OFF-POLICY LEARNING IN LARGE ACTION SPACES: OPTIMIZATION MATTERS MORE THAN ESTIMATION

Imad Aouali*
Criteo AI Lab
CREST, ENSAE, IP Paris
i.aouali@criteo.com

Otmane Sakhi*
Criteo AI Lab
o.sakhi@criteo.com

ABSTRACT

Off-policy evaluation (OPE) and off-policy learning (OPL) are foundational for decision-making in offline contextual bandits. Recent advances in OPL primarily optimize OPE estimators with improved statistical properties, assuming that better estimators inherently yield superior policies. Although theoretically justified, this estimator-centric approach neglects a critical practical obstacle: challenging optimization landscapes. In this paper, we provide theoretical insights and empirical evidence showing that current OPL methods encounter severe optimization issues, particularly as the action space grows. We show that estimator-aware policy parametrization can mitigate, but not fully resolve, optimization challenges. Building on this, we explore simpler weighted log-likelihood objectives and demonstrate that they enjoy substantially better optimization properties and still recover competitive, often superior, learned policies. Our findings emphasize the necessity of explicitly addressing optimization considerations in the development of OPL algorithms for large action spaces.

1 INTRODUCTION

The offline contextual bandit (Dudík et al., 2011) leverages logged data from past interactions to improve future decision-making, with wide applications in areas like recommendation systems (Bottou et al., 2013; Aouali et al., 2022). We consider a standard setting where we are given a dataset $\mathcal{D}_n = \{(X_i, A_i, R_i)\}_{i=1}^n$ of n i.i.d. tuples. Each tuple consists of a context $X_i \in \mathcal{X} \subseteq \mathbb{R}^d$ drawn from an unknown distribution ν, an action $A_i \in \mathcal{A} = [K]$ sampled from a known logging policy as $A_i \sim \pi_0(\cdot \mid X_i)$, and a corresponding reward $R_i \sim p(\cdot \mid X_i, A_i)$ sampled from the unknown reward distribution $p(\cdot \mid X_i, A_i)$, whose mean is $r(x, a) = \mathbb{E}_{R \sim p(\cdot|x,a)}[R]$. The performance of any new policy π is measured by its *value*, defined as the expected reward it would obtain:

$$V(\pi) = \mathbb{E}_{X \sim \nu, A \sim \pi(\cdot|X)}[r(X, A)]. \tag{1}$$

The goal of *off-policy learning (OPL)* is to leverage the offline dataset $\mathcal{D}_n$ to learn a policy $\hat{\pi}_n$ from a policy class Π that maximizes this value, i.e., $\hat{\pi}_n = \arg\max_{\pi \in \Pi} V(\pi)$.

The dominant paradigm in OPL is to optimize an *off-policy evaluation (OPE)* estimator $\hat{V}_n(\pi)$ that approximates the true policy value $V(\pi)$ (Swaminathan & Joachims, 2015a) such as $\hat{V}_n(\pi) \approx V(\pi)$. The learning problem is thus framed as $\hat{\pi}_n = \operatorname{argmax}_\pi \hat{V}_n(\pi)$, with the rationale that maximizing a more accurate estimate of the value yields a superior learned policy. However, this estimator-centric view overlooks a critical aspect: the optimization landscape. OPE objectives (Dudík et al., 2011; Dudík et al., 2012; Dudik et al., 2014; Wang et al., 2017; Farajtabar et al., 2018; Su et al., 2020; Metelli et al., 2021; Kuzborskij et al., 2021; Saito & Joachims, 2022) are highly non-concave with common parameterized policies (Chen et al., 2019), prone to suboptimal local maxima, an issue more pronounced in large action spaces. Notably, even sophisticated estimators designed to reduce variance fail to overcome this optimization barrier, suffering from difficult-to-optimize landscapes.

We show that one way to alleviate these difficulties is through *estimator-aware policy parametrization*: structuring the policy class to match the implicit biases of the estimator. Such parametrizations

*Equal contribution.

reduce the effective search space and can shorten optimization plateaus. While this strategy provides tangible benefits, it does not eliminate the fundamental non-concavity of OPE objectives, leaving optimization as the central bottleneck.

Motivated by this limitation, our work advocates an alternative approach based on *policy-weighted log-likelihood (PWLL)* objectives. Unlike traditional estimators, PWLL optimizes an objective $\hat{U}_n(\pi)$ designed for ease of optimization rather than accuracy in estimating $V(\pi)$. Although PWLL objectives perform poorly as value estimators, their favorable concave landscape significantly enhances their effectiveness for learning. Through theoretical and empirical analysis, we show that this optimization-centric approach consistently enables simpler PWLL objectives to outperform complex, state-of-the-art OPE objectives, particularly in large action spaces.

The remainder is organized as follows. Section 2 uses an asymptotic lens to analyze OPE objectives, then derives objective-aware policy parametrizations that partially alleviate optimization challenges faced by OPE objectives. Section 3 introduces PWLL objectives and establishes their favorable optimization properties. Section 4 presents large-scale experiments. We conclude in Section 5.

2 Analysis of OPE objectives

OPE objectives optimize an estimator $\hat{V}_n(\pi)$ of the value $V(\pi)$. While statistically motivated, this induces objective-specific biases in the learned policy and yields challenging optimization landscapes. To make this explicit, we study the *asymptotic solutions* $\pi_*^{\text{METHOD}} = \operatorname{argmax}_\pi \lim_{n\to\infty} \hat{V}_n^{\text{METHOD}}(\pi)$ obtained by maximizing each estimator in the infinite-data regime (proofs in Appendix C).

Why is the infinite-data view informative? In practice, the logging policy π_0 typically concentrates on a small, context-dependent support (Sachdeva et al., 2020) that is much smaller than K:

$$S_0(x) = \{a : \pi_0(a \mid x) > 0\}, \qquad k_0(x) = |S_0(x)| \ll K\,, \tag{2}$$

and we denote $k_0 = \mathbb{E}[k_0(X)]$ as the typical support size. For a fixed policy π, standard concentration results (e.g., (Sakhi et al., 2024)) give pointwise deviations of order $\sqrt{k_0/n}$ between $\hat{V}_n(\pi)$ and its expectation.[1] Hence, when n is large and k_0 is small (common in practice), the estimator is close to its value with infinite data (i.e., its expectation). Therefore, looking at the maximizer of the estimator in the infinite-data regime (i.e., asymptotic solution) is insightful. In addition, these asymptotic solutions could be obtained in closed-form, while the finite-data ones cannot. This is why we focus on the infinite-data view.

What does the infinite-data view reveal? Taking $n \to \infty$ removes sampling fluctuations and isolates the inductive bias of the objective. Different estimators converge to different policy structures even with infinite data. Thus, π_*^{METHOD} is governed by the estimator's design rather than by statistical noise. This perspective clarifies the policies each objective targets and motivates *objective-aware parametrizations*, aligning the policy class with the induced bias, *to ease optimization*. This forms the first line of improvements we propose in this paper.

