# UNCERTAINTY-AWARE OFF POLICY LEARNING

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Off-policy learning, referring to the procedure of policy optimization with access only to logged feedback data, has shown importance in various real-world applications, such as search engines, recommender systems, etc. While the ground-truth logging policy, which generates the logged data, is usually unknown, previous work directly takes its estimated value in off-policy learning, resulting in a biased estimator. This estimator has both high bias and variance on samples with small and inaccurate estimated logging probabilities. In this work, we explicitly model the uncertainty in the estimated logging policy and propose a novel Uncertainty-aware Inverse Propensity Score estimator (UIPS) for improved off-policy learning. Experiment results on synthetic and three real-world recommendation datasets demonstrate the advantageous sample efficiency of the proposed UIPS estimator.

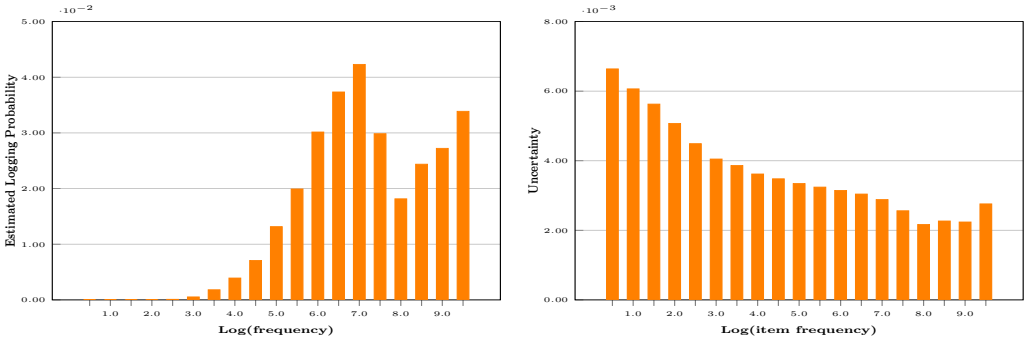
## 1 INTRODUCTION

In many real-world applications, including search engines (Agarwal et al. (2019)), online advertisements (Strehl et al. (2010)), recommender systems (Chen et al. (2019); Liu et al. (2022)), only logged feedback data is available for subsequent policy optimization. For example, in recommender systems, various complicated recommendation models (i.e., policies) (Zhou et al. (2018); Guo et al. (2017)) were optimized with the logged user interactions (e.g., clicks or staytime) to items recommended by previous recommendation policies. However, such logged data is known to be biased, since one does not know the feedback on items that previous policy (which is generally referred as the *logging policy*) did not take. This inevitably distorts the evaluation and optimization of a new policy when it tends to select items not in the logged data.

Off-policy learning emerges as a favorable way to learn an improved policy only from logged data by addressing the mismatch between the learning policy and the logging policy. One of the most commonly used off-policy learning methods is Inverse Propensity Scoring (IPS) (Chen et al. (2019); Munos et al. (2016)), which assigns per-sample importance weight to the training objective on the logged data, so as to get an unbiased optimization objective in expectation. The importance weight in IPS is the probability ratio between the learning policy and the logging policy.

However, the ground-truth logging policy is unavailable to the learner, e.g., it is not recorded in the data. One common treatment taken by previous work (Strehl et al. (2010); Liu et al. (2022); Chen et al. (2019); Ma et al. (2020)) is to first employ a supervised learning method (e.g., logistic regression, neural networks, etc.) to estimate the logging policy, and then take the estimated logging policy for off-policy learning. We theoretically show that such an approximation results in a biased estimator which is sensitive to those inaccurate and small estimated logging probabilities. Worse still, the small values of the estimated logging probabilities usually mean that there are fewer related samples in the logged data, so its estimation usually has high uncertainties, i.e., inaccurate estimation with high probability. Figure 1 shows a piece of empirical evidence from a large-scale recommendation benchmark KuaiRec dataset (Gao et al. (2022)), where items with lower frequencies in the logged dataset have lower estimated logging probabilities and higher uncertainties concurrently. The high bias and variance caused by these samples greatly hinder the performance of off-policy learning.

In this work, we explicitly take the uncertainty of the estimated logging policy into consideration and design a novel Ucertainty-aware Inverse Propensity Score estimator (UIPS) as the optimization objective for policy learning. UIPS introduces an additional weight to approach the ground-truth propensity from the estimated one, and learns an improved policy by alternating: (1) Find the optimal weight that makes the estimator as accurate as possible, taking into consideration the uncertainty




(a) Estimated Logging Probability (b) Uncertainty of Estimation

Figure 1: Estimated logging policy and its uncertainty under different item frequency on KuaiRec.

of estimated logging policy; (2) Improve the policy by optimizing the resulting estimator. We further find a closed-form solution for the optimal weight by deriving an upper bound on the mean squared error (MSE) to the ground-truth policy value. The optimal weight adjusts sample weights considering both the uncertainty of estimated logging probabilities and the propensity scores, rather than simply boosting or penalizing samples with high uncertain logging probabilities. Experiment results on synthetic and three real-world recommendation datasets demonstrate the sample efficiency of UIPS. All data and code can be found in supplementary materials for reproducibility.

To summarize, our contribution in this work is as follows:

- We point out that directly using the estimated logging policy leads to sub-optimal off-policy learning, since the resulting biased estimator is greatly distorted by samples with inaccurate and small estimated logging probabilities.
- We take the uncertainty of estimated logging policy into consideration and propose UIPS for more accurate off-policy learning.
- Experiments on synthetic and three real-world recommendation datasets demonstrate UIPS's strong advantage in sample efficiency in off-policy learning.

## 2 Preliminary: Off-Policy Learning

We focus on the standard contextual bandit setup to explain the key concepts. Following convention (Joachims et al. (2018); Saito & Joachims (2022); Su et al. (2020)), let $\boldsymbol{x} \in \mathcal{X} \subseteq R^d$ be a $d$-dimensional context vector drawn from an unknown probability distribution $p(\boldsymbol{x})$. Each context is associated with a finite set of actions denoted by $\mathcal{A}$, where $|\mathcal{A}| < \infty$. Let $\pi : \mathcal{A} \times \mathcal{X} \to [0, 1]$ denote a stochastic policy, such that $\pi(a|\boldsymbol{x})$ is the probability of selecting action $a$ under context $\boldsymbol{x}$ and $\sum_{a \in \mathcal{A}} \pi(a|\boldsymbol{x}) = 1$. Under a given context, reward $r_{\boldsymbol{x},a}$ is observed when action $a$ is chosen. Take news recommendation for example, $\boldsymbol{x}$ represents the state of a user, summarizing his/her interaction history with the recommender system, each action $a$ is a candidate news article, the policy is a recommendation algorithm, and the reward $r_{\boldsymbol{x},a}$ denotes the user feedback on article $a$, e.g., whether the user clicks the article. Let $V(\pi)$ denote the expected reward or value of the policy $\pi$:

$$V(\pi) = \mathbb{E}_{\boldsymbol{x} \sim p(\boldsymbol{x}), a \sim \pi(a|\boldsymbol{x})}[r_{\boldsymbol{x},a}]. \tag{1}$$

We look for a policy $\pi(a|\boldsymbol{x})$ to maximize $V(\pi)$. In the rest of the paper, we denote $\mathbb{E}_{\boldsymbol{x} \sim p(\boldsymbol{x}), a \sim \pi(a|\boldsymbol{x})}[\cdot]$ as $\mathbb{E}_{\pi}[\cdot]$ for simplicity.

In contrast to performing online updates by following the learning policy $\pi(a|\boldsymbol{x})$, in off-policy learning we can only access a set of logged feedback data denoted by $D := \{(\boldsymbol{x}_n, a_n, r_{\boldsymbol{x}_n,a_n}) | n \in [N]\}$, where $[N] := \{1, \ldots, N\}$. Given $\boldsymbol{x}_n$, the action $a_n$ was generated by a stochastic logging policy $\beta^*$, i.e., the probability action $a_n$ was selected is $\beta^*(a_n|\boldsymbol{x}_n)$. The actions $\{a_1, \ldots, a_N\}$ and their corresponding rewards $\{r_{\boldsymbol{x}_1,a_1}, \ldots, r_{\boldsymbol{x}_N,a_N}\}$ are generated independently given $\beta^*$. Due to the nature of policy optimization, the learning policy $\pi(a|\boldsymbol{x})$ is expected to be different from $\beta^*(a|\boldsymbol{x})$, unless $\beta^*(a|\boldsymbol{x})$ is already optimal. Moreover, in practice the situation could be further complicated. Again, consider the news recommendation scenario. Due to the scalability requirement, industrial

recommender systems usually adopt a two-stage framework (Ma et al. (2020)), where one or several candidate generation models first produce a candidate set and a separate ranking model reranks candidate items to present top-K item to users. While $\beta^*(a|\boldsymbol{x})$ depicts the whole two-stage process, the learning policy $\pi(a|\boldsymbol{x})$ is usually employed in one particular stage (e.g., the reranking stage), implying drastic differences between the logging and learning policies. The main challenge of off-policy learning is then to address the distribution discrepancy between $\beta^*(a|\boldsymbol{x})$ and $\pi(a|\boldsymbol{x})$, and learn a policy $\pi(a|\boldsymbol{x})$ to maximize $V(\pi)$ with access only to the logged dataset $D$.

One of most widely used methods to address the distribution shift between $\pi(a|\boldsymbol{x})$ and $\beta^*(a|\boldsymbol{x})$ is the Inverse Propensity Score (IPS) (Chen et al. (2019); Munos et al. (2016)). One can easily get that:

$$V(\pi) = \mathbb{E}_{\beta^*}\left[\frac{\pi(a|\boldsymbol{x})}{\beta^*(a|\boldsymbol{x})} r_{\boldsymbol{x},a}\right],$$

yielding the following empirical estimator of $V(\pi)$:

$$\hat{V}_{\mathrm{IPS}}(\pi) = \frac{1}{N}\sum_{n=1}^{N} \frac{\pi(a_n|\boldsymbol{x}_n)}{\beta^*(a_n|\boldsymbol{x}_n)} r_{\boldsymbol{x}_n,a_n}, \tag{2}$$

where $\pi(a_n|\boldsymbol{x}_n)/\beta^*(a_n|\boldsymbol{x}_n)$ is referred to as the propensity score. In the rest of paper, without further specification, we use the empirical estimation of expectation in our practical calculation. Various algorithms can be readily used for policy optimization under $\hat{V}_{\mathrm{IPS}}(\pi)$, including value-based methods (Silver et al. (2016)), policy-based methods (Levine & Koltun (2013); Schulman et al. (2015); Williams (1992)). In this work, we adopt a well-known policy gradient algorithm, REINFORCE (Williams (1992)). Assume policy $\pi(a|\boldsymbol{x})$ is parameterized by $\boldsymbol{\vartheta}$, via the "log-trick", the gradient of $\hat{V}_{\mathrm{IPS}}(\pi_{\boldsymbol{\vartheta}})$ with respect to $\boldsymbol{\vartheta}$ can be readily derived as follows:

$$\nabla_{\boldsymbol{\vartheta}} \hat{V}_{\mathrm{IPS}}(\pi_{\boldsymbol{\vartheta}}) = \frac{1}{N}\sum_{n=1}^{N} \frac{\pi(a_n|\boldsymbol{x}_n)}{\beta^*(a_n|\boldsymbol{x}_n)} r_{\boldsymbol{x}_n,a_n} \nabla_{\boldsymbol{\vartheta}} \log(\pi_{\boldsymbol{\vartheta}}(a_n|\boldsymbol{x}_n)). \tag{3}$$

**Approximation with unknown logging policy**. In many real-world applications, the logging policy, i.e., the $\beta^*(a|\boldsymbol{x})$ of each observation $(\boldsymbol{x}, a)$, is unknown. One reason is the legacy issue, i.e., the probabilities were not logged when collecting data. Another reason is that the exact value of $\beta^*(a|\boldsymbol{x})$ is intrinsically unavailable such as in the two-stage recommender systems. As the solution, previous work employs various supervised learning methods (e.g., logistic regression (Schnabel et al. (2016)), nerural networks (Chen et al. (2019), etc.) to estimate the logging policy, and replaces $\beta^*(a|\boldsymbol{x})$ with its estimated value $\hat{\beta}(a|\boldsymbol{x})$ to get the following estimator for policy learning:

$$\hat{V}_{\mathrm{BIPS}}(\pi_{\boldsymbol{\vartheta}}) = \frac{1}{N}\sum_{n=1}^{N} \frac{\pi_{\boldsymbol{\vartheta}}(a_n|\boldsymbol{x}_n)}{\hat{\beta}(a_n|\boldsymbol{x}_n)} r_{\boldsymbol{x}_n,a_n}. \tag{4}$$

However, as shown in the following proposition, inaccurate $\hat{\beta}(a|\boldsymbol{x})$ leads to high bias and variance of $\hat{V}_{\mathrm{BIPS}}(\pi_{\boldsymbol{\vartheta}})$. Worse still, smaller inaccurate $\hat{\beta}(a|\boldsymbol{x})$ further enlarges the this bias and variance.

