathcal{V}$. Then, the following two inequalities both hold:

$$bias(\hat{R}_u^{GS}(\hat{Z})) \geq bias(\hat{R}_u^{IPS}(\hat{Z})) = 0 \,,$$
$$conc(\hat{R}_u^{GS}(\hat{Z})) \leq conc(\hat{R}_u^{IPS}(\hat{Z})) \,.$$

Proof. The bias inequality trivially holds: from Proposition 2.1, we know that $\text{bias}(\hat{R}_u^{\text{IPS}}(\hat{Z})) = 0$. Therefore, since $\text{bias}(\hat{R}_u^{\text{GS}}(\hat{Z}))$ is a sum of non-negative quantities, we have that $\text{bias}(\hat{R}_u^{\text{GS}}(\hat{Z})) \geq \text{bias}(\hat{R}_u^{\text{IPS}}(\hat{Z}))$. Regarding the concentration term, we notice that the only difference between $\hat{R}_u^{\text{IPS}}$ and $\hat{R}_u^{\text{GS}}$ is in the weights. Hence, if we show that $\sum_{i\in\mathcal{S}_u^*} \hat{w}_{u,i}^2 \leq \sum_{i\in\mathcal{S}_u^*} \frac{1}{P_{u,i}^2}$ and that $\max_i \hat{w}_{u,i} \leq \max_i \frac{1}{P_{u,i}}$, we prove the Theorem. The latter is easy to see: $\hat{w}_{u,i}$ is defined as the mean of the inverse propensities of the items inside $\mathcal{S}_{u,v(i)}^*$. Furthermore, the mean of a set of values can not be larger than the maximum value. Hence, for any i:

$$\hat{w}_{u,i} \leq \max_{j\in\mathcal{S}_{u,v(i)}^*} \frac{1}{P_{u,j}} \leq \max_j \frac{1}{P_{u,j}} \,.$$

Now, we need to prove the former. Using Jensen's inequality, we get:

$$\left(\frac{1}{n_{u,v}^*}\sum_{i\in\mathcal{S}_{u,v}^*}\frac{1}{P_{u,i}}\right)^2 \leq \frac{1}{n_{u,v}^*}\sum_{i\in\mathcal{S}_{u,v}^*}\frac{1}{P_{u,i}^2}$$

for any v. Equivalently, by multiplying each side by $n_{u,v}^*$:

$$n_{u,v}^*\left(\frac{1}{n_{u,v}^*}\sum_{i\in\mathcal{S}_{u,v}^*}\frac{1}{P_{u,i}}\right)^2 \leq \sum_{i\in\mathcal{S}_{u,v}^*}\frac{1}{P_{u,i}^2} \,.$$

Therefore, it follows that:

$$\sum_{i\in\mathcal{S}_u^*}\hat{w}_{u,i}^2 = \sum_{v\in\mathcal{V}} n_{u,v}^*\left(\frac{1}{n_{u,v}^*}\sum_{i\in\mathcal{S}_{u,v}^*}\frac{1}{P_{u,i}}\right)^2$$
$$\leq \sum_{v\in\mathcal{V}}\sum_{i\in\mathcal{S}_{u,v}^*}\frac{1}{P_{u,i}^2} = \sum_{i\in\mathcal{S}_u^*}\frac{1}{P_{u,i}^2} \,.$$

□

5 EXPERIMENTS

In this section, we show the experiments we conducted in order to answer the following research question:

Is our novel Generalized Stratified estimator an accurate estimator with implicit feedback?

To do so, we will use datasets that provide a way for a ground truth unbiased evaluation, and we will compare the outcome of our estimator (and the outcome of baseline estimators as well) to the ground truth evaluation. In what follows, we illustrate the details of our experimental setup. For our experiments, we based our code on the codebase of Jadidinejad et al. [12], using the Cornac framework[1]. To support ease of reproducibility, we share our code (anonymously) here: https://drive.google.com/drive/folders/1xKvdfu40kQN8yi4LTV_FYfPUy4n1tdn2?usp=share_link.