2.1 Standard OPE objectives

Inverse propensity scoring (`IPS`). The foundational `IPS` estimator (Horvitz & Thompson, 1952) re-weights observed rewards by the ratio between the target policy π and the logging policy π_0:

$$\hat{V}_n^{\texttt{IPS}}(\pi) = \frac{1}{n}\sum_{i=1}^{n} \frac{\pi(A_i \mid X_i)}{\pi_0(A_i \mid X_i)} R_i. \tag{3}$$

With infinite data, `IPS` selects the best-rewarding action among those in the support of π_0:

$$\pi_*^{\texttt{IPS}}(a \mid x) = \mathbb{1}\left[a = \operatorname*{argmax}_{a' \in \mathcal{A}} r(x, a')\, \mathbb{1}[\pi_0(a' \mid x) > 0]\right]. \tag{4}$$

`IPS` is unbiased but can suffer from high variance due to large importance weight values.

[1] This can be extended to hold uniformly over a parametric class of dimension d (e.g., linear softmax), via standard Rademacher/VC arguments.

Clipped IPS (`cIPS`). To mitigate the high variance of `IPS`, `cIPS` (Bottou et al., 2013) clips small propensity scores at a threshold $\tau \in (0, 1)$:

$$\hat{V}_n^{\texttt{cIPS}}(\pi) = \frac{1}{n}\sum_{i=1}^{n} \frac{\pi(A_i \mid X_i)}{\max\{\pi_0(A_i \mid X_i), \tau\}} R_i. \tag{5}$$

This clipping introduces a bias. The asymptotic policy down-weights the rewards of rare actions, causing it to favor actions that were frequent under π_0, even if they are suboptimal:

$$\pi_*^{\texttt{cIPS}}(a \mid x) = \mathbb{1}\Big[a = \operatorname*{argmax}_{a' \in \mathcal{A}} \frac{\pi_0(a' \mid x)}{\max\{\pi_0(a' \mid x), \tau\}} r(x, a')\Big]. \tag{6}$$

Exponential smoothing (`ES`). Instead of hard clipping, `ES` (Aouali et al., 2023) smooths importance weights by raising propensities to a fractional power $\alpha \in (0, 1)$:

$$\hat{V}_n^{\texttt{ES}}(\pi) = \frac{1}{n}\sum_{i=1}^{n} \frac{\pi(A_i \mid X_i)}{\pi_0(A_i \mid X_i)^{\alpha}} R_i. \tag{7}$$

Its asymptotic policy balances reward maximization with preference for frequent actions:

$$\pi_*^{\texttt{ES}}(a \mid x) = \mathbb{1}\left[a = \operatorname*{argmax}_{a' \in \mathcal{A}} r(x, a')\pi_0(a' \mid x)^{1-\alpha}\right]. \tag{8}$$

Another variant of `ES` regularizes the entire importance weight as $(\frac{\pi}{\pi_0})^{\alpha}$ instead of only the denominator. In contrast to the deterministic policies derived from `IPS`, `cIPS`, and the `ES` formulation above, this approach yields a stochastic asymptotic policy: $\pi_*^{\texttt{ES}}(a \mid x) \propto r(x, a)^{1/(1-\alpha)}\pi_0(a \mid x)$. Other regularizations include logarithmic smoothing (Sakhi et al., 2024), implicit exploration (Gabbianelli et al., 2024), harmonic correction (Metelli et al., 2021), shrinkage (Su et al., 2020). Further details are available in Appendix B.

Doubly robust (`DR`). The `DR` estimator incorporates a reward model $\hat{r}(x, a)$ to reduce variance and enable generalization to actions outside π_0's support. A common clipped variant is:

$$\hat{V}_n^{\texttt{DR}}(\pi) = \frac{1}{n}\sum_{i=1}^{n} \frac{\pi(A_i \mid X_i)}{\max\{\pi_0(A_i \mid X_i), \tau\}} \left(R_i - \hat{r}(X_i, A_i)\right) + \mathbb{E}_{A \sim \pi(\cdot \mid X_i)}\left[\hat{r}(X_i, A)\right]. \tag{9}$$

Its solution interpolates between model-based reward prediction and unbiased correction using π_0:

$$\pi_*^{\texttt{DR}}(a \mid x) = \mathbb{1}\Big[a = \operatorname*{argmax}_{a' \in \mathcal{A}} \hat{r}(x, a') + \frac{\pi_0(a' \mid x)}{\max\{\pi_0(a' \mid x), \tau\}} \left(r(x, a') - \hat{r}(x, a')\right)\Big]. \tag{10}$$

2.2 Large-scale OPE objectives

In large action spaces, importance weights $\frac{\pi(a|x)}{\pi_0(a|x)}$ can become huge, leading to estimators with high variance. To mitigate this, modern methods compute marginalized importance weights over a lower-dimensional action representation, trading bias for reduced variance.

Marginalized IPS (`MIPS`). `MIPS` (Saito & Joachims, 2022) tackles large action spaces by clustering actions. It maps each action a to a cluster c via a function $\phi : \mathcal{A} \to \mathcal{C}$, where $|\mathcal{C}| \ll |\mathcal{A}|$. Estimation is then performed at the cluster level:

$$\hat{V}_n^{\texttt{MIPS}}(\pi) = \frac{1}{n}\sum_{i=1}^{n} \frac{\pi(C_i \mid X_i)}{\pi_0(C_i \mid X_i)} R_i, \quad \text{where } C_i = \phi(A_i) \text{ and } \pi(c \mid x) = \sum_{a \in c} \pi(a \mid x). \tag{11}$$

This cluster-level marginalization introduces bias: the asymptotic solution only selects the best *cluster* based on its average reward under π_0, and cannot differentiate between actions within that cluster:

$$\pi_*^{\texttt{MIPS}}(c \mid x) = \mathbb{I}\left[c = \operatorname*{argmax}_{c' \in \mathcal{C}} \left\{\frac{\sum_{a \in c'} \pi_0(a \mid x) r(x, a)}{\sum_{a \in c'} \pi_0(a \mid x)}\right\}\right]. \tag{12}$$

Hence, MIPS offers no specific guidance for selecting an action within the optimal cluster; any action is considered equally valid. Consequently, if actions are chosen uniformly at random from this optimal cluster, the resulting asymptotic action-level solution is:

$$\pi_*^{\texttt{MIPS}}(a \mid x) = \frac{\mathbb{I}\left[\phi(a) = \operatorname{argmax}_{c' \in \mathcal{C}} \left\{ \frac{\sum_{a \in c'} \pi_0(a|x) r(x,a)}{\sum_{a \in c'} \pi_0(a|x)} \right\}\right]}{|\phi(a)|}.$$

where $|\phi(a)|$ denotes the size of the cluster containing action a.