**Proposition 1.** *The bias and variance of $\hat{V}_{\mathrm{BIPS}}(\pi_{\boldsymbol{\vartheta}})$ can be derived as follows:*

$$\mathrm{Bias}\left(\hat{\mathrm{V}}_{\mathrm{BIPS}}(\pi_{\boldsymbol{\vartheta}})\right) = \mathbb{E}_D\left[\hat{V}_{\mathrm{BIPS}}(\pi_{\boldsymbol{\vartheta}}) - V(\pi_{\boldsymbol{\vartheta}})\right] = \mathbb{E}_{\pi_{\boldsymbol{\vartheta}}}\left[r_{\boldsymbol{x},a}\left(\frac{\beta^*(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})} - 1\right)\right]$$

$$N \cdot \mathrm{Var}_D\left(\hat{V}_{\mathrm{BIPS}}(\pi_{\boldsymbol{\vartheta}})\right) = \mathrm{Var}_{\pi_{\boldsymbol{\vartheta}}}\left(\frac{\beta^*(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})} r_{\boldsymbol{x},a}\right) + \mathbb{E}_{\pi_{\boldsymbol{\vartheta}}}\left[\left(\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\beta^*(a|\boldsymbol{x})} - 1\right) \cdot \frac{\beta^*(a|\boldsymbol{x})^2}{\hat{\beta}(a|\boldsymbol{x})^2} r^2_{\boldsymbol{x},a}\right]$$

Smaller $\hat{\beta}(a|\boldsymbol{x})$ usually implies fewer related training samples in the logged data, and thus $\hat{\beta}(a|\boldsymbol{x})$ will be inaccurate with a higher probability. To make it more explicit, we take KuaiRec dataset (Gao et al. (2022)) as an example and estimate the logging policy following (Chen et al. (2019)). Figure 1 shows the estimated $\hat{\beta}(a|\boldsymbol{x})$ and its corresponding uncertainties in items of different observation frequencies in the logged dataset. As uncertainty measures how large the confidence interval is about the current estimation, higher uncertainty implies that the true value may be away from the empirical mean estimate with a high probability. We defer the discussion about our detailed uncertainty calculation in Section 3. We can observe from Figure 1 that as item frequency decreases, the estimated logging probability also decreases, but the estimation uncertainty increases. This implies that smaller $\hat{\beta}(a|\boldsymbol{x})$ is usually 1) more inaccurate and 2) associated with high uncertainty.

As a result, with high bias and variance caused by inaccurate $\hat{\beta}(a|\boldsymbol{x})$, it is erroneous to improve $\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})$ by simply optimizing $\hat{V}_{\mathrm{BIPS}}(\pi_{\boldsymbol{\vartheta}})$. We propose uncertainty-aware off-policy learning to address this challenge.

# 3 UNCERTAINTY-AWARE OFF-POLICY LEARNING

Our idea is incorporating the uncertainty of the logging policy estimation into policy learning. Observing that

$$V(\pi_{\boldsymbol{\vartheta}}) = \mathbb{E}_{\beta^*}\left[\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})} \cdot \frac{\hat{\beta}(a|\boldsymbol{x})}{\beta^*(a|\boldsymbol{x})} \cdot r_{\boldsymbol{x},a}\right],$$

we propose to learn the optimal policy by optimizing the following empirical estimator:

$$\hat{V}_{\mathrm{UIPS}}(\pi_{\boldsymbol{\vartheta}}) = \frac{1}{N}\sum_{n=1}^{N} \frac{\pi_{\boldsymbol{\vartheta}}(a_n|\boldsymbol{x}_n)}{\hat{\beta}(a_n|\boldsymbol{x}_n)} \cdot \phi_{\boldsymbol{x}_n,a_n} \cdot r_{\boldsymbol{x}_n,a_n} \tag{5}$$

where $\phi_{\boldsymbol{x}_n,a_n}$ is a weight, which reflects $\hat{\beta}(a_n|\boldsymbol{x}_n)/\beta^*(a_n|\boldsymbol{x}_n)$, to be selected to make $\hat{V}_{\mathrm{UIPS}}(\pi_{\boldsymbol{\vartheta}})$ as close to $V(\pi_{\boldsymbol{\vartheta}})$ as possible. Intuitively, one should give small weights to samples whose $\hat{\beta}(a|\boldsymbol{x})$ is far below the ground-truth $\beta^*(a|\boldsymbol{x})$. Thus, we divide offline policy improvement into two steps, and repeat them until certain convergence condition is met:

- **Uncertainty aware policy evaluation**: Derive the optimal uncertainty aware $\phi_{\boldsymbol{x},a}$ to make $\hat{V}_{\mathrm{UIPS}}(\pi_{\boldsymbol{\vartheta}})$ as accurate as possible.
- **Policy Improvement**: Learn an improved policy $\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})$ by optimizing $\hat{V}_{\mathrm{UIPS}}(\pi_{\boldsymbol{\vartheta}})$.

## 3.1 Uncertainty Aware Policy Evaluation

**Optimal uncertainty aware weight $\phi_{\boldsymbol{x},a}$.** We measure the accuracy of $\hat{V}_{\mathrm{UIPS}}(\pi_{\boldsymbol{\vartheta}})$ by its mean squared error (MSE) to $V(\pi_{\boldsymbol{\vartheta}})$ following previous work (Su et al. (2020); Saito & Joachims (2022)). MSE captures both the bias and variance of an estimator, since it is the summation of squared bias and variance. We then locate the $\phi_{\boldsymbol{x},a}$ that can minimize the MSE. In particular, we demonstrate the optimal $\phi_{\boldsymbol{x},a}$ has a closed-form formula which relates to both the value of $\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})/\hat{\beta}(a|\boldsymbol{x})$ and the estimation uncertainty of $\hat{\beta}(a|\boldsymbol{x})$.

More specifically, instead of directly minimizing the MSE, which is intractable, we find the desirable $\phi_{\boldsymbol{x},a}$ by minimizing the upper bound of MSE in the following theorem.

**Theorem 1.** *Assume $r_{\boldsymbol{x},a} \in [0,1]$, the mean squared error (*MSE*) between $\hat{V}_{\mathrm{UIPS}}(\pi_{\boldsymbol{\vartheta}})$ and ground-truth estimator $V(\pi_{\boldsymbol{\vartheta}})$ is upper bounded as follows:*

$$\begin{aligned}\mathrm{MSE}\left(\hat{V}_{\mathrm{UIPS}}(\pi_{\boldsymbol{\vartheta}})\right) &= \mathbb{E}_D\left[\left(\hat{V}_{\mathrm{UIPS}}(\pi_{\boldsymbol{\vartheta}}) - V(\pi_{\boldsymbol{\vartheta}})\right)^2\right] = \mathrm{Bias}\left(\hat{V}_{\mathrm{UIPS}}(\pi_{\boldsymbol{\vartheta}})\right)^2 + \mathrm{Var}\left(\hat{V}_{\mathrm{UIPS}}(\pi_{\boldsymbol{\vartheta}})\right) \\ &\leq \mathbb{E}_{\pi_{\boldsymbol{\vartheta}}}\left[r^2_{\boldsymbol{x},a}\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\beta^*(a|\boldsymbol{x})}\right] \cdot \mathbb{E}_{\beta^*}\left[\left(\frac{\beta^*(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})}\phi_{\boldsymbol{x},a} - 1\right)^2\right] + \mathbb{E}_{\beta^*}\left[\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})^2}{\hat{\beta}(a|\boldsymbol{x})^2}\phi^2_{\boldsymbol{x},a}\right]\end{aligned}$$

The upper bound in Theorem 1 strictly increases with the two expectations related to $\phi_{\boldsymbol{x},a}$, which implies that for some choice $\lambda \in [0,\infty]$, the MSE-optimizing $\phi_{\boldsymbol{x},a}$ can be derived by minimizing:

$$\lambda\mathbb{E}_{\beta^*}\left[\left(\frac{\beta^*(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})}\phi_{\boldsymbol{x},a} - 1\right)^2\right] + \mathbb{E}_{\beta^*}\left[\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})^2}{\hat{\beta}(a|\boldsymbol{x})^2}\phi^2_{\boldsymbol{x},a}\right]. \tag{6}$$

We cannot directly minimize Eq.(6) since the unknown $\beta^*(a|\boldsymbol{x})$ is involved. However, various ways (Gal & Ghahramani (2016); Xu et al. (2021)) can be employed to get the confidence interval which will contain $\beta^*(a|\boldsymbol{x})$ with high probability. More specifically, following previous work (Joachims et al. (2018)), we assume $\beta^*(a|\boldsymbol{x})$ can be modelled by a softmax function on top of an unknown function $f_{\boldsymbol{\theta}^*}(\boldsymbol{x},a)$, i.e., the realizable assumption. Then we can get:

$$\beta^*(a|\boldsymbol{x}) = \frac{\exp(f_{\boldsymbol{\theta}^*}(\boldsymbol{x},a))}{\sum_{a'}\exp(f_{\boldsymbol{\theta}^*}(\boldsymbol{x},a'))}, \quad \hat{\beta}(a|\boldsymbol{x}) = \frac{\exp(f_{\boldsymbol{\theta}}(\boldsymbol{x},a))}{\sum_{a'}\exp(f_{\boldsymbol{\theta}}(\boldsymbol{x},a'))}, \tag{7}$$

where $f_{\boldsymbol{\theta}}(\boldsymbol{x},a)$ is an estimate of $f_{\boldsymbol{\theta}^*}(\boldsymbol{x},a)$. Following the conventional definition of confidence interval (Abbasi-Yadkori et al. (2011)), we define $\gamma$ and $U_{\boldsymbol{x},a}$ such that $|f_{\boldsymbol{\theta}^*}(\boldsymbol{x},a) - f_{\boldsymbol{\theta}}(\boldsymbol{x},a)| \leq \gamma U_{\boldsymbol{x},a}$ holds with probability at least 1-$\delta$, where $\gamma$ is a function of $\delta$ (typically the smaller $\delta$ is, the larger $\gamma$ is). Then $\gamma U_{\boldsymbol{x},a}$ measures the width of confidence interval of $f_{\boldsymbol{\theta}}(\boldsymbol{x},a)$ against its ground-truth $f_{\boldsymbol{\theta}^*}(\boldsymbol{x},a)$. This implies that $\beta^*(a|\boldsymbol{x}) \in \boldsymbol{B}_{\boldsymbol{x},a}$ with probability at least 1-$\delta$, where:

$$\boldsymbol{B}_{\boldsymbol{x},a} = \left[\frac{\hat{Z}\exp(-\gamma U_{\boldsymbol{x},a})}{Z^*}\hat{\beta}(a|\boldsymbol{x}), \frac{\hat{Z}\exp(\gamma U_{\boldsymbol{x},a})}{Z^*}\hat{\beta}(a|\boldsymbol{x})\right], \quad Z^* = \sum_{a'}\exp(f_{\theta^*}(a'|\boldsymbol{x})), \hat{Z} = \sum_{a'}\exp(f_{\theta}(a'|\boldsymbol{x})).$$