[1] https://github.com/PreferredAI/cornac

Algorithm 1 Experimental pipeline

```
Input: A dataset with MNAR implicit feedback $\mathcal{D}^{MNAR}$, a dataset with MCAR implicit feedback $\mathcal{D}^{MCAR}$, a random $seed$, a list of Recommendation Models, the estimator to be evaluated $\hat{R}$
Output: Kendall's $\tau$ correlation between the ranking provided by $\hat{R}$ and the one obtained on $\mathcal{D}^{MCAR}$
procedure EVALUATE($\mathcal{D}^{MNAR}$, $\mathcal{D}^{MCAR}$, $seed$, Models, $\hat{R}$)
    $\mathcal{D}^{MNAR}_{train}, \mathcal{D}^{MNAR}_{test} \leftarrow$ Split($\mathcal{D}^{MNAR}$, $seed$)    ▹ Split MNAR dataset into training and testing sets
    Train(Models, $\mathcal{D}^{MNAR}_{train}$)    ▹ Train all models on training set
    $r_{\hat{R}} \leftarrow$ Evaluate(Models, $\mathcal{D}^{MNAR}_{test}$, $\hat{R}$)    ▹ Evaluate models on testing set using estimator $\hat{R}$, obtain a ranking of the models
    $r_{truth} \leftarrow$ Evaluate(Models, $\mathcal{D}^{MCAR}$, $\hat{R}^{\text{naive}}$)    ▹ Evaluate models on MCAR dataset using sample average, obtain the ground-truth ranking
    $\tau \leftarrow$ Kendall's $\tau(r_{\hat{R}}, r_{truth})$    ▹ Compute Kendall's $\tau$ between rankings
    return $\tau$
end procedure
```

5.1 Experimental Setup

5.1.1 Datasets. In our empirical validation, we use two real-world datasets widely used in the literature on unbiased recommender evaluation [e.g., 4, 29, 38, 41]: *Yahoo! R3* [21] and *Coat* [33]. The fundamental characteristic they both have is that they consist of a subset with Missing-Not-At-Random (MNAR) logged data and a subset with Missing-Completely-At-Random (MCAR) data.

The Yahoo! R3 dataset contains user-song ratings, with a subset of approximately 300,000 MNAR ratings from 15,400 users on 1000 songs. The MCAR set is composed of ratings from a subset of 5,400 users on 10 randomly selected songs. The Coat dataset consists of ratings from 290 users, with an MNAR subset of ratings on 24 self-selected coats (the domain is outfit recommendations) from the inventory of the platform. The MCAR test set comprises ratings on 16 randomly selected coats from a total stock of 300 coats.

A dataset with such a characteristic enables the evaluation of estimators in this way: we can treat the MNAR subset as a standard dataset in the recommendation system domain and estimate the performance of a recommendation model on such a dataset with one of the estimators we want to evaluate; at the same time, we can evaluate the same recommendation model with the simple Naive estimator on the MCAR, which is proven to be an unbiased estimate of the performance of the evaluated model, thanks to the MCAR property. In this way, we can compare the outcome of the estimator on the MNAR dataset with the unbiased evaluation done on the MCAR dataset and evaluate the estimation accuracy. Since we are investigating the scenario of implicit feedback, we follow standard practice in the literature and convert the ratings in the dataset to a binary format by only keeping ratings that are 4 or higher [1, 3, 12, 20].

5.1.2 Estimators and Evaluation method. We will compare the following three estimators:

- *Generalized Stratified (GS) estimator:* The estimator proposed in Eq. (6).
- *Naive estimator:* Standard baseline widely used for evaluating the performance of recommender algorithm, also called Average-Over-All [41], defined in Eq. (1). When using implicit data this estimator is equivalent to the original Stratified estimator proposed by Jadidinejad et al. [12], as shown in Section 3.

- *Inverse Propensity Scoring (IPS) estimator:* A typical estimator that accounts for the propensity for de-biasing the evaluation, defined in Eq. (2). Notice that, in Eq. (2), IPS uses the actual number of relevant items for a given user, which is generally unknown. Nonetheless, we can estimate this value with the Self-Normalization trick shown in Eq. (4).