Conjunct effect modeling (OffCEM). Building on MIPS, OffCEM (Saito et al., 2023) uses a reward model $\hat{r}$ to correct for the cluster-level aggregation bias, in a doubly robust fashion:

$$\hat{V}_n^{\texttt{OffCEM}}(\pi) = \frac{1}{n} \sum_{i=1}^{n} \left(\frac{\pi(C_i \mid X_i)}{\pi_0(C_i \mid X_i)} \left(R_i - \hat{r}(X_i, A_i)\right) + \mathbb{E}_{A \sim \pi(\cdot|X_i)}[\hat{r}(X_i, A)] \right). \tag{13}$$

The resulting asymptotic policy selects the action that maximizes the model-predicted reward $\hat{r}$, plus a cluster-level correction term that accounts for model error:

$$\pi_*^{\texttt{OffCEM}}(a \mid x) = \mathbb{I}\left[a = \operatorname*{argmax}_{a' \in \mathcal{A}} \left\{ \hat{r}(x, a') + \frac{\sum_{\bar{a} \in \phi(a')} \pi_0(\bar{a} \mid x)(r(x, \bar{a}) - \hat{r}(x, \bar{a}))}{\sum_{\bar{a} \in \phi(a')} \pi_0(\bar{a} \mid x)} \right\} \right]. \tag{14}$$

Two-stage decomposition (POTEC). In this work, we see POTEC (Saito et al., 2025) as an *optimization strategy of* OffCEM (rather than seeing it as a new estimator). It restricts the policy to a cluster-informed form,

$$\pi(a \mid x) = \sum_{c \in \mathcal{C}} \pi^{\text{RM}}(a \mid x, c) \pi^{\text{CL}}(c \mid x),$$

where $\pi^{\text{RM}}(a \mid x, c) = \mathbb{1}[a = \operatorname{argmax}_{a' \in c} \hat{r}(x, a')]$ is fixed, model-based policy that deterministically selects the best action within each cluster. Learning is then simplified to finding the optimal cluster-level policy π^{CL} that maximizes the OffCEM objective in Eq. (13):

$$\hat{V}_n^{\texttt{POTEC}}(\pi^{\text{CL}}) = \frac{1}{n} \sum_{i=1}^{n} \left(\frac{\pi^{\text{CL}}(C_i \mid X_i)}{\pi_0(C_i \mid X_i)} \left(R_i - \hat{r}(X_i, A_i)\right) + \sum_{c \in \mathcal{C}} \pi^{\text{CL}}(c \mid X_i) \hat{r}_c^*(X_i) \right), \tag{15}$$

where $\hat{r}_c^*(x) = \max_{a \in c} \hat{r}(x, a)$ is the estimated reward of the best action in cluster c. This practical decomposition has the same optimal asymptotic solution as OffCEM: $\pi_*^{\texttt{POTEC}} = \pi_*^{\texttt{OffCEM}}$.

Policy convolution (PC). Moving beyond hard clustering, PC (Sachdeva et al., 2023) leverages the assumption that actions close in an embedding space yield similar rewards. For each action a, it aggregates over its neighborhood of nearest neighbors $N_\epsilon(a) = \{a' : d(a, a') < \epsilon\}$, where d is a pre-defined distance metric (e.g., ℓ_2 distance between embeddings):

$$\hat{V}_n^{\texttt{PC}}(\pi) = \frac{1}{n} \sum_{i=1}^{n} \frac{\pi(N_\epsilon(A_i) \mid X_i)}{\pi_0(N_\epsilon(A_i) \mid X_i)} R_i, \quad \text{with } \pi(N_\epsilon(a) \mid x) = \sum_{a' \in N_\epsilon(a)} \pi(a' \mid x). \tag{16}$$

The induced asymptotic policy is deterministic: it selects the action a' that maximizes an aggregated neighborhood score. Each logged neighbor $\bar{a} \in N_\epsilon(a')$ contributes its reward $r(x, \bar{a})$, weighted by the conditional probability of observing $\bar{a}$ under the logging policy restricted to its neighborhood.

$$\pi_*^{\texttt{PC}}(a \mid x) = \mathbb{I}\left[a = \operatorname*{argmax}_{a' \in \mathcal{A}} \left\{ \sum_{\bar{a} \in N_\epsilon(a')} \frac{\pi_0(\bar{a} \mid x) r(x, \bar{a})}{\pi_0(N_\epsilon(\bar{a}) \mid x)} \right\} \right]. \tag{17}$$

Other recent IPS variants for large action spaces (Peng et al., 2023; Cief et al., 2024; Taufiq et al., 2024) are often extensions of MIPS that relax its core assumptions. We focused on four methods (MIPS, OffCEM, POTEC, and PC), which we consider representative of this family. Since these variants largely share the same MIPS foundation and optimization procedure (with the notable exception of POTEC), we expect our findings to be generally applicable.

2.3 OPTIMIZATION CHALLENGES

The statistical properties of OPE estimators are only half the story. In practice, their effectiveness is often limited by a more immediate obstacle: a challenging optimization landscape. OPE objectives become very difficult to optimize when paired with standard, expressive policy classes like the softmax. This section explores why this happens and introduces *objective-aware parametrization* as a strategy to mitigate, though not entirely solve, the problem.

To analyze the optimization process, we consider policies parametrized by a softmax function. For any given context x, the policy is defined over an *effective action space*, $\mathcal{A}_{\text{EFF}}(x) \subseteq \mathcal{A}$, which is the set of actions that can be assigned non-zero probability. The policy takes the form:

$$\pi_\theta(a|x) = \frac{\exp(s_\theta(x,a))}{\sum_{a' \in \mathcal{A}_{\text{EFF}}(x)} \exp(s_\theta(x,a'))} \quad \text{for } a \in \mathcal{A}_{\text{EFF}}(x)\,, \tag{18}$$

where $s_\theta(x,a)$ is a learnable score function. A common choice are linear softmax scores,

$$\text{lightweight: } s_\theta(x,a) = h(x,a)^\top \theta\,, \qquad \text{heavyweight: } s_\theta(x,a) = h(x)^\top \theta_a\,, \tag{19}$$

and we respectively call them, *lightweight parametrization* (learning a single shared parameter vector θ) and *heavyweight parametrization* (learning separate parameters θ_a for each action).

The size of this effective action space, $K_{\text{EFF}}(x) = |\mathcal{A}_{\text{EFF}}(x)|$, is the critical factor governing optimization. The following propositions (proofs in Appendix D, adapted from (Chen et al., 2019; Mei et al., 2020a)) reveal just how severe the problem can be.

First, gradient-based methods can get stuck in suboptimal regions for extended periods.

Proposition 2.1 (Optimization plateaus)**.** *For any OPE estimator $\hat{V}_n$ that is linear in π, and even with a linear softmax policy, there exist problems where gradient descent is trapped in a suboptimal region for a number of iterations that scales linearly with the effective action space size, K_{EFF}.*

Second, the optimization landscape is highly non-concave, plagued by poor local optima that can trap the learning algorithm.

Proposition 2.2 (Local maxima)**.** *Under similar conditions, the optimization landscape for OPE objectives can have a number of local maxima that is exponential in K_{EFF}.*

These results highlight that K_{EFF} plays a central role in the optimization properties of OPL. A common implementation choice is to set the effective action space to the full action space for all contexts, i.e., $\mathcal{A}_{\text{EFF}}(x) = \mathcal{A}$, which makes $K_{\text{EFF}}(x) = K$. In large-scale settings where K can be in the millions, this is a recipe for optimization failure, as learning must navigate a landscape with potentially long plateaus and an exponentially large number of local maxima.