Since $\beta^*(a|\boldsymbol{x})$ can be any value in $\boldsymbol{B}_{\boldsymbol{x},a}$, we adopt the idea of robust optimization (Chen et al. (2020)) and find the optimal $\phi_{\boldsymbol{x},a}$ by solving the following optimization problem:

$$\min_{\phi_{\boldsymbol{x},a}} \max_{\beta_{\boldsymbol{x},a} \in \boldsymbol{B}_{\boldsymbol{x},a}} \quad \lambda \mathbb{E}_{\beta^*}\left[\left(\frac{\beta_{\boldsymbol{x},a}}{\hat{\beta}(a|\boldsymbol{x})}\phi_{\boldsymbol{x},a} - 1\right)^2\right] + \mathbb{E}_{\beta^*}\left[\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})^2}{\hat{\beta}(a|\boldsymbol{x})^2}\phi^2_{\boldsymbol{x},a}\right]. \tag{8}$$

The following theorem derives a closed-form formula for the optimal solution of (8) .

**Theorem 2.** *Let $\eta_1, \eta_2 \in [\exp(-\gamma U^{\max}_{\boldsymbol{x}}), \exp(\gamma U^{\max}_{\boldsymbol{x}})]$, where $U^{\max}_{\boldsymbol{x}} = \max_a U_{\boldsymbol{x},a}$. The optimization problem in Eq.(8) has a closed-form solution as follows:*

$$\phi^*_{\boldsymbol{x},a} = \min\left(\lambda / \left[\frac{\lambda}{\eta_1}\exp\left(-\gamma U_{\boldsymbol{x},a}\right) + \frac{\eta_1 \pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})^2}{\hat{\beta}(a|\boldsymbol{x})^2 \exp(-\gamma U_{\boldsymbol{x},a})}\right], 2\eta_2 / \left[\exp\left(\gamma U_{\boldsymbol{x},a}\right) + \exp\left(-\gamma U_{\boldsymbol{x},a}\right)\right]\right)$$

**Insights on $\phi^*_{\boldsymbol{x},a}$.** The second term of $\phi^*_{\boldsymbol{x},a}$ (i.e., $2\eta_2 / \left[\exp\left(\gamma U_{\boldsymbol{x},a}\right) + \exp\left(-\gamma U_{\boldsymbol{x},a}\right)\right]$ ) acts like a capping threshold to ensure $\phi^*_{\boldsymbol{x},a} \leq 2\eta_2$ holds even with small $\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})/\hat{\beta}(a|\boldsymbol{x})$ (as shown in Lemma 1 in Appendix A.1). The key component is the first term, and Lemma 1 implies that:

- If the propensity score $\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})/\hat{\beta}(a|\boldsymbol{x})$ is above the threshold $\sqrt{\lambda}/\eta_1$, UIPS will assign a smaller weight to samples with more uncertain/inaccurate $\hat{\beta}(a|\boldsymbol{x})$ to prevent the distortions from those with a large propensity score but an inaccurate logging probability.
- If the propensity score $\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})/\hat{\beta}(a|\boldsymbol{x})$ is below the threshold (but not small enough to activate the second term), then the propensity score at the worse case (i.e., taking $\boldsymbol{B}^-_{\boldsymbol{x},a} = \hat{\beta}(a|\boldsymbol{x})\hat{Z}\exp\left(-\gamma U_{\boldsymbol{x},a}\right)/Z^*$ as denominator) matters. If the propensity score at the worse case is under control, i.e., $\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})/\boldsymbol{B}^-_{\boldsymbol{x},a} < \sqrt{\lambda}$, a larger $U_{\boldsymbol{x},\boldsymbol{a}}$ implies a small propensity score $\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})/\hat{\beta}(a|\boldsymbol{x})$, and UIPS tends to boost this safe sample with a higher $\phi^*_{\boldsymbol{x},a}$. Otherwise $\phi^*_{\boldsymbol{x},a}$ still decreases as $U_{\boldsymbol{x},a}$ becomes higher.

**Uncertainty estimation.** Now we describe how to calculate $U_{\boldsymbol{x},a}$, i.e., the uncertainty of the estimated $\hat{\beta}(a|\boldsymbol{x})$. In this work, we choose to estimate $\beta^*(a|\boldsymbol{x})$ using a neural network, due to its encouraging representation learning capacity. And various ways (Gal & Ghahramani (2016); Xu et al. (2021)) can be leveraged to perform the uncertainty estimation in a neural network. For example, (Gal & Ghahramani, 2016) proposed to estimate uncertainty using dropout; and (Xu et al., 2021) provided a theoretical bound. Here we adopt the result in (Xu et al. (2021)) due to its computational efficiency and theoretical soundness. Following the proof of Theorem 4.4 in (Xu et al. (2021)), given the logged dataset $D$, we can get with high probability $\exists\gamma$:

$$|f_{\boldsymbol{\theta}}(\boldsymbol{x}_n, a_n) - f_{\boldsymbol{\theta}^*}(\boldsymbol{x}_n, a_n))| \leq \gamma\sqrt{\boldsymbol{g}(\boldsymbol{x}_n, a_n)^T \boldsymbol{M}_D^{-1} \boldsymbol{g}(\boldsymbol{x}_n, a_n)}$$

where $\boldsymbol{g}(\boldsymbol{x}_n, a_n)$ is the gradient of function $f_{\boldsymbol{\theta}}(\boldsymbol{x}_n, a_n)$, i.e., $\boldsymbol{g}(\boldsymbol{x}_n, a_n) = \nabla_{\boldsymbol{\theta}} f_{\boldsymbol{\theta}}(\boldsymbol{x}_n, a_n)$, and $M_D = \sum_{n=1}^N \boldsymbol{g}(\boldsymbol{x}_n, a_n)\boldsymbol{g}(\boldsymbol{x}_n, a_n)^T$, implying $U_{\boldsymbol{x}_n, a_n} = \sqrt{\boldsymbol{g}(\boldsymbol{x}_n, a_n)^T \boldsymbol{M}_D^{-1} \boldsymbol{g}(\boldsymbol{x}_n, a_n)}$.

### 3.2 Policy Improvement

After getting the optimal $\phi^*_{\boldsymbol{x},a}$ as in Theorem 2, the policy $\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})$ can be updated by the following REINFORCE gradient:

$$\nabla_{\vartheta} V_{\text{UIPS}}(\pi_{\boldsymbol{\vartheta}}) = \mathbb{E}_{\beta^*}\left[\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})} \cdot \phi^*_{\boldsymbol{x},a} \cdot r_{\boldsymbol{x},a} \nabla_{\vartheta} \log(\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x}))\right]. \tag{9}$$

UIPS then iterates policy evaluation and policy improvement for policy learning until converge.

## 4 Empirical Evaluation

In this section, we evaluate UIPS on both synthetic datasets and three real-world datasets with unbiased data. We compare UIPS with the following baselines, which can be grouped into five categories:

- **Cross-Entropy (CE)**: A supervised learning method with the cross-entropy loss as its objective, which is the commonly used learning approach for a model with softmax output. No off-policy correction is performed in this method.

- **IPS-Cap** (Chen et al. (2019)): The standard IPS based off-policy learning, which prunes propensity scores to control variance, i.e., taking $\min(c, \frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})})$ as the propensity score. Setting $c$ to a small value can reduce variance, but introduces bias.
- **MinVar** & **stableVar** (Zhan et al. (2021)), **Shrinkage** (Su et al. (2020)): This line of work improves off-policy evaluation estimators by reweighing each sample. For example, MinVar and stableVar reweigh each sample by $\frac{h_{\boldsymbol{x},a}}{\sum_{a'} h_{\boldsymbol{x},a'}}$ with $h_{\boldsymbol{x},a} = \frac{\hat{\beta}(a|\boldsymbol{x})}{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})^2}$ and $h_{\boldsymbol{x},a} = \frac{\sqrt{\hat{\beta}(a|\boldsymbol{x})}}{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}$ respectively, since they find that $\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})^2/\hat{\beta}(a|\boldsymbol{x})$ is directly related to variance. Su et al. (2020) proposes to shrink the propensity scores by multiplying a weight $\lambda/(\lambda + \frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})^2}{\hat{\beta}(a|\boldsymbol{x})^2})$, which is a special case of the proposed UIPS with $U_{\boldsymbol{x},a} = 0$ and $\eta_1 = 1$. All these work simply treats $\hat{\beta}(a|\boldsymbol{x})$ as $\beta^*(a|\boldsymbol{x})$, and none of them consider the accuracy or uncertainty of $\hat{\beta}(a|\boldsymbol{x})$.
- **SNIPS** (Swaminathan & Joachims (2015c)), **BanditNet** (Joachims et al. (2018)),**POEM** (Swaminathan & Joachims (2015b)), **POXM** (Lopez et al. (2021)), **Adaptive** (Liu et al. (2022)): This line of work aims for more stable and accurate policy learning. For example, SNIPS normalizes the estimator by the sum of propensity scores in each batch. BanditNet extends SNIPS and leverages an additional Lagrangian term to normalize the estimator by an approximated sum of propensity scores of all samples. POEM jointly optimizes the estimator and its variance. POXM controls estimation variance by pruning samples with small logging probabilities. Adaptive proposes a new formulation to utilize negative samples.
- **UIPS-P** and **UIPS-O** : Two variants of our proposed UIPS with different ways of leveraging uncertainties. UIPS-P directly penalizes samples whose estimated logging probabilities are of high uncertainties, i.e., taking $\phi_{\boldsymbol{x},a} = 1.0/\exp(\gamma U_{\boldsymbol{x},a})$, which follows previous work on offline reinforcement learning (Wu et al. (2021); An et al. (2021)). UIPS-O adversarially uses the worst propensity scores $(\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})/\boldsymbol{B}^-_{\boldsymbol{x},a})$ for policy learning, i.e., $\phi_{\boldsymbol{x},a} = 1.0/\exp(-\gamma U_{\boldsymbol{x},a})$.