In order to compare them, we first need to define which are the recommendation models we want to estimate the performance of. We included various recommendation models from different families of approaches (such as non-personalized, matrix factorization, BPR, etc.). Specifically, we included:

- **MostPop**: a simple non-personalized model that recommends the most popular items to each user [6];
- **Matrix Factorization (MF)**: a de-facto standard recommendation model that maps users and items into latent vectors and makes prediction by multiplying the latent factors of a given user and item with a dot product [17]. We use also several variants of the standard MF model, such as:
  - *Probabilistic Matrix Factorization* (PMF), an extension of MF that should be able to better handle large and sparse data [31];
  - *Weighted Matrix Factorization* (WMF), that takes into account also the uncertainty of the predictions [10];
  - *Non-negative Matrix Factorization* (NMF), which is a MF technique with the additional constraint of having only non-negative latent factors [18];
  - *Maximum Margin Matrix Factorization* (MMMF), which maximizes the Hinge ranking loss [39];
- **SVD**: a Singular Value Decomposition of the user-item interaction matrix, as proposed in many papers such as [5, 16];
- **Bayesian Personalized Ranking (BPR)**: a model that optimizes a ranking accuracy via gradient ascent by drawing a triple: user, positive item and negative item [28].

For the ones using latent factors, we varied the latent factor size in $\{10, 20, \ldots, 100\}$. This procedure led to a total of 81 recommenders being evaluated with the three estimators mentioned before. We set $\lambda(\hat{Z}_{u,i}) = \mathbf{1}(\hat{Z}_{u,i} \leq K)$, i.e., the estimators are trying to estimate the value of the Recall@K metric. We use different cutoffs: $K \in \{5, 10, 20, 30, 100\}$.

In order to compare the accuracy of the estimators, our main goal is to evaluate how well they are able identify the most effective recommenders. To this end, we first rank the recommendation models by their estimated performance and then use Kendall's Tau [15] to compare this ranking with the one obtained from the unbiased evaluation ground truth. We can obtain a ground truth ranking using the Naive estimator on the MAR test set, since a sample average is an unbiased estimator under the MAR assumption. Intuitively, Kendall's Tau calculates the quota of couples of recommenders that are in the same relative order in both rankings, hence the higher Kendall's Tau the better. We repeat the experimental procedure with 20 random seeds for statistical significance. The experimental pipeline is summarized in Algorithm 1. For the computation of the experiments, we employed an instance with 16 cores and 32 GB of RAM with Ubuntu 20.04. The total computation time for all the seeds and all estimators was about 160 hours.

5.1.3 Estimating Propensities. The datasets we have chosen have no information on the propensities. Therefore, we need to estimate them from the data. Recall that the propensity score $P_{u,i}$ is a measure of how likely it is that user u is going to be exposed to item i. By following standard practice from prior work [e.g., 12, 29, 35, 41], we simplify the propensity estimation process by assuming that, for any user-item pair (u, i), the propensity $P_{u,i}$ is proportional to the power of the observed popularity

	Recall@5	Recall@10	Recall@20	Recall@30	Recall@100
GS	0.7148	**0.7763**	**0.8008**	**0.7521**	**0.7261**
IPS	0.6968	0.7518	0.7314	**0.7750**	0.6823
Naive	0.7168	**0.7756**	**0.7880**	0.7234	**0.7275**

Table 1. Kendall's τ rank correlation coefficient between the evaluated estimators and the ground truth ranking on the Yahoo dataset. When there are statistically significant differences (Welch's t-test [40], p-value of 0.05 with Bonferroni correction), we highlight in bold the best estimator and the estimators that are not significantly different from the best. When using implicit data the Naive estimator is equivalent to the original Stratified estimator proposed by Jadidinejad et al. [12], as shown in Section 3.