Surprisingly, even sophisticated methods designed specifically for large action spaces often fall into this trap. At first glance, methods like `MIPS`, `OffCEM`, and `PC` appear to operate in a smaller space because their objective functions involve marginalized probabilities like $\pi(C_i \mid X_i)$ in `MIPS` and `OffCEM` or $\pi(N_\epsilon(A_i) \mid X_i)$ in `PC`. However, these marginalized terms are defined as sums over an underlying action-level policy:

$$\pi(C_i \mid X_i) = \sum_{a \in C_i} \pi(a \mid X_i)\,, \qquad \text{and} \qquad \pi(N_\epsilon(a) \mid x) = \sum_{a' \in N_\epsilon(a)} \pi(a' \mid x)\,.$$

If this foundational policy, $\pi(a \mid x)$, is parameterized as a standard softmax over the entire action space $\mathcal{A}$, then the optimization procedure must still compute gradients with respect to the scores of all K actions. The learning problem does not shrink; the complexity is merely hidden inside the definition of the cluster-level probabilities.

The only exception among these is `POTEC`, which we view not just as an estimator but as a deliberate *optimization strategy*. By fixing the intra-cluster policy π^{RM} and learning only the cluster-level policy π^{CL}, `POTEC` fundamentally changes the problem. It forces the optimization to occur directly in the cluster space, making its effective action space $\mathcal{C}$ and its cardinality $|\mathcal{C}| \ll K$. This structural choice circumvents the bottleneck entirely, easing the optimization landscape.

2.3.1 DESIGN IMPLICATIONS: OBJECTIVE-AWARE PARAMETRIZATION

The choice of K_{EFF} introduces a fundamental trade-off. A smaller effective action space leads to a simpler optimization landscape, it also risks excluding the optimal action and reduces the expressiveness of the policy class. If $\mathcal{A}_{\text{EFF}}(x)$ is chosen arbitrarily and is too restrictive, it may lead to bad performance. The challenge is to find the *sweet spot*: a parametrization that is constrained enough to be optimizable, yet expressive enough to contain the best possible policy for a given objective.

This is precisely where our asymptotic analysis helps. The asymptotic solution π_*^{METHOD} for each estimator reveals the minimal sufficient set of actions required to find the objective's maximizer. By aligning the policy parametrization with this insight, we can shrink the search space without sacrificing performance. This is the core principle of *objective-aware parametrization*.

For instance, the asymptotic solutions for `IPS`, `cIPS`, `ES` are always confined to the support of the logging policy, $S_0(x)$. This tells us that setting $\mathcal{A}_{\text{EFF}}(x) = S_0(x)$ is a sufficient parametrization. Similarly, for `OffCEM` and `MIPS`, the cluster-level structure of their asymptotic solutions suggests that a two-stage decomposition like `POTEC` is the most effective parametrization. These observations allow us to identify sufficient policy parametrizations that reduce the degrees of freedom and make learning more tractable. Based on this, we make the following claims, which we validate empirically in Section 4:

Claim 2.3. *For importance-weighted objectives (e.g., `IPS`, `cIPS`, `ES`), parameterizing the policy π with support restricted to that of the logging policy, $S_0(x)$, improves optimization and leads to superior learned policies.*

Claim 2.4. *For large-scale objectives (e.g., `OffCEM`), using a two-stage decomposition (`POTEC`-style) that optimizes a policy at the cluster level yields better optimization and final performance than a naive action-level parametrization.*

While this objective-aware parametrization makes the optimization landscape of OPE objectives more navigable by shrinking the search space, it only treats the symptoms (plateaus and local maxima) without curing the underlying non-concavity. Thus, we propose next a more fundamental shift: abandoning accurate value estimation in favor of tractable optimization.

3 ANALYSIS OF PWLL OBJECTIVES

To overcome the optimization challenges of OPE objectives, we consider policy-weighted log-likelihood (PWLL) objectives. These methods trade accurate value estimation for a well-behaved, concave optimization landscape, leading to more robust and effective policy learning.

General form. Given a positive weighting function $g(r, p_0)$, the PWLL objective is:

$$\hat{U}_n^g(\pi) = \frac{1}{n}\sum_{i=1}^{n} g(R_i, \pi_0(A_i \mid X_i)) \log \pi(A_i \mid X_i). \tag{20}$$

Unlike OPE objectives, this form is logarithmic in the policy π. This change has a profound impact on the optimization properties as we show in the next proposition (proof in Appendix D.).

Proposition 3.1. *For linear softmax policies π_θ, the PWLL objective $\hat{U}_n^g(\pi_\theta)$ is concave, and is strongly concave once ℓ_2 regularization is used.*

This property guarantees that a unique global maximum exists and can be found efficiently with gradient-based methods, avoiding the issues of local maxima and plateaus that plague OPE objectives. Different choices of the weighting function g yield different learning algorithms.

Local policy improvement (`LPI`). Liang & Vlassis (2022) set $g(r, p_0) = r$, which optimizes the log-likelihood of actions weighted by their observed rewards:

$$\hat{U}_n^{\texttt{LPI}}(\pi) = \frac{1}{n}\sum_{i=1}^{n} R_i \log \pi(A_i \mid X_i)\,. \tag{21}$$

The asymptotic policy balances reward-seeking with imitation of the logging policy:

$$\pi_*^{\texttt{LPI}}(a \mid x) \propto r(x, a)\pi_0(a \mid x)\,. \tag{22}$$

Clipped LPI (`cLPI`). To reduce the logging policy's influence, `cLPI` uses importance-weight clipping, setting $g(r, p_0) = \frac{r}{\max(p_0, \tau)}$:

$$\hat{U}_n^{\texttt{cLPI}}(\pi) = \frac{1}{n}\sum_{i=1}^{n} \frac{R_i}{\max\{\pi_0(A_i \mid X_i), \tau\}} \log \pi(A_i \mid X_i)\,. \tag{23}$$

In a similar spirit to `cIPS`, its asymptotic solution corrects for action frequency under π_0, down-weighting the influence of rare actions due to the clipping:

$$\pi_*^{\texttt{cLPI}}(a \mid x) \propto r(x, a) \frac{\pi_0(a \mid x)}{\max\{\pi_0(a \mid x), \tau\}}. \tag{24}$$

KL regularization (`RegKL`). To further amplify the reward signal relative to the logging policy prior, `RegKL` uses an exponential weighting function $g(r, p_0) = \exp(r/\beta)$:

$$\hat{U}_n^{\texttt{RegKL}}(\pi) = \frac{1}{n}\sum_{i=1}^{n} \exp(R_i/\beta) \log \pi(A_i \mid X_i)\,. \tag{25}$$

The asymptotic policy is proportional to the logging policy, weighted by the exponentiated reward:

$$\pi_*^{\texttt{RegKL}}(a \mid x) \propto \mathbb{E}_{r \sim p(\cdot \mid x, a)}\big[\exp(r/\beta)\big]\pi_0(a \mid x). \tag{26}$$

The temperature parameter β smoothly interpolates between behavior cloning ($\beta \to \infty$) and greedy reward maximization ($\beta \to 0$).