### 4.1 Synthetic Data

**Data generation.** Following previous work (Ma et al. (2020); Lopez et al. (2021)), we generate a synthetic dataset by a supervision-to-bandit conversion on Wiki10-31K dataset (Bhatia et al. (2016)), which is an extreme multi-label classification dataset. The Wiki10-31K dataset contains approximately 20K samples. Each sample is a associated with a feature vector $\tilde{\boldsymbol{x}}$ of 101,938 dimensions and a label vector $\boldsymbol{y}_{\tilde{\boldsymbol{x}}}$ of 31K classes with more than one positive class. Let $\boldsymbol{y}_{\tilde{\boldsymbol{x}},a}$ denote the label of class $a$ under $\tilde{\boldsymbol{x}}$ and we take each class as an action. We adopt the Wiki10-31K dataset rather than ones in the UCI machine learning repository (Swaminathan & Joachims (2015a)), since it will be much harder with such a large action space.

We then split the dataset into train, validation, test sets with size 11K:3K:6K. The test set is from the official split. Since the original feature vector $\tilde{\boldsymbol{x}}$ is too sparse, for ease of learning, we first embed it to dimension $d$ by $\boldsymbol{x} = \boldsymbol{W}\tilde{\boldsymbol{x}}$, and synthesize the ground-truth logging policy $\beta^*(a|\boldsymbol{x})$ by:

$$\beta^*(a|\boldsymbol{x}) = \frac{\exp\big(\boldsymbol{x}^T\boldsymbol{\theta}^*_a/\tau\big)}{\sum_{a'}\exp\big(\boldsymbol{x}^T\boldsymbol{\theta}^*_{a'}/\tau\big)}, \tag{10}$$

where $\boldsymbol{W}$ and $\{\boldsymbol{\theta}^*_a\}$ are pre-learned parameters by applying a logistic regression model on the train set, $\tau$ is a hyper-parameter that controls the skewness of logging distribution. A small value of $\tau$ leads to a near-deterministic policy, while a larger $\tau$ makes logging policy smoother. For each sample in train set, given $\boldsymbol{x}$, we sample an action $a$ according to $\beta^*(a|\boldsymbol{x})$, and obtain the reward $r_{\boldsymbol{x},a} = \boldsymbol{y}_{\tilde{\boldsymbol{x}},a}$, resulting bandit feedback $(\boldsymbol{x}, a, r_{\boldsymbol{x},a})$. We repeat above process $N$ times to collect the logged dataset. In our experiments, we take $d = 64, N = 100$. Due to space limit, more implementation details can be found in Appendix A.2.

**Evaluation metrics**. To evaluate the learned policy $\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})$, we calculate Precision@K (P@K), Recall@K (R@K) and NDCG@K as in previous work (Lopez et al. (2021); Ma et al. (2020)). Higher P@K, R@K and NDCG@K implies a better policy.

Table 1 shows the experiment results on three synthetic datasets generated under different $\tau$. We can first observe that as $\tau$ increases, i.e., the probability of selecting positive actions decreases, the performance of most algorithms drop, including CE, IPS-Cap, UIPS, Shrinkage, POEM, Adaptive, etc. However, UIPS still achieves the best performance on all three datasets under all three metrics. SNIPS, BanditNet , POXM are more robust to small logging probabilities of positive actions,

| | $\tau = 0.5$ | | | $\tau = 1$ | | | $\tau = 2$ | | |
|---|---|---|---|---|---|---|---|---|---|
| Algorithm | P@5 | R@5 | NDCG@5 | P@5 | R@5 | NDCG@5 | P@5 | R@5 | NDCG@5 |
| CE | 0.5559 | 0.1575 | 0.6048 | 0.5508 | 0.1561 | 0.5999 | 0.5447 | 0.1544 | 0.5933 |
| IPS-Cap | 0.5583 | 0.1576 | 0.6093 | 0.5554 | 0.1573 | 0.6028 | 0.5471 | 0.1549 | 0.5946 |
| MinVar | 0.5372 | 0.1521 | 0.5905 | 0.5342 | 0.1508 | 0.5855 | 0.5478 | 0.1551 | 0.5962 |
| StableVar | 0.4430 | 0.1269 | 0.5073 | 0.5450 | 0.1546 | 0.5959 | 0.5349 | 0.1512 | 0.5851 |
| Shrinkage | 0.5573 | 0.1577 | 0.6059 | 0.5608 | 0.1580 | 0.6122 | 0.5465 | 0.1545 | 0.5963 |
| SNIPS | 0.4287 | 0.1225 | 0.4994 | 0.4714 | 0.1327 | 0.5418 | 0.5065 | 0.1420 | 0.5748 |
| BanditNet | 0.4711 | 0.1329 | 0.5393 | 0.4811 | 0.1351 | 0.5498 | 0.4876 | 0.1363 | 0.5590 |
| POEM | 0.5541 | 0.1562 | 0.6046 | 0.5540 | 0.1565 | 0.6039 | 0.5459 | 0.1545 | 0.5946 |
| POXM | 0.4707 | 0.1328 | 0.5392 | 0.4646 | 0.1302 | 0.5380 | 0.4870 | 0.1361 | 0.5584 |
| Adaptive | 0.4203 | 0.1165 | 0.4909 | 0.4160 | 0.1154 | 0.4912 | 0.3939 | 0.1097 | 0.4704 |
| UIPS-P | 0.4864 | 0.1366 | 0.5566 | 0.4827 | 0.1354 | 0.5528 | 0.4852 | 0.1355 | 0.5566 |
| UIPS-O | 0.4871 | 0.1382 | 0.5528 | 0.4797 | 0.1348 | 0.5479 | 0.5124 | 0.1441 | 0.5796 |
| UIPS | **0.5666** | **0.1606** | **0.6169** | **0.5624** | **0.1583** | **0.6154** | **0.5497** | **0.1553** | **0.6005** |

Table 1: Experimental results on synthetic datasets.

| | **Low Frequent Action Related Samples ( High Uncertainty)** | | | **High Frequent Action Related Samples (Low Uncertainty)** | | |
|---|---|---|---|---|---|---|
| Algorithm | P@5(RI) | R@5(RI) | NDCG@5(RI) | P@5(RI) | R@5(RI) | NDCG@5(RI) |
| CE | 0.5186 | 0.1228 | 0.5521 | 0.5931 | 0.1921 | 0.6575 |
| IPS-Cap | 0.5170(-0.31%) | 0.1218(-0.81%) | 0.5539(+0.32%) | 0.5996(+1.10%) | 0.1935(+0.73%) | 0.6647(+1.10%) |
| Shrinkage | 0.5145(-0.79%) | 0.1212(-1.30%) | 0.5519(-0.04%) | 0.5982(+0.86%) | 0.1931(+0.52%) | 0.6628(+0.81%) |
| UIPS | 0.5276(+1.74%) | 0.1250(+1.79%) | 0.5623(+1.85%) | 0.6055(+2.09%) | 0.1961(+2.08%) | 0.6715(+2.13%) |

Table 2: Performance under different uncertainties.

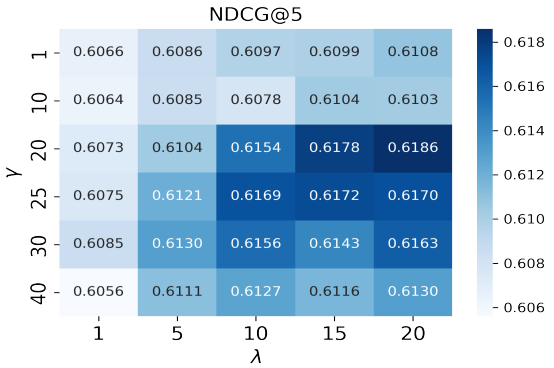



Figure 2: Effect of $\lambda$ and $\gamma$ on NDCG@5.

| Algorithm | **MSE** |
|---|---|
| IPS-CaP | 0.4953 |
| minVar | 0.8928 |
| stableVar | 0.8112 |
| Shrinage | 0.5125 |
| UIPS | 0.4516 |

Figure 3: MSE of different off-policy evaluation estimators.

since their performance increase as $\tau$ increases. Moreover, when it is harder to accurately estimate the logging probabilities, for example, the logging policy is skewed to part of positive actions (i.e., $\tau = 0.5$), or the logging policy is too smooth to sample enough positive actions (i.e., $\tau = 2$), considering the estimation uncertainty generally leads to larger improvements. For example, compared to Shrinkage (a special case of UIPS with uncertainties always being zero), UIPS achieves 1.8% (0.7%) improvements on NDCG@5 when $\tau = 0.5$ ($\tau = 2.0$), which is larger than 0.5% with $\tau = 1.0$. Finally, regardless of the scale of propensity scores, blindly reweighing through uncertainties also leads to poor performance, as shown by UIPS-P and UIPS-O.

**Performance under different uncertainty levels.** As shown in Figure 1, low frequency actions in the logged dataset suffer higher uncertainties in their propensity estimation. Thus, we divide the test set into two subsets according to the average frequency of associated actions, where the uncertainty in the subset associated with low-frequency actions is on average 9% higher than that in the subset associated with high-frequency actions. Table 2 shows the results on these two subsets when $\tau = 0.5$. We only report the results of the best three baselines due to space limit. One can clearly observe that only UIPS performed better than CE on the test set associated with low frequency actions, implying the advantage of UIPS in dealing with the inaccurately estimated logging probabilities.

**Ablation Study.** In this experiment, we aim to answer two questions: (1) Can $\hat{V}_{\text{UIPS}}(\pi_{\boldsymbol{\vartheta}})$ in Eq. (5) lead to more accurate off-policy evaluation? (2) How will UIPS perform with different hyper-parameters. Due to space limit, we report results on synthetic dataset with $\tau = 0.5$.

To answer the first question, we evaluate the following $\epsilon$-greedy policy: $\pi(a|\boldsymbol{x}) = \frac{1-\epsilon}{|M_x|} \cdot \mathbb{I}\{a \in M_x\} + \epsilon/|\mathcal{A}|$, where $M_x$ contains all positive actions associated with feature vector $\boldsymbol{x}$. Then for

| | Yahoo | | | Coat | | | KuaiRec | | |
|---|---|---|---|---|---|---|---|---|---|
| Algorithm | P@5 | R@5 | NDCG@5 | P@5 | R@5 | NDCG@5 | P@50 | R@50 | NDCG@50 |
| CE | 0.2840 | 0.7648 | 0.6153 | 0.2766 | 0.4548 | 0.4493 | 0.8769 | 0.0239 | 0.8769 |
| IPS-Cap | 0.2718 | 0.7323 | 0.5801 | 0.2841 | 0.4741 | 0.4509 | 0.8746 | 0.0238 | 0.8791 |
| MinVar | **0.2878** | 0.7783 | 0.6296 | 0.2862 | 0.4803 | 0.4456 | 0.8811 | 0.0240 | 0.8864 |
| StableVar | 0.2803 | 0.7499 | 0.5973 | 0.2834 | 0.4667 | 0.4433 | 0.8516 | 0.0231 | 0.8631 |
| Shrinkage | 0.2842 | 0.7670 | 0.6247 | 0.2828 | 0.4656 | 0.4535 | 0.8759 | 0.0238 | 0.8695 |
| SNIPS | 0.2265 | 0.5979 | 0.4512 | 0.2717 | 0.4381 | 0.4139 | 0.8460 | 0.0230 | 0.8455 |
| BanditNet | 0.2474 | 0.6632 | 0.5172 | **0.2890** | 0.4674 | 0.4298 | 0.8869 | 0.0242 | 0.8918 |
| POEM | 0.2703 | 0.7280 | 0.5770 | 0.2834 | 0.4636 | 0.4479 | 0.8768 | 0.0239 | 0.8825 |
| POXM | 0.2277 | 0.6040 | 0.4738 | 0.2703 | 0.4374 | 0.4065 | 0.9059 | 0.0248 | 0.9133 |
| Adaptive | 0.2846 | 0.7634 | 0.6048 | 0.2848 | 0.4672 | 0.4259 | 0.8382 | 0.0227 | 0.8505 |
| UIPS-P | 0.2004 | 0.5188 | 0.3714 | 0.2786 | 0.4518 | 0.4118 | 0.8811 | 0.0240 | 0.8749 |
| UIPS-O | 0.1987 | 0.5142 | 0.3705 | 0.271 | 0.4341 | 0.4205 | 0.8819 | 0.0240 | 0.8764 |
| UIPS | **0.2878** | **0.7805** | **0.6329** | **0.2890** | **0.4821** | **0.4654** | **0.9124** | **0.0250** | **0.9180** |

Table 3: Experimental results on real-world unbiased datasets.

each $\boldsymbol{x}$ in the test set, we sample 1K data points in a similar way as discussed previously to calculate the value of estimators. Table 3 shows the average MSE of the estimators to ground-truth policy value under 20 different random seeds. We only compared with baselines on off-policy evaluation estimator, i.e., IPS-Cap, MinVar, stableVar and Shrinkage. One can observe from Table 3 that UIPS does lead to the smallest MSE, implying the most accurate off-policy evaluation.