	Recall@5	Recall@10	Recall@20	Recall@30	Recall@100
GS	0.4187	0.5403	0.5553	0.5611	0.6600
IPS	0.4219	0.5367	0.5564	0.5634	0.6497
Naive	0.4065	0.5439	0.5544	0.5587	0.6703

Table 2. Kendall's τ rank correlation coefficient between the evaluated estimators and the ground truth ranking on the Coat dataset. When there are statistically significant differences (Welch's t-test [40], p-value of 0.05 with Bonferroni correction), we highlight in bold the best estimator and the estimators that are not significantly different from the best. When using implicit data the Naive estimator is equivalent to the original Stratified estimator proposed by Jadidinejad et al. [12], as shown in Section 3.

of item i:

$$P_{u,i} \propto (n_u^*)^{\frac{\gamma+1}{2}}$$

In particular, we follow the indication of Yang et al. [41] and set $\gamma = 2$[2].

5.2 Results

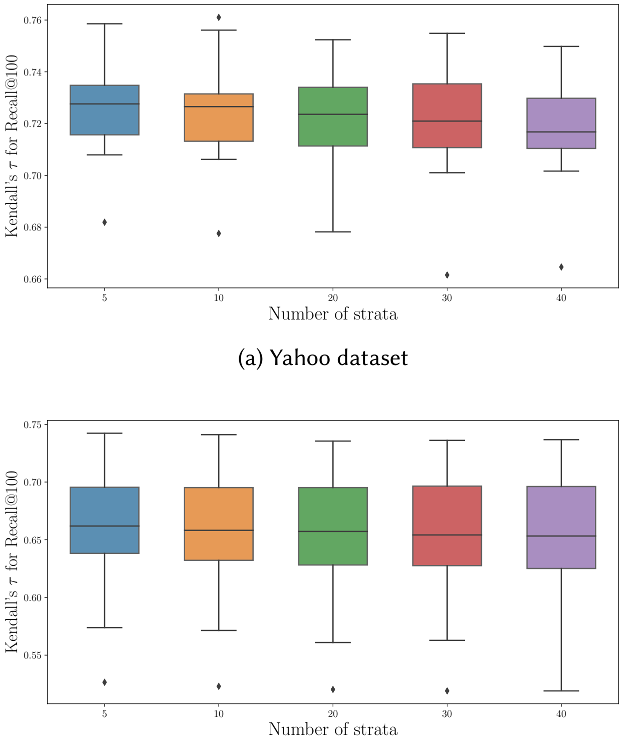
The experimental results are reported in Table 1 (Yahoo! R3) and Table 2 (Coat). We measure statistical significance with a Welch's t-test [40] at a p-value of 0.05 with Bonferroni correction to account for the multiple comparisons. For the Generalized Stratified estimator, we use 5 strata.

When generalizing a method, it is common to encounter the problem of adding complexity and parameters that can decrease the method's practical applicability. In this regard, the proposed Generalized Stratified (GS) estimator appears to be a successful solution, as demonstrated by experimental results. Furthermore, results indicate that it combines the strengths of the Naive estimator and the Inverse Probability of Sampling (IPS) estimator.

Experimental results on the Coat dataset showed that the GS estimator was statistically indistinguishable from both the Naive estimator and the IPS estimator, despite having a more complex design. However, it is essential to note that the small size of the Coat dataset introduces some variance into the results.

On the Yahoo! R3 dataset, the experimental results showed that the GS estimator was either the best estimator or statistically equal to the best estimator, highlighting its potential for practical applications. Notably, in some cases, the GS estimator was statistically equal to the Naive estimator, which was found to be the best estimator for the task, while for Recall@30, it was statistically equal to IPS, which was superior to Naive. These results suggest that the proposed GS estimator is

[2] Yang et al. [41] tried values of $\gamma \in \{1.5, 2, 2.5, 3\}$, showing that they get similar outcomes. Hence, we choose $\gamma = 2$.



(a) Yahoo dataset

(b) Coat dataset

Fig. 1. Box plot for Kendall's τ rank correlation coefficient between the GS estimator and the ground truth ranking for the metric Recall@100, while varying the number of strata used in GS.

adaptable and can perform equally well or better than both Naive and IPS estimators, depending on the specific task and dataset.