Note that `BPR` (Rendle et al., 2012) can be seen as an approximate PWLL objective, and we included it in our experiments. In fact, this general form of PWLL lends itself to numerous variations by modifying the weighting function g. For instance, one could introduce variants inspired by regularized IPS like Exponential Smoothing (`esLPI`) to fine-tune the bias-variance trade-off. While many such variants can be proposed for specific use cases, the central message of our work is that the well-behaved optimization landscape of the PWLL family is of greater practical importance than the estimation accuracy of OPE objectives. Thus, an exploration of these PWLL variants is beyond our scope. We contend that the foundational methods analyzed above, `LPI`, `cLPI`, and `RegKL`, along with the widely used `BPR` are sufficient to demonstrate the inherent advantages of PWLL objectives.

4 Empirical Analysis

We conduct our empirical evaluation on three large-scale recommendation datasets: `MovieLens` ($K = 60$k) (Lam & Herlocker, 2016), `Twitch` ($K = 200$k) (Rappaz et al., 2021), and `GoodReads` ($K = 1$M) (Wan et al., 2019). These benchmarks feature action spaces with up to one million items, representing some of the largest settings studied in the offline policy learning literature. For all experiments, we employ the common softmax inner-product policies. We compare methods from both objective families. For OPE objectives, we include `IPS`, `ES`, `DR`, `MIPS`, `OffCEM`, `POTEC`, and `PC` in Section 2. For PWLL objectives, we evaluate `LPI`, `cLPI`, `RegKL`, and `BPR` in Section 3. All implementation details are provided in Appendix E.

4.1 Optimization is the Main Bottleneck

To test our central hypothesis that *optimization challenges are a more significant barrier than estimation accuracy*, we evaluate how OPL objectives perform under various optimization configurations. If an algorithm's success is highly dependent on specific hyperparameters like batch size or learning rate, it suggests a difficult, non-robust optimization landscape. This experiment directly probes the practical trainability of each method, a key aspect our paper argues is often overlooked.

The results strongly support our claim. As shown in Fig. 1, *OPE objectives are highly sensitive* to batch size and learning rate schedule: minor changes can cause performance collapse, making them difficult to tune and train reliably. In contrast, *PWLL objectives remain robust*, achieving consistently high reward across all configurations. This stability translates directly into better learned policies: *PWLL objectives outperform OPE objectives on all datasets*. Even `POTEC`, a state-of-the-art method designed for large action spaces, is surpassed by the much simpler and easier-to-optimize `cLPI`. This supports our central claim: *optimization stability is key to effective OPL.*

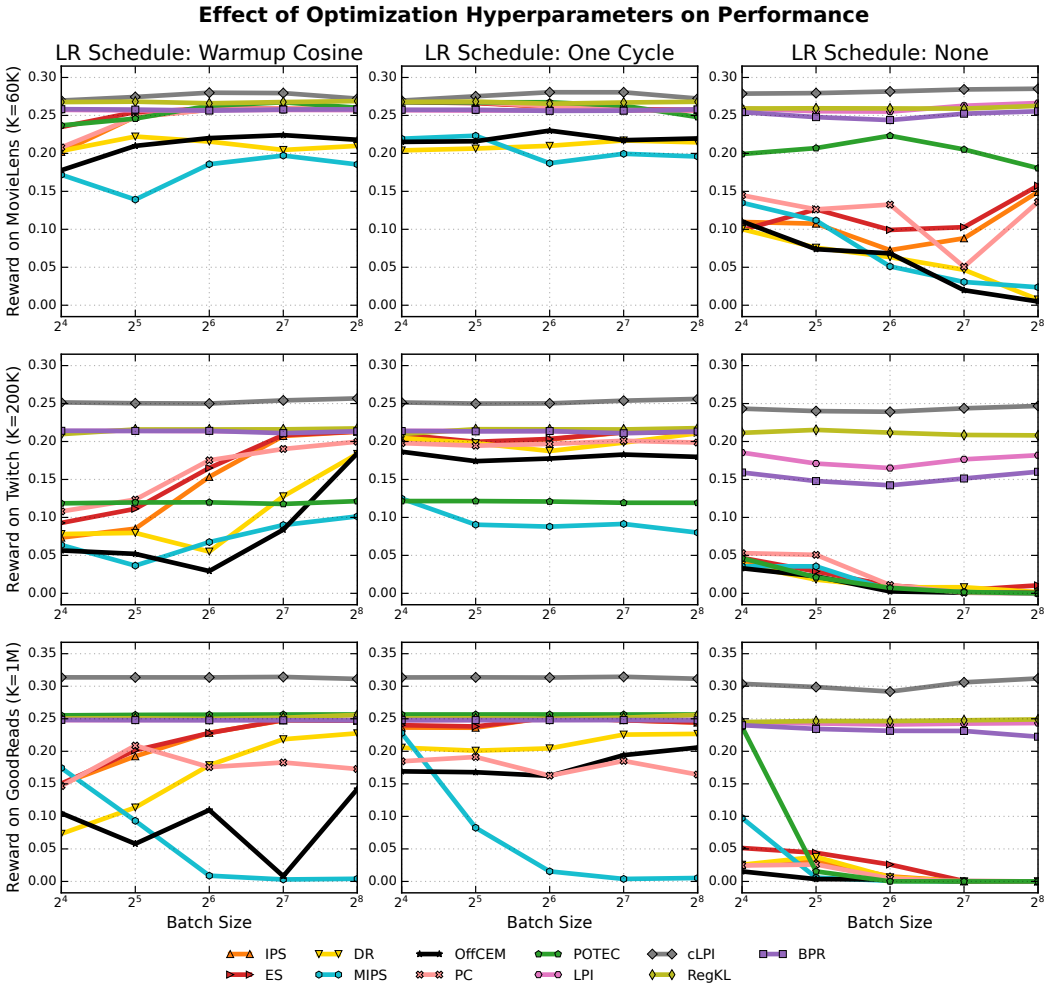

Figure 1: Effect of batch size and learning rate schedule on final validation reward using three large-scale datasets. OPE objectives are highly sensitive, while PWLL objectives are robust.

One might assume that an objective designed for estimation fidelity, such as a low-MSE OPE estimator, would naturally yield a better policy. Our findings show this is not the case. The superiority of PWLL objectives, which are poor value estimators by design, provides compelling evidence against this estimator-centric view. This reinforces our main takeaway: in large-scale OPL, a tractable optimization landscape is a more critical feature for a learning objective than its statistical accuracy. For completeness, an experiment tracking the MSE of methods is given in Appendix E.

The figure also supports Claim 2.4. Indeed, there is a consistent performance gap between `POTEC` and `OffCEM`. Both methods are designed to maximize the same asymptotic objective as we show in Section 2; their statistical goals are identical. The divergence in performance, therefore, can be attributed entirely to their differing optimization strategies. `POTEC`'s use of a two-stage, cluster-level optimization proves far more effective than `OffCEM`'s naive, action-level parametrization.