To answer the second question, we fixed $\eta_1, \eta_2$, and vary $\lambda$ and $\gamma$ to track the performance of UIPS. Recall that a larger $\gamma$ implies a higher chance the derived interval contains $\beta^*(a|\boldsymbol{x})$, while $\sqrt{\lambda}/\eta_1$ is the threshold determining the value of $\phi^*_{\boldsymbol{x},a}$ as shown in Lemma 1. Figure 2 reports NDCG@5 under diferent $\gamma$ and $\lambda$. Results on P@5 and R@5 can be found in Appendix A.2. We can observe that to make UIPS perform, $\boldsymbol{B}_{\boldsymbol{x},a}$ needs to be of high confidence, e.g., $\gamma = 25$ performed the best when $\tau = 0.5$. Moreover, the threshold $\sqrt{\lambda}/\eta_1$ cannot be too small or too large.

### 4.2 Real-World Data

Off-policy learning has been its utility in recommendation scenarios (Chen et al. (2019); Ma et al. (2020)), where context vector $\boldsymbol{x}$ denotes the state of a user and each candidate item is taken as an action. To further demonstrate the efficiency of UIPS in real-world scenarios, we evaluate it on three recommendation datasets with unbiased logging policies: (1) Yahoo!R3[1]; (2)Coat[2]; (3)KuaiRec (Gao et al. (2022)), from music, fashion and micro-video recommendation scenario respectively. All these datasets contain a set of biased data collected from users' interactions on the platform, and a set of unbiased data collected from a randomized controlled trial where items are randomly selected. The statistics of the three datasets are summarized in Table 4 in Appendix A.2. As in (Ding et al. (2022)), the biased data is used for training, and the unbiased data is for testing, with a small part of unbiased data split for validation purpose (5% on Yahoo and Coat, and 15% on KuaiRec). We take the reward as 1 if : (1) the rating is larger than 3 in Yahoo!R3 and Coat datasets; (2) the user watched more than 70% of the video in KuaiRec. Otherwise, the reward is labeled as 0. Due to space limit, one can refer to Appendix A.2 for more implementation details, e.g., model architectures, etc.

We still adopt P@K, R@K and NDCG@K as our evaluation metrics. Following (Ding et al. (2022)), we take $K = 5$ on Yahoo!R3 and Coat datasets, and $K = 50$ on KuaiRec dataset. From Table 3, we can observe that on all three datasets, the proposed UIPS achieves the highest precision, recall and NDCG. IPS-Cap cannot consistently outperform CE: it outperforms CE on Coat and KuaiRec, but fails on the Yahoo dataset. One possible reason is due to the inaccuracy of the estimated logging probabilities, since number of logged samples per user is much smaller on Yahoo dataset. For example, 70% of testing users have fewer than 24 logged interactions in Yahoo dataset, while each user in Coat has 24 recorded interactions and the number is even larger on KuaiRec dataset. BanditNet, POEM and POXM tend to perform better with a larger action space, while MinVar, StableVar and Shrinkage as well as Adaptive are more suitable for scenarios with small action size. UIPS still outperforms Shrinkage, highlighting the importance of modeling uncertainty in the estimated logging policy. However, reweighing based solely on uncertainties, ignoring the corresponding propensity scores, will also lead to poor performance, as shown by UIPS-P and UIPS-O.

[1] `https://webscope.sandbox.yahoo.com/`

[2] `https://www.cs.cornell.edu/~schnabts/mnar/`

## 5 RELATED WORK

This work is the first of its kind to take into consideration the uncertainty of estimated logging policy for improved policy learning. The following two lines of work are related to this paper.

**Off-policy learning.** In many real-world applications, such as search engines, recommender systems, etc., interactive online model update is expensive and risky (Jiang & Li (2016)). Off-policy learning has therefore attracted increasing interest, since it can leverage the already logged feedback data (Agarwal et al. (2019); Chen et al. (2019); Liu et al. (2022)). The main challenge in off-policy learning is how to address the mismatch between the logging policy and the learning policy. One line of work (Achiam et al. (2017); Schulman et al. (2015)) circumvents this by constraining the learning policy not too far from the logging policy. However, such constraint is too restrictive thus not applicable in some scenarios such as recommender systems where user behaviors and items change rapidly. Another more common and widely-applied approach is to leverage Inverse Propensity Score (IPS) method to correct the discrepancy between two policies. And various methods are proposed for stabilized learning (Swaminathan & Joachims (2015c;a;b)) and variance control (Lopez et al. (2021); Liu et al. (2022)) on top of IPS. However, all these work directly use the estimated logging policy for off-policy correction, leading to sub-optimal performance as shown in our experiments. Some other work further extend IPS-based off-policy learning for more complex problems, such as slate recommendation (Swaminathan et al. (2017)), two-stage recommender systems (Ma et al. (2020)), etc. But they still fail to realize the effect of accuracy of the estimated logging policy. A recent work (Ding et al. (2022)) on causal recommendation also argues that propensity scores may not be correct due to unobserved confounders. However, they assume the effect of the unobserved confounder for any sample can be bounded by a pre-defined hyper-parameter, and adversarially search for the worst-case propensity to update model parameters. Adapting to off-policy learning, it is a special case of our UIPS-O variant with uncertainty as a pre-defined constant.

Off-policy learning can be directly built on off-policy evaluation. In this line of research, several work (Su et al. (2020); Zhan et al. (2021)) also propose to control the high variance of learning caused by small logging probabilities by instance reweighing. However, they directly take the estimated logging policy as true logging policy for correction, thus worse than UIPS as shown in experiments. A recent work (Saito & Joachims (2022)) assumes additional structure in action space and proposes the marginalized IPS. Instead, our work considers the uncertainty when estimating the logging policy and thus does not add new assumptions about the problem space.

**Uncertainty-aware Learning.** Estimation uncertainty has been extensively used for making trade-offs between exploration and exploitation in online learning (Xu et al. (2021); Zhou et al. (2020); Abbasi-Yadkori et al. (2011)). Recently, several work on offline reinforcement learning (Wu et al. (2021); An et al. (2021); Bai et al. (2022)) penalizes the value function of out-of-distribution states and actions by directly subtracting uncertainty to tackle the extrapolating error. However, blindly penalizing samples of high uncertainty (i.e., UIPS-P) is problematic, as we showed in our experiments. Proper correction depends on both uncertainty in logging policy estimation and the actual value of estimated logging probabilities.

## 6 CONCLUSION

In this paper, we propose a novel Uncertainty-aware Inverse Propensity Score estimator (UIPS) to explicitly model the uncertainty about estimated logging policy for improved off-policy learning. UIPS weighs each logged instance to approach the ground-truth estimator and a closed-form solution of the optimal weight is derived by minimizing the upper bound of the mean squared error (MSE). An improved policy can be obtained by optimizing the resulting estimator. Extensive experiments on synthetic datasets and three real-world datasets demonstrate the efficiency of UIPS .

As demonstrated in this work, explicitly modeling the uncertainty of estimated logging policy is crucial for effective off-policy learning; but the best use of this uncertainty is not to simply down-weigh or drop instances with uncertain estimations, but to balance it with the actually estimated logging probabilities in a per-instance basis. As our future work, it is promising to investigate how UIPS can be extended to value-based learning methods, e.g., actor-critics. And on the other hand, it is also important to analyze how tight our upper bound analysis of MSE is; and if possible, find new ways to tighten it for improvements.

## REFERENCES


Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*, pp. 2312–2320, 2011.

Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *International conference on machine learning*, pp. 22–31. PMLR, 2017.

Aman Agarwal, Ivan Zaitsev, Xuanhui Wang, Cheng Li, Marc Najork, and Thorsten Joachims. Estimating position bias without intrusive interventions. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, pp. 474–482, 2019.

Gaon An, Seungyong Moon, Jang-Hyun Kim, and Hyun Oh Song. Uncertainty-based offline reinforcement learning with diversified q-ensemble. *Advances in neural information processing systems*, 34:7436–7447, 2021.

Chenjia Bai, Lingxiao Wang, Zhuoran Yang, Zhihong Deng, Animesh Garg, Peng Liu, and Zhaoran Wang. Pessimistic bootstrapping for uncertainty-driven offline reinforcement learning. *arXiv preprint arXiv:2202.11566*, 2022.

K. Bhatia, K. Dahiya, H. Jain, P. Kar, A. Mittal, Y. Prabhu, and M. Varma. The extreme classification repository: Multi-label datasets and code, 2016. URL `http://manikvarma.org/downloads/XC/XMLRepository.html`.

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

Ruidi Chen, Ioannis Ch Paschalidis, et al. Distributionally robust learning. *Foundations and Trends® in Optimization*, 4(1-2):1–243, 2020.

Sihao Ding, Peng Wu, Fuli Feng, Yitong Wang, Xiangnan He, Yong Liao, and Yongdong Zhang. Addressing unmeasured confounder for recommendation with sensitivity analysis. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 305–315, 2022.

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pp. 1050–1059. PMLR, 2016.

Chongming Gao, Shijun Li, Wenqiang Lei, Jiawei Chen, Biao Li, Peng Jiang, Xiangnan He, Jiaxin Mao, and Tat-Seng Chua. Kuairec: A fully-observed dataset and insights for evaluating recommender systems. In *Proceedings of the 31st ACM International Conference on Information and Knowledge Management*, CIKM '22, 2022. doi: 10.1145/3511808.3557220. URL `https://doi.org/10.1145/3511808.3557220`.

Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. Deepfm: a factorization-machine based neural network for ctr prediction. *arXiv preprint arXiv:1703.04247*, 2017.

Nan Jiang and Lihong Li. Doubly robust off-policy value evaluation for reinforcement learning. In *International Conference on Machine Learning*, pp. 652–661. PMLR, 2016.

Thorsten Joachims, Adith Swaminathan, and Maarten De Rijke. Deep learning with logged bandit feedback. In *International Conference on Learning Representations*, 2018.

Sergey Levine and Vladlen Koltun. Guided policy search. In *International conference on machine learning*, pp. 1–9. PMLR, 2013.