In summary, the experimental evidence indicates that the proposed GS estimator effectively combines the strengths of Naive and IPS estimators while avoiding their limitations, making it a promising solution for practical applications.

To assess the sensitivity of the estimator to the number of strata, Figure 1 shows how the Kendall's Tau varies. We can see how the proposed estimator is robust to the number of strata and therefore does not require extensive tuning. Moreover, as was theoretically demonstrated in Section 4, with a low number of strata the Generalized Stratified estimator behaves more similarly to the Naive estimator, while with a higher number of strata it behaves more similarly to IPS. Thus, the number of strata can be adjusted to achieve the desired bias-variance trade-off, depending on the specific scenario of interest. Our findings align with similar results reported in the original paper by Jadidinejad et al. [12].

6 CONCLUSION

In the field of recommendation systems, developing a reliable methodology for offline evaluation is crucial for both researchers and practitioners. This paper focuses on unbiased offline evaluation of recommendation systems, analyzing a recently proposed estimator called the Stratified estimator. Our analysis shows that the original Stratified estimator reduces to a simple average when applied to implicit feedback datasets, which is the most prominent experimental setting in the field. Therefore, we propose a novel theoretical formulation for a Generalized Stratified (GS) estimator, which is specifically designed for implicit feedback. Our theoretical analysis shows that the GS estimator is a generalization of the widely known unbiased estimator called Inverse Propensity Scoring (IPS). Moreover, we theoretically demonstrate that by changing the stratification, the generalized estimator allows for better control of the bias-variance trade-off, thus improving the estimation of IPS. Finally, we provide empirical evidence that the proposed estimator is competitive with traditional baselines in real-world implicit feedback data. Overall, our results highlight the usefulness of the proposed GS estimator in improving the accuracy of unbiased offline evaluation for recommendation systems. The theoretical analysis and empirical evaluation demonstrate its effectiveness in handling implicit feedback datasets and its potential to improve upon existing methods. Our findings suggest that the proposed GS estimator can be a valuable tool for both researchers and practitioners in the field.

REFERENCES

[1] Bahare Askari, Jaroslaw Szlichta, and Amirali Salehi-Abari. 2021. Variational Autoencoders for Top-K Recommendation with Implicit Feedback. In *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*, Fernando Diaz, Chirag Shah, Torsten Suel, Pablo Castells, Rosie Jones, and Tetsuya Sakai (Eds.). ACM, 2061–2065. https://doi.org/10.1145/3404835.3462986

[2] Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. 2013. *Concentration Inequalities - A Nonasymptotic Theory of Independence.* Oxford University Press. https://doi.org/10.1093/acprof:oso/9780199535255.001.0001

[3] Rocío Cañamares, Pablo Castells, and Alistair Moffat. 2020. Offline evaluation options for recommender systems. *Inf. Retr. J.* 23, 4 (2020), 387–410. https://doi.org/10.1007/s10791-020-09371-3

[4] 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 *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval.* 21–30.

[5] Paolo Cremonesi, Yehuda Koren, and Roberto Turrin. 2010. Performance of recommender algorithms on top-n recommendation tasks. In *Proceedings of the 2010 ACM Conference on Recommender Systems, RecSys 2010, Barcelona, Spain, September 26-30, 2010*, Xavier Amatriain, Marc Torrens, Paul Resnick, and Markus Zanker (Eds.). ACM, 39–46. https://doi.org/10.1145/1864708.1864721

[6] Maurizio Ferrari Dacrema, Paolo Cremonesi, and Dietmar Jannach. 2019. Are we really making much progress? A worrying analysis of recent neural recommendation approaches. In *Proceedings of the 13th ACM Conference on Recommender Systems, RecSys 2019, Copenhagen, Denmark, September 16-20, 2019*, Toine Bogers, Alan Said, Peter Brusilovsky, and Domonkos Tikk (Eds.). ACM, 101–109. https://doi.org/10.1145/3298689.3347058