4.2 Objective-Aware Parametrization

To empirically validate Claim 2.3, we compare a naive, whole-action-space parametrization against our proposed objective-aware approach, which restricts the policy's effective action space to the logging policy support, $S_0(x)$. As shown for the `IPS` objective in Fig. 2, the naive approach is highly unstable, with performance collapsing under simple learning configurations. In contrast, the objective-aware version is very robust, achieving high reward consistently across all batch sizes and schedules. This benefit extends even to inherently stable PWLL objectives like `cLPI`, which achieve even better performance with the restricted support. This provides strong evidence for Claim 2.3:

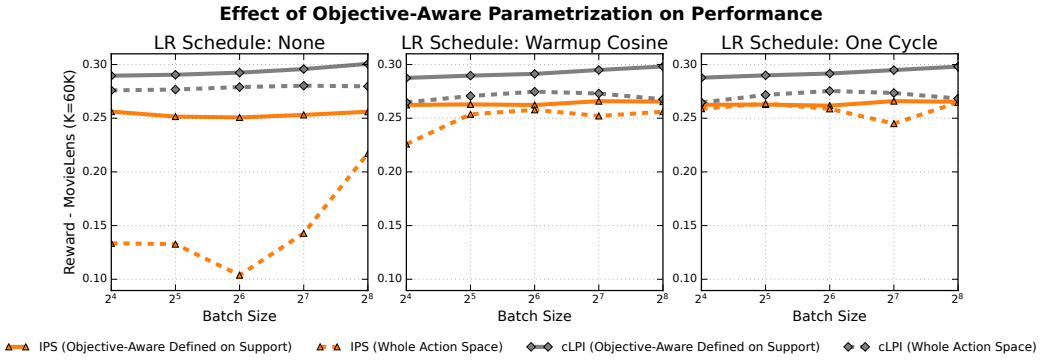

Figure 2: The effect of objective-aware parametrization for `IPS` and `cLPI` on `MovieLens`.

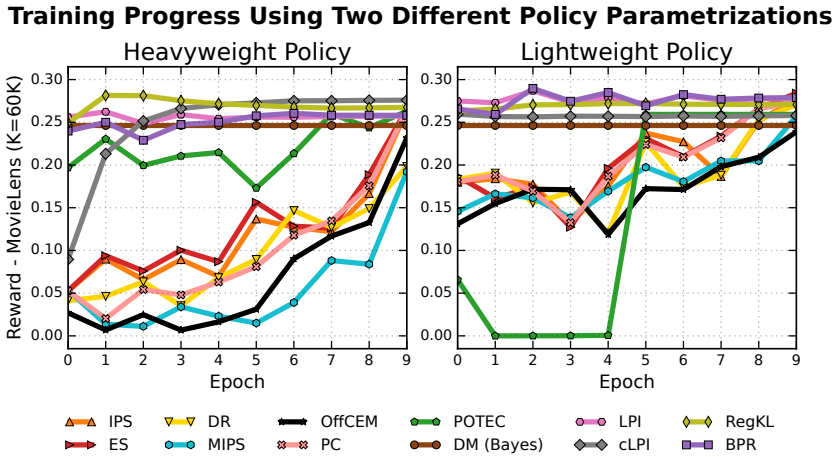

Figure 3: Training progress over 10 epochs on `MovieLens`, comparing heavyweight vs. lightweight policies.

aligning the policy structure with the objective's inductive bias simplifies the optimization landscape, leading to greater stability and superior learned policies. This finding holds across all datasets, with full results available in Appendix E.

4.3 Lightweight Policy Parametrization Helps

To further isolate the impact of policy complexity on optimization, we compare lightweight and heavyweight policy parametrizations (defined in Eq. (19)) for each objective. As shown for the `MovieLens` dataset in Fig. 3, the benefits of a simpler model are clear: lightweight policies converge faster and often achieve slightly higher final rewards. This finding reinforces a key theme of our work: a more tractable optimization landscape can be more critical to achieving strong performance than the greater expressive capacity of a complex policy. Results for other datasets are deferred to Appendix E.

5 Conclusion

The common approach to OPL, which focuses on optimizing increasingly sophisticated OPE estimators, neglects a crucial factor: the optimization landscape. We show, both theoretically and empirically, that for large action spaces, this landscape becomes challenging, affecting the practical effectiveness of these methods. Our study of this landscape motivates clever policy parameterizations and PWLL objectives. By exploiting the estimator's inductive biases, these approaches alleviate optimization challenges and induce strong concavity for common policy classes. Our experiments confirm that this focus on optimization pays off: these simple changes make learning more robust, easier to tune, and converge to superior policies. This work advocates for a shift in

focus for OPL research in large-scale settings, from estimator design towards the development of objectives with favorable optimization properties.

REFERENCES

Imad Aouali, Amine Benhalloum, Martin Bompaire, Achraf Ait Sidi Hammou, Sergey Ivanov, Benjamin Heymann, David Rohde, Otmane Sakhi, Flavian Vasile, and Maxime Vono. Reward optimizing recommendation using deep learning and fast maximum inner product search. In *proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*, pp. 4772–4773, 2022.

Imad Aouali, Victor-Emmanuel Brunel, David Rohde, and Anna Korba. Exponential smoothing for off-policy learning. In *International Conference on Machine Learning*, pp. 984–1017. PMLR, 2023.

Imad Aouali, Victor-Emmanuel Brunel, David Rohde, and Anna Korba. Unified pac-bayesian study of pessimism for offline policy learning with regularized importance sampling. In *Proceedings of the Fortieth Conference on Uncertainty in Artificial Intelligence*, pp. 88–109. PMLR, 2024.

Imad Aouali, Victor-Emmanuel Brunel, David Rohde, and Anna Korba. Bayesian off-policy evaluation and learning for large action spaces. In *Proceedings of The 28th International Conference on Artificial Intelligence and Statistics*, pp. 136–144. PMLR, 2025.

Léon Bottou, Jonas Peters, Joaquin Quiñonero-Candela, Denis X Charles, D Max Chickering, Elon Portugaly, Dipankar Ray, Patrice Simard, and Ed Snelson. Counterfactual reasoning and learning systems: The example of computational advertising. *Journal of Machine Learning Research*, 14 (11), 2013.

Minmin Chen, Ramki Gummadi, Chris Harris, and Dale Schuurmans. Surrogate objectives for batch policy optimization in one-step decision making. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Matej Cief, Jacek Golebiowski, Philipp Schmidt, Ziawasch Abedjan, and Artur Bekasov. Learning action embeddings for off-policy evaluation. In *European Conference on Information Retrieval*, pp. 108–122. Springer, 2024.

Miroslav Dudík, John Langford, and Lihong Li. Doubly robust policy evaluation and learning. *International Conference on Machine Learning*, 2011.

Miroslav Dudík, Dumitru Erhan, John Langford, and Lihong Li. Sample-efficient nonstationary policy evaluation for contextual bandits. In *Proceedings of the Twenty-Eighth Conference on Uncertainty in Artificial Intelligence*, UAI'12, pp. 247–254, Arlington, Virginia, USA, 2012. AUAI Press.