Yaxu Liu, Jui-Nan Yen, Bowen Yuan, Rundong Shi, Peng Yan, and Chih-Jen Lin. Practical counterfactual policy learning for top-k recommendations. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1141–1151, 2022.

Romain Lopez, Inderjit S Dhillon, and Michael I Jordan. Learning from extreme bandit feedback. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 8732–8740, 2021.

Jiaqi Ma, Zhe Zhao, Xinyang Yi, Ji Yang, Minmin Chen, Jiaxi Tang, Lichan Hong, and Ed H Chi. Off-policy learning in two-stage recommender systems. In *Proceedings of The Web Conference 2020*, pp. 463–473, 2020.

Rémi Munos, Tom Stepleton, Anna Harutyunyan, and Marc Bellemare. Safe and efficient off-policy reinforcement learning. *Advances in neural information processing systems*, 29, 2016.

Yuta Saito and Thorsten Joachims. Off-policy evaluation for large action spaces via embeddings. *arXiv preprint arXiv:2202.06317*, 2022.

Tobias Schnabel, Adith Swaminathan, Ashudeep Singh, Navin Chandak, and Thorsten Joachims. Recommendations as treatments: Debiasing learning and evaluation. In *international conference on machine learning*, pp. 1670–1679. PMLR, 2016.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International conference on machine learning*, pp. 1889–1897. PMLR, 2015.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484–489, 2016.

Alex Strehl, John Langford, Lihong Li, and Sham M Kakade. Learning from logged implicit exploration data. *Advances in neural information processing systems*, 23, 2010.

Yi Su, Maria Dimakopoulou, Akshay Krishnamurthy, and Miroslav Dudík. Doubly robust off-policy evaluation with shrinkage. In *International Conference on Machine Learning*, pp. 9167–9176. PMLR, 2020.

Adith Swaminathan and Thorsten Joachims. Batch learning from logged bandit feedback through counterfactual risk minimization. *The Journal of Machine Learning Research*, 16(1):1731–1755, 2015a.

Adith Swaminathan and Thorsten Joachims. Counterfactual risk minimization: Learning from logged bandit feedback. In *International Conference on Machine Learning*, pp. 814–823. PMLR, 2015b.

Adith Swaminathan and Thorsten Joachims. The self-normalized estimator for counterfactual learning. *advances in neural information processing systems*, 28, 2015c.

Adith Swaminathan, Akshay Krishnamurthy, Alekh Agarwal, Miro Dudik, John Langford, Damien Jose, and Imed Zitouni. Off-policy evaluation for slate recommendation. *Advances in Neural Information Processing Systems*, 30, 2017.

Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3):229–256, 1992.

Yue Wu, Shuangfei Zhai, Nitish Srivastava, Joshua Susskind, Jian Zhang, Ruslan Salakhutdinov, and Hanlin Goh. Uncertainty weighted actor-critic for offline reinforcement learning. *arXiv preprint arXiv:2105.08140*, 2021.

Pan Xu, Zheng Wen, Handong Zhao, and Quanquan Gu. Neural contextual bandits with deep representation and shallow exploration. In *International Conference on Learning Representations*, 2021.

Ruohan Zhan, Vitor Hadad, David A Hirshberg, and Susan Athey. Off-policy evaluation via adaptive weighting with data from contextual bandits. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pp. 2125–2135, 2021.

Dongruo Zhou, Lihong Li, and Quanquan Gu. Neural contextual bandits with ucb-based exploration. In *International Conference on Machine Learning*, pp. 11492–11502. PMLR, 2020.

Guorui Zhou, Xiaoqiang Zhu, Chenru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. Deep interest network for click-through rate prediction. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 1059–1068, 2018.

# A APPENDIX

## A.1 THEORETICAL PROOF.

**Proof of Proposition 1:**

*Proof.* With the linearity of expectation, we have $\mathbb{E}_D\left[\hat{V}_{\mathrm{BIPS}}(\pi_{\boldsymbol{\vartheta}})\right] = \mathbb{E}_{\beta^*}\left[\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})} r_{\boldsymbol{x},a}\right]$, thus:

$$\begin{aligned}
\mathrm{Bias}\left(\hat{\mathrm{V}}_{\mathrm{BIPS}}(\pi_{\boldsymbol{\vartheta}})\right) &= \mathbb{E}_D\left[\hat{V}_{\mathrm{BIPS}}(\pi_{\boldsymbol{\vartheta}}) - V(\pi_{\boldsymbol{\vartheta}})\right] \\
&= \mathbb{E}_{\beta^*}\left[\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})} r_{\boldsymbol{x},a}\right] - \mathbb{E}_{\beta^*}\left[\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\beta^*(a|\boldsymbol{x})} r_{\boldsymbol{x},a}\right] \\
&= \mathbb{E}_{\beta^*}\left[\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\beta^*(a|\boldsymbol{x})} r_{\boldsymbol{x},a}\left(\frac{\beta^*(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})} - 1\right)\right] \\
&= \mathbb{E}_{\pi_{\boldsymbol{\vartheta}}}\left[r_{\boldsymbol{x},a}\left(\frac{\beta^*(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})} - 1\right)\right]. \qquad (11)
\end{aligned}$$

For variance, since samples are independently sampled from logging policy, thus :

$$\mathrm{Var}_D\left(\hat{V}_{\mathrm{BIPS}}(\pi_{\boldsymbol{\vartheta}})\right) = \frac{1}{N}\mathrm{Var}_{\beta^*}\left(\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})} r_{\boldsymbol{x},a}\right).$$

By re-scaling, we get:

$$\begin{aligned}
N \cdot \mathrm{Var}_D\left(\hat{V}_{\mathrm{BIPS}}(\pi_{\boldsymbol{\vartheta}})\right) &= \mathrm{Var}_{\beta^*}\left(\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})} r_{\boldsymbol{x},a}\right) \\
&= \mathbb{E}_{\beta^*}\left[\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})^2}{\hat{\beta}(a|\boldsymbol{x})^2} r_{\boldsymbol{x},a}^2\right] - \left(\mathbb{E}_{\beta^*}\left[\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})} r_{\boldsymbol{x},a}\right]\right)^2 \\
&= \mathbb{E}_{\pi_{\boldsymbol{\vartheta}}}\left[\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\beta^*(a|\boldsymbol{x})} \cdot \frac{\beta^*(a|\boldsymbol{x})^2}{\hat{\beta}(a|\boldsymbol{x})^2} r_{\boldsymbol{x},a}^2\right] - \left(\mathbb{E}_{\pi_{\boldsymbol{\vartheta}}}\left[\frac{\beta^*(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})} r_{\boldsymbol{x},a}\right]\right)^2 \\
&= \mathrm{Var}_{\pi_{\boldsymbol{\vartheta}}}\left(\frac{\beta^*(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})} r_{\boldsymbol{x},a}\right) + \mathbb{E}_{\pi_{\boldsymbol{\vartheta}}}\left[\left(\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\beta^*(a|\boldsymbol{x})} - 1\right) \cdot \frac{\beta^*(a|\boldsymbol{x})^2}{\hat{\beta}(a|\boldsymbol{x})^2} r_{\boldsymbol{x},a}^2\right]. \qquad (12)
\end{aligned}$$

Then we complete the proof. □

**Proof of Theorem 1:**

*Proof.* We can get:

$$\begin{aligned}
\mathrm{MSE}\left(\hat{V}_{\mathrm{UIPS}}(\pi_{\boldsymbol{\vartheta}})\right) &= \mathbb{E}_D\left[\left(\hat{V}_{\mathrm{UIPS}}(\pi_{\boldsymbol{\vartheta}}) - V(\pi_{\boldsymbol{\vartheta}})\right)^2\right] \\
&= \left(\mathbb{E}_D\left[\hat{V}_{\mathrm{UIPS}}(\pi_{\boldsymbol{\vartheta}}) - V(\pi_{\boldsymbol{\vartheta}})\right]\right)^2 + \mathrm{Var}_D\left(\hat{V}_{\mathrm{UIPS}}(\pi_{\boldsymbol{\vartheta}}) - V(\pi_{\boldsymbol{\vartheta}})\right) \\
&= \left(\mathbb{E}_D\left[\hat{V}_{\mathrm{UIPS}}(\pi_{\boldsymbol{\vartheta}}) - V(\pi_{\boldsymbol{\vartheta}})\right]\right)^2 + \mathrm{Var}_D\left(\hat{V}_{\mathrm{UIPS}}(\pi_{\boldsymbol{\vartheta}})\right) \\
&= \mathrm{Bias}(\hat{V}_{\mathrm{UIPS}}(\pi_{\boldsymbol{\vartheta}}))^2 + \mathrm{Var}(\hat{V}_{\mathrm{UIPS}}(\pi_{\boldsymbol{\vartheta}})).
\end{aligned}$$

We first bound the bias term:

$$\begin{aligned}
\mathrm{Bias}(\hat{V}_{\mathrm{UIPS}}(\pi_{\boldsymbol{\vartheta}})) &= \mathbb{E}_D\left[\hat{V}_{\mathrm{UIPS}}(\pi_{\boldsymbol{\vartheta}}) - V(\pi_{\boldsymbol{\vartheta}})\right] \\
&= \mathbb{E}_{\beta^*}\left[\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})} \phi_{\boldsymbol{x},a} r_{\boldsymbol{x},a}\right] - V(\pi_{\boldsymbol{\vartheta}}) \qquad (1) \\
&= \mathbb{E}_{\beta^*}\left[\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})} \phi_{\boldsymbol{x},a} r_{\boldsymbol{x},a} - \frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\beta^*(a|\boldsymbol{x})} r_{\boldsymbol{x},a}\right] \\
&= \mathbb{E}_{\beta^*}\left[r_{\boldsymbol{x},a} \frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\beta^*(a|\boldsymbol{x})} \cdot \left(\frac{\beta^*(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})} \phi_{\boldsymbol{x},a} - 1\right)\right] \\
&\leq \sqrt{\mathbb{E}_{\pi_{\boldsymbol{\vartheta}}}\left[r_{\boldsymbol{x},a}^2 \frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\beta^*(a|\boldsymbol{x})}\right]} \cdot \sqrt{\mathbb{E}_{\beta^*}\left[\left(\frac{\beta^*(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})} \phi_{\boldsymbol{x},a} - 1\right)^2\right]} \qquad (2)
\end{aligned}$$

Equality (1) follows the linearity of expectation. Inequality (2) is due to the Cauchy-Schwarz inequality. We then bound the variance term:

$$\mathrm{Var}(\hat{V}_{\mathrm{UIPS}}(\pi_{\boldsymbol{\vartheta}})) = \frac{1}{N}\mathrm{Var}_{\beta^*}\left(\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})}\phi_{\boldsymbol{x},a}r_{\boldsymbol{x},a}\right) \tag{1}$$

$$= \frac{1}{N}\left(\mathbb{E}_{\beta^*}\left[\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})^2}{\hat{\beta}(a|\boldsymbol{x})^2}\phi^2_{\boldsymbol{x},a}r^2_{\boldsymbol{x},a}\right] - \left(\mathbb{E}_{\beta^*}\left[\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})}\phi_{\boldsymbol{x},a}r_{\boldsymbol{x},a}\right]\right)^2\right) \tag{13}$$

$$\leq \frac{1}{N}\mathbb{E}_{\beta^*}\left[\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})^2}{\hat{\beta}(a|\boldsymbol{x})^2}\phi^2_{\boldsymbol{x},a}r^2_{\boldsymbol{x},a}\right] \leq \mathbb{E}_{\beta^*}\left[\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})^2}{\hat{\beta}(a|\boldsymbol{x})^2}\phi^2_{\boldsymbol{x},a}\right] \tag{14}$$