[7] Nicolò Felicioni. 2022. Enhancing Counterfactual Evaluation and Learning for Recommendation Systems. In *RecSys '22: Sixteenth ACM Conference on Recommender Systems, Seattle, WA, USA, September 18 - 23, 2022*, Jennifer Golbeck, F. Maxwell Harper, Vanessa Murdock, Michael D. Ekstrand, Bracha Shapira, Justin Basilico, Keld T. Lundgaard, and Even Oldridge (Eds.). ACM, 739–741. https://doi.org/10.1145/3523227.3547429

[8] Alexandre Gilotte, Clément Calauzènes, Thomas Nedelec, Alexandre Abraham, and Simon Dollé. 2018. Offline A/B Testing for Recommender Systems. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, WSDM 2018, Marina Del Rey, CA, USA, February 5-9, 2018*, Yi Chang, Chengxiang Zhai, Yan Liu, and Yoelle Maarek (Eds.). ACM, 198–206. https://doi.org/10.1145/3159652.3159687

[9] Alois Gruson, Praveen Chandar, Christophe Charbuillet, James McInerney, Samantha Hansen, Damien Tardieu, and Ben Carterette. 2019. Offline Evaluation to Make Decisions About PlaylistRecommendation Algorithms. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, WSDM 2019, Melbourne, VIC, Australia, February 11-15, 2019*, J. Shane Culpepper, Alistair Moffat, Paul N. Bennett, and Kristina Lerman (Eds.). ACM, 420–428. https://doi.org/10.1145/3289600.3291027

[10] Yifan Hu, Yehuda Koren, and Chris Volinsky. 2008. Collaborative Filtering for Implicit Feedback Datasets. In *Proceedings of the 8th IEEE International Conference on Data Mining (ICDM 2008), December 15-19, 2008, Pisa, Italy.* IEEE Computer Society, 263–272. https://doi.org/10.1109/ICDM.2008.22

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

[12] Amir Hossein Jadidinejad, Craig Macdonald, and Iadh Ounis. 2022. The Simpson's Paradox in the Offline Evaluation of Recommendation Systems. *ACM Trans. Inf. Syst.* 40, 1 (2022), 4:1–4:22. https://doi.org/10.1145/3458509

[13] Olivier Jeunen. 2019. Revisiting offline evaluation for implicit-feedback recommender systems. In *Proceedings of the 13th ACM Conference on Recommender Systems, RecSys 2019, Copenhagen, Denmark, September 16-20, 2019*, Toine Bogers, Alan Said, Peter Brusilovsky, and Domonkos Tikk (Eds.). ACM, 596–600. https://doi.org/10.1145/3298689.3347069

[14] Thorsten Joachims, Adith Swaminathan, and Tobias Schnabel. 2017. Unbiased learning-to-rank with biased feedback. In *Proceedings of the tenth ACM international conference on web search and data mining.* 781–789.

[15] Maurice George Kendall. 1938. A New Measure of Rank Correlation. *Biometrika* 30, 1-2 (1938), 81–93. https://doi.org/10.1093/biomet/30.1-2.81 arXiv:https://academic.oup.com/biomet/article-pdf/30/1-2/81/423380/30-1-2-81.pdf

[16] Yehuda Koren. 2008. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, Nevada, USA, August 24-27, 2008*, Ying Li, Bing Liu, and Sunita Sarawagi (Eds.). ACM, 426–434. https://doi.org/10.1145/1401890.1401944

[17] Yehuda Koren, Robert M. Bell, and Chris Volinsky. 2009. Matrix Factorization Techniques for Recommender Systems. *Computer* 42, 8 (2009), 30–37. https://doi.org/10.1109/MC.2009.263

[18] Daniel D. Lee and H. Sebastian Seung. 2000. Algorithms for Non-negative Matrix Factorization. In *Advances in Neural Information Processing Systems 13, Papers from Neural Information Processing Systems (NIPS) 2000, Denver, CO, USA*, Todd K. Leen, Thomas G. Dietterich, and Volker Tresp (Eds.). MIT Press, 556–562. https://proceedings.neurips.cc/paper/2000/hash/f9d1152547c0bde01830b7e8bd60024c-Abstract.html