Miroslav Dudik, Dumitru Erhan, John Langford, and Lihong Li. Doubly robust policy evaluation and optimization. *Statistical Science*, 29(4):485–511, 2014.

Mehrdad Farajtabar, Yinlam Chow, and Mohammad Ghavamzadeh. More robust doubly robust off-policy evaluation. In *International Conference on Machine Learning*, pp. 1447–1456. PMLR, 2018.

Germano Gabbianelli, Gergely Neu, and Matteo Papini. Importance-weighted offline learning done right. In *International Conference on Algorithmic Learning Theory*, pp. 614–634. PMLR, 2024.

Daniel G Horvitz and Donovan J Thompson. A generalization of sampling without replacement from a finite universe. *Journal of the American statistical Association*, 47(260):663–685, 1952.

Edward L Ionides. Truncated importance sampling. *Journal of Computational and Graphical Statistics*, 17(2):295–311, 2008.

Olivier Jeunen and Bart Goethals. Pessimistic reward models for off-policy learning in recommendation. In *Fifteenth ACM Conference on Recommender Systems*, pp. 63–74, 2021.

Ilja Kuzborskij, Claire Vernade, Andras Gyorgy, and Csaba Szepesvári. Confident off-policy evaluation and selection through self-normalized importance weighting. In *International Conference on Artificial Intelligence and Statistics*, pp. 640–648. PMLR, 2021.

Shyong Lam and Jon Herlocker. MovieLens Dataset. http://grouplens.org/datasets/movielens/, 2016.

Tor Lattimore and Csaba Szepesvari. *Bandit Algorithms*. Cambridge University Press, 2019.

Dawen Liang and Nikos Vlassis. Local policy improvement for recommender systems. *arXiv preprint arXiv:2212.11431*, 2022.

Ben London and Ted Sandler. Bayesian counterfactual risk minimization. In *International Conference on Machine Learning*, pp. 4125–4133. PMLR, 2019.

Jincheng Mei, Chenjun Xiao, Bo Dai, Lihong Li, Csaba Szepesvari, and Dale Schuurmans. Escaping the gravitational pull of softmax. In *Advances in Neural Information Processing Systems*, volume 33, pp. 21130–21140. Curran Associates, Inc., 2020a.

Jincheng Mei, Chenjun Xiao, Csaba Szepesvári, and Dale Schuurmans. On the global convergence rates of softmax policy gradient methods. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org, 2020b.

Alberto Maria Metelli, Alessio Russo, and Marcello Restelli. Subgaussian and differentiable importance sampling for off-policy evaluation and learning. *Advances in Neural Information Processing Systems*, 34:8119–8132, 2021.

Jie Peng, Hao Zou, Jiashuo Liu, Shaoming Li, Yibao Jiang, Jian Pei, and Peng Cui. Offline policy evaluation in large action spaces via outcome-oriented action grouping. In *Proceedings of the ACM Web Conference 2023*, pp. 1220–1230, 2023.

Jérémie Rappaz, Julian McAuley, and Karl Aberer. *Recommendation on Live-Streaming Platforms: Dynamic Availability and Repeat Consumption*, pp. 390–399. Association for Computing Machinery, 2021.

Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618*, 2012.

James M Robins and Andrea Rotnitzky. Semiparametric efficiency in multivariate regression models with missing data. *Journal of the American Statistical Association*, 90(429):122–129, 1995.

Noveen Sachdeva, Yi Su, and Thorsten Joachims. Off-policy bandits with deficient support. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 965–975, 2020.

Noveen Sachdeva, Lequn Wang, Dawen Liang, Nathan Kallus, and Julian McAuley. Off-policy evaluation for large action spaces via policy convolution. *arXiv preprint arXiv:2310.15433*, 2023.

Yuta Saito and Thorsten Joachims. Off-policy evaluation for large action spaces via embeddings. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162, pp. 19089–19122. PMLR, 2022.

Yuta Saito, Qingyang Ren, and Thorsten Joachims. Off-policy evaluation for large action spaces via conjunct effect modeling. In *international conference on Machine learning*, pp. 29734–29759. PMLR, 2023.

Yuta Saito, Jihan Yao, and Thorsten Joachims. POTEC: Off-policy contextual bandits for large action spaces via policy decomposition. In *The Thirteenth International Conference on Learning Representations*, 2025.

Otmane Sakhi, Stephen Bonner, David Rohde, and Flavian Vasile. Blob: A probabilistic model for recommendation that combines organic and bandit signals. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 783–793, 2020.

Otmane Sakhi, Pierre Alquier, and Nicolas Chopin. PAC-Bayesian offline contextual bandits with guarantees. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pp. 29777–29799. PMLR, 2023a.

Otmane Sakhi, David Rohde, and Alexandre Gilotte. Fast offline policy optimization for large scale recommendation. In *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'23/IAAI'23/EAAI'23. AAAI Press, 2023b.

Otmane Sakhi, Imad Aouali, Pierre Alquier, and Nicolas Chopin. Logarithmic smoothing for pessimistic off-policy evaluation, selection and learning. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 80706–80755. Curran Associates, Inc., 2024.

Anshumali Shrivastava and Ping Li. Asymmetric lsh (alsh) for sublinear time maximum inner product search (mips). In *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.

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. The self-normalized estimator for counterfactual learning. *advances in neural information processing systems*, 28, 2015b.

Muhammad Faaiz Taufiq, Arnaud Doucet, Rob Cornish, and Jean-Francois Ton. Marginal density ratio for off-policy evaluation in contextual bandits. *Advances in Neural Information Processing Systems*, 36, 2024.

Mengting Wan, Rishabh Misra, Ndapa Nakashole, and Julian J. McAuley. Fine-grained spoiler detection from large-scale review corpora. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pp. 2605–2610. Association for Computational Linguistics, 2019.

Yu-Xiang Wang, Alekh Agarwal, and Miroslav Dudık. Optimal and adaptive off-policy evaluation in contextual bandits. In *International Conference on Machine Learning*, pp. 3589–3597. PMLR, 2017.

A Extended Related Work

Offline contextual bandits. The contextual bandit framework is widely used for online learning under uncertainty (Lattimore & Szepesvari, 2019). Yet, many applications pose challenges for online exploration, motivating offline approaches that optimize decisions from logged data (Bottou et al., 2013). Since large datasets of past interactions are often available, policies can be improved without new experimentation (Swaminathan & Joachims, 2015a). This setting, known as offline (or off-policy) contextual bandits (Dudík et al., 2011), relies on off-policy evaluation (OPE) to estimate policy performance from logged data. These estimators are then used to learn value-maximizing policies (Off-policy learning, OPL).