Combining the bound of bias and variance, we can complete the proof. □

**Proof of Theorem 2:**

*Proof.* We first define several notations:

- $T(\phi_{\boldsymbol{x},a}, \beta_{\boldsymbol{x},a}) = \lambda\mathbb{E}_{\beta^*}\left[\left(\frac{\beta_{\boldsymbol{x},a}}{\hat{\beta}(a|\boldsymbol{x})}\phi_{\boldsymbol{x},a} - 1\right)^2\right] + \mathbb{E}_{\beta^*}\left[\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})^2}{\hat{\beta}(a|\boldsymbol{x})^2}\phi^2_{\boldsymbol{x},a}\right]$.
- $\tilde{T}(\phi_{\boldsymbol{x},a}) = \max_{\beta_{\boldsymbol{x},a}\in\boldsymbol{B}_{\boldsymbol{x},a}} T(\phi_{\boldsymbol{x},a}, \beta_{\boldsymbol{x},a})$ denotes the maximum value of inner problem.
- $T^* = \min_{\phi_{\boldsymbol{x},a}} \tilde{T}(\phi_{\boldsymbol{x},a}) = \min_{\phi_{\boldsymbol{x},a}} \max_{\beta_{\boldsymbol{x},a}\in\boldsymbol{B}_{\boldsymbol{x},a}} T(\phi_{\boldsymbol{x},a}, \beta_{\boldsymbol{x},a})$ denote the optimal min-max value. And $\phi^*_{\boldsymbol{x},a} = \arg\min_{\phi_{\boldsymbol{x},a}} \tilde{T}(\phi_{\boldsymbol{x},a})$ .
- $\boldsymbol{B}^-_{\boldsymbol{x},a} := \frac{\hat{Z}\exp(-\gamma U_{\boldsymbol{x},a})}{Z^*}\hat{\beta}(a|\boldsymbol{x})$, and $\boldsymbol{B}^+_{\boldsymbol{x},a} := \frac{\hat{Z}\exp(\gamma U_{\boldsymbol{x},a})}{Z^*}\hat{\beta}(a|\boldsymbol{x})$.

We first find the maximum value of inner problem, i.e., $\tilde{T}(\phi_{\boldsymbol{x},a})$ for any fixed $\phi_{\boldsymbol{x},a}$. And there are three cases shown in Figure 4:

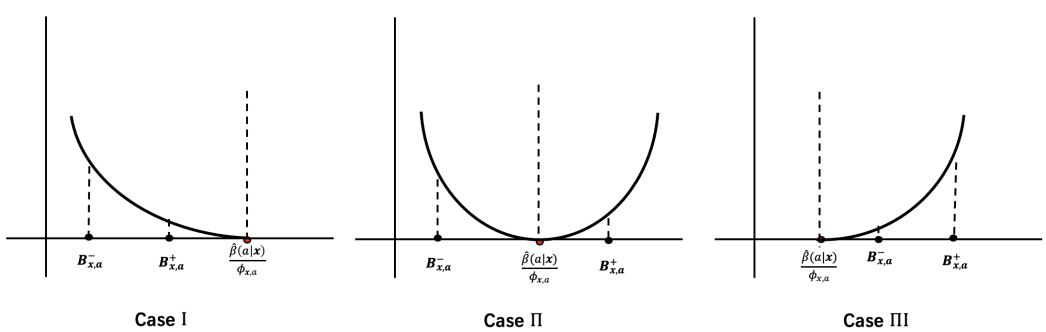



Figure 4: Three cases for maximizing inner problem.

**Case I:** When $\frac{\hat{\beta}(a|\boldsymbol{x})}{\phi_{\boldsymbol{x},a}} \geq \boldsymbol{B}^+_{\boldsymbol{x},a}$, , $\tilde{T}(\phi_{\boldsymbol{x},a})$ achieves the maximum value at $\beta_{\boldsymbol{x},a} = \boldsymbol{B}^-_{\boldsymbol{x},a}$. In other words, $\tilde{T}(\phi_{\boldsymbol{x},a}) = T(\phi_{\boldsymbol{x},a}, \boldsymbol{B}^-_{\boldsymbol{x},a})$ when $\phi_{\boldsymbol{x},a} \leq \frac{\hat{\beta}(a|\boldsymbol{x})}{\boldsymbol{B}^+_{\boldsymbol{x},a}}$.

**Case II:** When $\boldsymbol{B}^-_{\boldsymbol{x},a} \leq \frac{\hat{\beta}(a|\boldsymbol{x})}{\phi_{\boldsymbol{x},a}} \leq \boldsymbol{B}^+_{\boldsymbol{x},a}$, i.e., $\frac{Z^*\exp(-\gamma U_{\boldsymbol{x},a})}{\hat{Z}} \leq \phi_{\boldsymbol{x},a} \leq \frac{Z^*\exp(\gamma U_{\boldsymbol{x},a})}{\hat{Z}}$, then $\tilde{T}(\phi_{\boldsymbol{x},a})$ will be the maximum between $T(\phi_{\boldsymbol{x},a}, \boldsymbol{B}^-_{\boldsymbol{x},a})$ and $T(\phi_{\boldsymbol{x},a}, \boldsymbol{B}^+_{\boldsymbol{x},a})$.

More specifically, when $\frac{\hat{\beta}(a|\boldsymbol{x})}{\phi_{\boldsymbol{x},a}} \leq \frac{\boldsymbol{B}^+_{\boldsymbol{x},a}+\boldsymbol{B}^-_{\boldsymbol{x},a}}{2}$, i.e., $\phi_{\boldsymbol{x},a} \geq \frac{2\hat{\beta}(a|\boldsymbol{x})}{\boldsymbol{B}^+_{\boldsymbol{x},a}+\boldsymbol{B}^-_{\boldsymbol{x},a}}$, $\tilde{T}(\phi_{\boldsymbol{x},a}) = T(\phi_{\boldsymbol{x},a}, \boldsymbol{B}^+_{\boldsymbol{x},a})$. Otherwise when $\phi_{\boldsymbol{x},a} < \frac{2\hat{\beta}(a|\boldsymbol{x})}{\boldsymbol{B}^+_{\boldsymbol{x},a}+\boldsymbol{B}^-_{\boldsymbol{x},a}}$, $\tilde{T}(\phi_{\boldsymbol{x},a}) = T(\phi_{\boldsymbol{x},a}, \boldsymbol{B}^-_{\boldsymbol{x},a})$.

**Case III:** When $\phi_{\boldsymbol{x},a} \geq \frac{\hat{\beta}(a|\boldsymbol{x})}{\boldsymbol{B}^-_{\boldsymbol{x},a}}$. implying $\frac{\hat{\beta}(a|\boldsymbol{x})}{\phi_{\boldsymbol{x},a}} \leq \boldsymbol{B}^-_{\boldsymbol{x},a}$, $\tilde{T}(\phi_{\boldsymbol{x},a}) = T(\phi_{\boldsymbol{x},a}, \boldsymbol{B}^+_{\boldsymbol{x},a})$.

Overall, we get that:

$$\tilde{T}(\phi_{\boldsymbol{x},a}) = \begin{cases} T(\phi_{\boldsymbol{x},a}, \boldsymbol{B}^-_{\boldsymbol{x},a}), & \phi_{\boldsymbol{x},a} \in (-\infty, \frac{2\hat{\beta}(a|\boldsymbol{x})}{\boldsymbol{B}^+_{\boldsymbol{x},a}+\boldsymbol{B}^-_{\boldsymbol{x},a}}] \\ T(\phi_{\boldsymbol{x},a}, \boldsymbol{B}^+_{\boldsymbol{x},a}) & \phi_{\boldsymbol{x},a} \in [\frac{2\hat{\beta}(a|\boldsymbol{x})}{\boldsymbol{B}^+_{\boldsymbol{x},a}+\boldsymbol{B}^-_{\boldsymbol{x},a}}, \infty) \end{cases} \tag{15}$$

Next we try to find the minimum value of $\tilde{T}(\phi_{\boldsymbol{x},a})$. We first observe that without considering constraint on $\phi_{\boldsymbol{x},a}$, when

$$\phi^+_{\boldsymbol{x},a} = \frac{\lambda}{\lambda \frac{\hat{Z}\exp(\gamma U_{\boldsymbol{x},a})}{Z^*} + \frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})^2}{\hat{\beta}(a|\boldsymbol{x})^2 \frac{\hat{Z}\exp(\gamma U_{\boldsymbol{x},a})}{Z^*}}},$$

$T(\phi_{\boldsymbol{x},a}, \boldsymbol{B}^+_{\boldsymbol{x},a})$ achieves the global minimum value. However, $\phi^+_{\boldsymbol{x},a} \leq \frac{\hat{\beta}(a|\boldsymbol{x})}{\boldsymbol{B}^+_{\boldsymbol{x},a}}$, which implies when $\phi_{\boldsymbol{x},a} \in [\frac{2\hat{\beta}(a|\boldsymbol{x})}{\boldsymbol{B}^+_{\boldsymbol{x},a}+\boldsymbol{B}^-_{\boldsymbol{x},a}}, \infty)$, the minimum value of $T(\phi_{\boldsymbol{x},a}, \boldsymbol{B}^+_{\boldsymbol{x},a})$ achieves at $\frac{2\hat{\beta}(a|\boldsymbol{x})}{\boldsymbol{B}^+_{\boldsymbol{x},a}+\boldsymbol{B}^-_{\boldsymbol{x},a}}$.

On the other hand, without considering any constraint on $\phi_{\boldsymbol{x},a}$ , global minimum value of $T(\phi_{\boldsymbol{x},a}, \boldsymbol{B}^-_{\boldsymbol{x},a})$ achieves at:

$$\phi^-_{\boldsymbol{x},a} = \frac{\lambda}{\lambda \frac{\hat{Z}\exp(-\gamma U_{\boldsymbol{x},a})}{Z^*} + \frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})^2}{\hat{\beta}(a|\boldsymbol{x})^2 \frac{\hat{Z}\exp(-\gamma U_{\boldsymbol{x},a})}{Z^*}}}. \tag{16}$$

Thus if $\phi^-_{\boldsymbol{x},a} \leq \frac{2\hat{\beta}(a|\boldsymbol{x})}{\boldsymbol{B}^+_{\boldsymbol{x},a}+\boldsymbol{B}^-_{\boldsymbol{x},a}}$, $\phi^*_{\boldsymbol{x},a} = \phi^-_{\boldsymbol{x},a}$. Otherwise $\phi^*_{\boldsymbol{x},a} = \frac{2\hat{\beta}(a|\boldsymbol{x})}{\boldsymbol{B}^+_{\boldsymbol{x},a}+\boldsymbol{B}^-_{\boldsymbol{x},a}}$.