[19] Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. 2020. Offline Reinforcement Learning: Tutorial, Review, and Perspectives on Open Problems. *CoRR* abs/2005.01643 (2020). arXiv:2005.01643 https://arxiv.org/abs/2005.01643

[20] Dawen Liang, Rahul G. Krishnan, Matthew D. Hoffman, and Tony Jebara. 2018. Variational Autoencoders for Collaborative Filtering. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web, WWW 2018, Lyon, France, April 23-27, 2018*, Pierre-Antoine Champin, Fabien Gandon, Mounia Lalmas, and Panagiotis G. Ipeirotis (Eds.). ACM, 689–698. https://doi.org/10.1145/3178876.3186150

[21] Benjamin M. Marlin and Richard S. Zemel. 2009. Collaborative prediction and ranking with non-random missing data. In *Proceedings of the 2009 ACM Conference on Recommender Systems, RecSys 2009, New York, NY, USA, October 23-25, 2009*, Lawrence D. Bergman, Alexander Tuzhilin, Robin D. Burke, Alexander Felfernig, and Lars Schmidt-Thieme (Eds.). ACM, 5–12. https://doi.org/10.1145/1639714.1639717

[22] Andreas Maurer and Massimiliano Pontil. 2009. Empirical Bernstein Bounds and Sample-Variance Penalization. In *COLT 2009 - The 22nd Conference on Learning Theory, Montreal, Quebec, Canada, June 18-21, 2009.* http://www.cs.mcgill.ca/%7Ecolt2009/papers/012.pdf#page=1

[23] Harrie Oosterhuis, Rolf Jagerman, and Maarten de Rijke. 2020. Unbiased Learning to Rank: Counterfactual and Online Approaches. In *Companion of The 2020 Web Conference 2020, Taipei, Taiwan, April 20-24, 2020*, Amal El Fallah Seghrouchni, Gita Sukthankar, Tie-Yan Liu, and Maarten van Steen (Eds.). ACM / IW3C2, 299–300. https://doi.org/10.1145/3366424.3383107

[24] Rong Pan, Yunhong Zhou, Bin Cao, Nathan Nan Liu, Rajan M. Lukose, Martin Scholz, and Qiang Yang. 2008. One-Class Collaborative Filtering. In *Proceedings of the 8th IEEE International Conference on Data Mining (ICDM 2008), December 15-19, 2008, Pisa, Italy.* IEEE Computer Society, 502–511. https://doi.org/10.1109/ICDM.2008.16

[25] Judea Pearl. 2009. *Causality.* Cambridge university press.

[26] Doina Precup, Richard S. Sutton, and Satinder Singh. 2000. Eligibility Traces for Off-Policy Policy Evaluation. In *Proceedings of the Seventeenth International Conference on Machine Learning (ICML 2000), Stanford University, Stanford, CA, USA, June 29 - July 2, 2000*, Pat Langley (Ed.). Morgan Kaufmann, 759–766.

[27] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian Personalized Ranking from Implicit Feedback. In *UAI 2009, Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, Montreal, QC, Canada, June 18-21, 2009*, Jeff A. Bilmes and Andrew Y. Ng (Eds.). AUAI Press, 452–461. https://dslpitt.org/uai/displayArticleDetails.jsp?mmnu=1&smnu=2&article_id=1630&proceeding_id=25

[28] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian Personalized Ranking from Implicit Feedback. In *UAI 2009, Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, Montreal, QC, Canada, June 18-21, 2009*, Jeff A. Bilmes and Andrew Y. Ng (Eds.). AUAI Press, 452–461. https://www.auai.org/uai2009/papers/UAI2009_0139_48141db02b9f0b02bc7158819ebfa2c7.pdf

[29] Yuta Saito. 2020. Asymmetric Tri-training for Debiasing Missing-Not-At-Random Explicit Feedback. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, Jimmy X. Huang, Yi Chang, Xueqi Cheng, Jaap Kamps, Vanessa Murdock, Ji-Rong Wen, and Yiqun Liu (Eds.). ACM, 309–318. https://doi.org/10.1145/3397271.3401114

[30] Yuta Saito and Masahiro Nomura. 2022. Towards Resolving Propensity Contradiction in Offline Recommender Learning. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022, Vienna, Austria, 23-29 July 2022*, Luc De Raedt (Ed.). ijcai.org, 2211–2217. https://doi.org/10.24963/ijcai.2022/307

[31] Ruslan Salakhutdinov and Andriy Mnih. 2007. Probabilistic Matrix Factorization. In *Advances in Neural Information Processing Systems 20, Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 3-6, 2007*, John C. Platt, Daphne Koller, Yoram Singer, and Sam T. Roweis (Eds.). Curran Associates, Inc., 1257–1264. https://proceedings.neurips.cc/paper/2007/hash/d7322ed717dedf1eb4e6e52a37ea7bcd-Abstract.html

[32] Sven Schmit and Ramesh Johari. 2018. Learning with abandonment. In *International Conference on Machine Learning.* PMLR, 4509–4517.

[33] Tobias Schnabel, Adith Swaminathan, Ashudeep Singh, Navin Chandak, and Thorsten Joachims. 2016. Recommendations as Treatments: Debiasing Learning and Evaluation. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016 (JMLR Workshop and Conference Proceedings, Vol. 48)*, Maria-Florina Balcan and Kilian Q. Weinberger (Eds.). JMLR.org, 1670–1679. http://proceedings.mlr.press/v48/schnabel16.html

[34] Guy Shani and Asela Gunawardana. 2011. Evaluating Recommendation Systems. In *Recommender Systems Handbook*, Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul B. Kantor (Eds.). Springer, 257–297. https://doi.org/10.1007/978-0-387-85820-3_8

[35] Harald Steck. 2011. Item popularity and recommendation accuracy. In *Proceedings of the 2011 ACM Conference on Recommender Systems, RecSys 2011, Chicago, IL, USA, October 23-27, 2011*, Bamshad Mobasher, Robin D. Burke, Dietmar Jannach, and Gediminas Adomavicius (Eds.). ACM, 125–132. https://doi.org/10.1145/2043932.2043957

[36] Harald Steck. 2013. Evaluation of recommendations: rating-prediction and ranking. In *Seventh ACM Conference on Recommender Systems, RecSys '13, Hong Kong, China, October 12-16, 2013*, Qiang Yang, Irwin King, Qing Li, Pearl Pu, and George Karypis (Eds.). ACM, 213–220. https://doi.org/10.1145/2507157.2507160

[37] Philip S. Thomas, Georgios Theocharous, and Mohammad Ghavamzadeh. 2015. High-Confidence Off-Policy Evaluation. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA*, Blai Bonet and Sven Koenig (Eds.). AAAI Press, 3000–3006. http://www.aaai.org/ocs/index.php/AAAI/AAAI15/paper/view/10042

[38] Yixin Wang, Dawen Liang, Laurent Charlin, and David M Blei. 2020. Causal inference for recommender systems. In *Proceedings of the 14th ACM Conference on Recommender Systems*. 426–431.

[39] Markus Weimer, Alexandros Karatzoglou, and Alexander J. Smola. 2008. Improving maximum margin matrix factorization. *Mach. Learn.* 72, 3 (2008), 263–276. https://doi.org/10.1007/s10994-008-5073-7

[40] Bernard L Welch. 1947. The generalization of "Student's" problem when several different population varlances are involved. *Biometrika* 34, 1-2 (1947), 28–35.

[41] Longqi Yang, Yin Cui, Yuan Xuan, Chenyang Wang, Serge J. Belongie, and Deborah Estrin. 2018. Unbiased offline recommender evaluation for missing-not-at-random implicit feedback. In *Proceedings of the 12th ACM Conference on Recommender Systems, RecSys 2018, Vancouver, BC, Canada, October 2-7, 2018*, Sole Pera, Michael D. Ekstrand, Xavier Amatriain, and John O'Donovan (Eds.). ACM, 279–287. https://doi.org/10.1145/3240323.3240355