Overall,

$$\phi^*_{\boldsymbol{x},a} = \min\left( \frac{\lambda}{\lambda \frac{\hat{Z}\exp(-\gamma U_{\boldsymbol{x},a})}{Z^*} + \frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})^2}{\hat{\beta}(a|\boldsymbol{x})^2 \frac{\hat{Z}\exp(-\gamma U_{\boldsymbol{x},a})}{Z^*}}}, \frac{2}{\frac{\hat{Z}\exp(\gamma U_{\boldsymbol{x},a})}{Z^*} + \frac{\hat{Z}\exp(-\gamma U_{\boldsymbol{x},a})}{Z^*}} \right) \tag{17}$$

We use $\eta_1$ and $\eta_2$ to represent $\frac{Z^*}{\hat{Z}}$ in the two terms respectively. We can get

$$\phi^*_{\boldsymbol{x},a} = \min\left( \frac{\lambda}{\frac{\lambda}{\eta_1}\exp(-\gamma U_{\boldsymbol{x},a}) + \frac{\eta_1 \pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})^2}{\hat{\beta}(a|\boldsymbol{x})^2 \exp(-\gamma U_{\boldsymbol{x},a})}}, \frac{2\eta_2}{\exp(\gamma U_{\boldsymbol{x},a}) + \exp(-\gamma U_{\boldsymbol{x},a})} \right) \tag{18}$$

where $\eta_1, \eta_2 \in [\exp(-\gamma U_s^{\max}), \exp(\gamma U_s^{\max})]$, since $\hat{Z}\exp(-\gamma U_s^{\max}) \leq Z^* = \sum_{a'} \exp(f_{\theta^*}(a'|\boldsymbol{x})) \leq \hat{Z}\exp(U_s^{\max})$. Usually we set $\eta_1 \leq \eta_2$. We introduce two parameters since the scale of $\eta_1$ is closely related to the scale of $\lambda$, while the scale of $\eta_2$ is independent.

Then we complete the proof . □

**Lemma 1.** *Assume $\eta_1 \leq \eta_2$, then with fixed $\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})$ and $\hat{\beta}(a|\boldsymbol{x})$, and $\alpha_{\boldsymbol{x},a} = \sqrt{\frac{\lambda}{2\eta_1\eta_2} - \frac{\lambda(1-\eta_1)}{\eta_1^2}\exp(-2\gamma U_{\boldsymbol{x},a})}$, we have the following observations:*

- *If $\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})} \leq \alpha_{\boldsymbol{x},a}$, $\phi^*_{\boldsymbol{x},a} = 2\eta_2 / [\exp(\gamma U_{\boldsymbol{x},a}) + \exp(-\gamma U_{\boldsymbol{x},a})]$. Otherwise $\phi^*_{\boldsymbol{x},a} = \lambda / \left[\frac{\lambda}{\eta_1}\exp(-\gamma U_{\boldsymbol{x},a}) + \frac{\eta_1 \pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})^2}{\hat{\beta}^2(a|\boldsymbol{x})\exp(-\gamma U_{\boldsymbol{x},a})}]\right]$. In other words, $\phi^*_{\boldsymbol{x},a} \leq 2\eta_2$ always holds.*
- *If $\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})} \geq \frac{\sqrt{\lambda}}{\eta_1}$ , then $\phi^*_{s,a}$ will always decrease as $U_{\boldsymbol{x},a}$ increases.*
- *If $\alpha_{\boldsymbol{x},a} \leq \frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})} < \frac{\sqrt{\lambda}}{\eta_1}\exp(-\gamma U_{\boldsymbol{x},a})$, larger $U_{\boldsymbol{x},a}$ brings larger $\phi^*_{\boldsymbol{x},a}$. Otherwise $\phi^*_{\boldsymbol{x},a}$ still decreases as $U_{\boldsymbol{x},a}$ increases.*

*Proof.* For the first observation, deriving

$$\frac{\lambda}{\frac{\lambda}{\eta_1}\exp(-\gamma U_{\boldsymbol{x},a}) + \frac{\eta_1 \pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})^2}{\hat{\beta}^2(a|\boldsymbol{x})\exp(-\gamma U_{\boldsymbol{x},a})}} \leq \frac{2\eta_2}{\exp(\gamma U_{\boldsymbol{x},a}) + \exp(-\gamma U_{\boldsymbol{x},a})}$$

we can get the result.

For the second and third observation, since $\eta_1 \leq \eta_2$, then $\alpha_{\boldsymbol{x},a} \leq \frac{\sqrt{\lambda}}{\sqrt{2}\eta_1} \leq \frac{\sqrt{\lambda}}{\eta_1}$. Let $\mathcal{L}(u) = \frac{\lambda}{\eta_1}\exp(-\gamma u) + \frac{\eta_1 \pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})^2}{\hat{\beta}(a|\boldsymbol{x})^2 \exp(-\gamma u)}$, we can have:

$$\nabla_u \mathcal{L}(u) = -\gamma \frac{\lambda}{\eta_1}\exp(-\gamma u) + \gamma \frac{\eta_1 \pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})^2}{\hat{\beta}(a|\boldsymbol{x})^2}\exp(\gamma u)$$

By letting $\nabla_u \mathcal{L}(u) \geq 0$, we can get $u \geq \frac{1}{\gamma}\log\left(\frac{\sqrt{\lambda}\hat{\beta}(a|\boldsymbol{x})}{\eta_1 \pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}\right)$. This implies when $U_{\boldsymbol{x},a} \geq \frac{1}{\gamma}\log\left(\frac{\sqrt{\lambda}\hat{\beta}(a|\boldsymbol{x})}{\eta_1 \pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}\right)$, $\phi^*_{\boldsymbol{x},a}$ will decrease as $U_{\boldsymbol{x},a}$ increases. Otherwise as $U_{\boldsymbol{x},a}$ increases, $\phi^*_{\boldsymbol{x},a}$ also increases.

Specially, when $\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})} \geq \frac{\sqrt{\lambda}}{\eta_1}$, $\log\left(\frac{\sqrt{\lambda}\hat{\beta}(a|\boldsymbol{x})}{\eta_1 \pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}\right) \leq 0$. Given $U_{\boldsymbol{x},a} \geq 0$, this implies that $\phi^*_{\boldsymbol{x},a}$ will always decrease as $U_{\boldsymbol{x},a}$ increases in this case.

Otherwise, when $\alpha_{\boldsymbol{x},a} \leq \frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})} \leq \frac{\sqrt{\lambda}}{\eta_1}\exp(-\gamma U_{\boldsymbol{x},a})$, with

$$\frac{\pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}{\hat{\beta}(a|\boldsymbol{x})\exp(-\gamma U_{\boldsymbol{x},a})} \leq \frac{\sqrt{\lambda}}{\eta_1} \Longrightarrow U_{\boldsymbol{x},a} \leq \frac{1}{\gamma}\log\left(\frac{\sqrt{\lambda}\hat{\beta}(a|\boldsymbol{x})}{\eta_1 \pi_{\boldsymbol{\vartheta}}(a|\boldsymbol{x})}\right)$$

.

larger $U_{\boldsymbol{x},a}$ implies larger $\phi_{\boldsymbol{x},a}$. Otherwise $\phi^*_{\boldsymbol{x},a}$ still decreases as $U_{\boldsymbol{x},a}$ increases. This completes the proof. □

### A.2 EXPERIMENTAL DETAILS.

#### A.2.1 SYNTHETIC DATA

**Implementation Details.** We model the logging policy as in Equation (7) with $f_{\boldsymbol{\theta}}(\boldsymbol{x}, a) = \boldsymbol{x}^T \boldsymbol{\theta}_a$, where $\{\boldsymbol{\theta}_a\}$ are parameters to learn. To train the logging policy, we take all samples in the logged dataset $D$ as positive instances, and randomly sample non-selected actions as negative instances as in (Chen et al. (2019)). We used grid search to select the hyperparameters based on the model's performance on validation dataset: the learning rate was searched in $\{1e^{-5}, 1e^{-4}, 1e^{-3}, 1e^{-2}\}$; $\lambda, \gamma, \eta_1$ were searched in {1, 2,5, 10, 15, 20, 25, 30, 40, 50}. And $\eta_2$ was searched in {1, 10, 100, 1000}. For baseline algorithms, we performed a similar grid search as mentioned above, and the search range follows the original papers.

**Ablation Study.** Effect of hyperparameter $\gamma$ and $\lambda$ on precision and recall can be found in Figure 5a and Figure 5b respectively.

#### A.2.2 REAL-WORLD DATA

**Statistics of data.** The statistics of three real-world recommendation datasets with unbiased data can be found in Table 4.

| Dataset | #User | #Item | #Biased Data | #Unbiased Data |
|---|---|---|---|---|
| Yahoo | 15,400 | 1,000 | 311,704 | 54,000 |
| Coat | 290 | 300 | 6,960 | 4,640 |
| Product | 7,176 | 10,729 | 12,530,806 | 4,676,570 |

Table 4: The statistics of three real-world datasets.

**Implementation Details.** We adopt a two-tower neural network architecture to implement both the logging and learning policy, as shown in Figure 6. For the learning policy, the user representation and the item representation are first modelled through two separate neural networks (i.e., the user tower and the item tower), and then their element-by-element product vector is projected to predict the user's preference for the item. We then re-use the user state generated from the user tower of

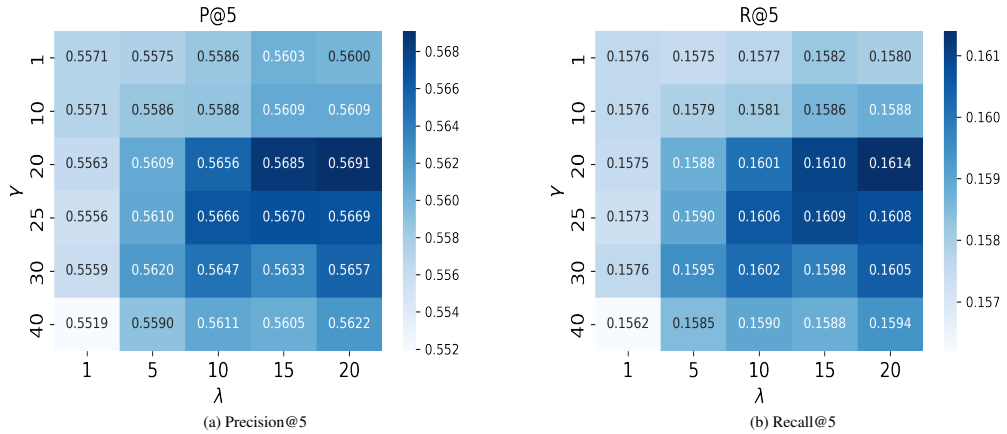



Figure 5: Effect of $\lambda$ and $\gamma$ on Precision@5 and Recall@5.

the learning policy, and model the logging policy with another separate item tower, following (Chen et al. (2019)). We also block gradients to prevent the logging policy interfering the user state of the learning policy. In each learning epoch, we will first estimate the logging policy, and then take the estimated logging probabilities as well as their uncertainties to optimize the learning policy. All hyperparameters are searched in a similar way described in Section 4.1.

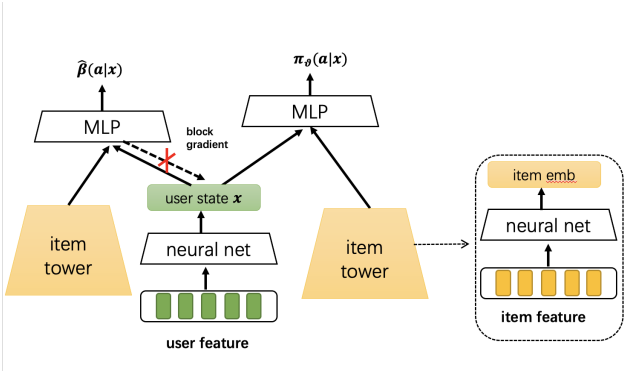



Figure 6: Model architecture of the logging and the learning policy in real-world datasets