Off-policy evaluation. OPE (Dudík et al., 2011; Dudík et al., 2012; Dudik et al., 2014; Wang et al., 2017; Farajtabar et al., 2018; Su et al., 2020; Metelli et al., 2021; Kuzborskij et al., 2021; Saito & Joachims, 2022; Sakhi et al., 2020; Jeunen & Goethals, 2021) has attracted significant attention in recent years, with methods falling into three main categories. The direct method (DM) fits a model to predict expected costs for each context–action pair and then uses it to estimate policy value (Jeunen & Goethals, 2021; Aouali et al., 2025), a strategy that has proven particularly effective in large-scale recommender systems (Sakhi et al., 2020; Jeunen & Goethals, 2021). Inverse propensity scoring (IPS) instead reweights observed outcomes to correct for the bias of the logging policy (Horvitz & Thompson, 1952; Dudík et al., 2012). While IPS is unbiased under absolute continuity, it is highly sensitive to variance and bias when this condition is violated (Sachdeva et al., 2020). A wide range of techniques has been proposed to address this issue, including clipping (Ionides, 2008; Bottou et al., 2013), shrinkage (Su et al., 2020), smoothing (Metelli et al., 2021; Aouali et al., 2023; Sakhi et al., 2024), implicit exploration (Gabbianelli et al., 2024), and self-normalization (Swaminathan & Joachims, 2015b), among others (Aouali et al., 2024). A third line of work combines these two approaches in doubly robust (DR) estimators, which integrate modeling with reweighting for improved bias–variance trade-offs (Robins & Rotnitzky, 1995; Dudík et al., 2011; Dudik et al., 2014; Farajtabar et al., 2018). Our work focuses on off-policy learning using these estimators.

Off-policy learning. OPL is typically built on DM, IPS, or DR. DM selects actions by maximizing predicted reward, either deterministically or stochastically, while IPS and DR optimize a parameterized policy via stochastic gradient descent (Swaminathan & Joachims, 2015a), where the unknown gradient of the true risk must be estimated using reweighting. Beyond these approaches, statistical learning tools have introduced new objectives grounded in PAC-based pessimism, providing stronger theoretical guarantees (London & Sandler, 2019; Sakhi et al., 2023a). Our contribution complements this literature by examining the optimization landscape of OPL in large action spaces, which remains largely underexplored.

Large action spaces. Regularization can improve IPS in moderate settings, but scaling to large action spaces requires additional structure. One prominent direction leverages action embeddings: for example, marginalized IPS (MIPS) (Saito & Joachims, 2022) reduces variance by exploiting embedding information while remaining unbiased if the embeddings capture the causal effects of actions on costs. High-dimensional embeddings, however, can still induce variance, and misspecified embeddings can introduce bias. Recent work addresses these issues by learning embeddings directly from data (Peng et al., 2023; Cief et al., 2024) or relaxing causal assumptions (Taufiq et al., 2024; Saito et al., 2023). A complementary line of research addresses the computational challenges of OPL in large action spaces. training policies over large action spaces scales linearly with the number of actions K, making it computationally prohibitive. Recent advances incorporate fast maximum inner product search (MIPS) (Shrivastava & Li, 2014) into the training loop, reducing complexity to logarithmic in K (Sakhi et al., 2023b). Unlike prior contributions, our work focuses on the optimization landscape of OPL in large action spaces and offers practical guidelines, supported by theoretical justification, to make optimization more tractable. We view this as a fundamental yet relatively unexplored research direction.

B Additional Objectives and Their Asymptotic Solutions

All these previous estimators are linear on the policy π. Another way of reducing variance is smoothing the importance weights, leading to non-linear estimators on the policy and obtaining stochastic, non-deterministic optimal solutions.

IW Exponential Smoothing (IW-ES) Exponential Smoothing (Aouali et al., 2023) can also be applied on the Importance weights themselves instead of the logging propensity, resulting in an estimator:

$$\hat{V}_n^{\texttt{IW-ES}}(\pi) = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{\pi(A_i|X_i)}{\pi_0(A_i|X_i)}\right)^{\alpha} R_i\,.$$

This estimator recovers the following when $n \to \infty$:

$$\begin{aligned} V^{\texttt{IW-ES}}(\pi) &= \mathbb{E}_{x\sim\nu, a\sim\pi_0(\cdot|x)}\left[\left(\frac{\pi(a|x)}{\pi_0(a|x)}\right)^{\alpha} r(x,a)\right] \\ &= \mathbb{E}_{x\sim\nu, a\sim\pi(\cdot|x)}\left[\left(\frac{\pi(a|x)}{\pi_0(a|x)}\right)^{\alpha-1} r(x,a)\right]. \end{aligned}$$

To identify the optimal policy, we need to find the maximizer of this objective w.r.t π. This objective decomposes on the contexts x, meaning that its global minimizer is a minimizer for each x dependent sub-problem. For each $x \in \mathcal{X}$, we can write down our maximization objective as:

$$\begin{aligned} &\max_{\pi(\cdot|x)} \mathbb{E}_{a\sim\pi(\cdot|x)}\left[\left(\frac{\pi(a|x)}{\pi_0(a|x)}\right)^{\alpha-1} r(x,a)\right] \\ &\text{s.t} \quad \sum_{a\in\mathcal{A}} \pi(a|x) = 1. \\ &\qquad \forall a \in \mathcal{A}, \pi(a|x) \geq 0. \end{aligned}$$

This can be solved by setting the Lagragian to 0. There exists a $\lambda \geq 0$ with which the optimal policy $\pi_*^{\texttt{IW-ES}}$ verifies:

$$\begin{aligned} L_\lambda^{\texttt{IW-ES}}(\pi_*^{\texttt{IW-ES}}) = 0 &\iff \alpha\left(\frac{\pi_*^{\texttt{IW-ES}}(a|x)}{\pi_0(a|x)}\right)^{\alpha-1} r(x,a) - \lambda = 0 \\ &\iff \pi_*^{\texttt{IW-ES}}(a|x) \propto r(x,a)^{1/(1-\alpha)}\pi_0(a|x)\,, \end{aligned}$$

which ends the proof.

Logarithmic Smoothing (LS) Hard clipping the weights makes the learning problem non-smooth, with an optimal solution hard to derive. We focus on a new estimator that has been proposed (Sakhi et al., 2024), which can be interpreted as soft clipping, with strong concentration guarantees that logarithmically smooths the importance weights of the estimator with $\lambda \geq 0$:

$$\hat{V}_n^{\texttt{LS}-\lambda}(\pi) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{\lambda}\log\left(1+\lambda\frac{\pi(A_i|X_i)}{\pi_0(A_i|X_i)}\right) R_i\,. \tag{27}$$

Note that the `LS` estimator was introduced with the reward inside the log but we stick to this definition for the ease of derivations it brings. We can derive its optimal policy by solving the following for any $x \in \mathcal{X}$:

$$\begin{aligned} &\pi_*^{\texttt{LS}-\lambda}(a|x) \propto \frac{1}{\lambda}\left(\frac{r(x,a)}{C_0 + L_a} - 1\right)\pi_0(a|x) \\ &\lambda + 1 = \mathbb{E}_{\pi_0(\cdot|x)}\left[\frac{r(x,a)}{C_0+L_a}\right] \\ &\forall a \in \mathcal{A}, \quad L_a = 0 \iff \pi(a|x) > 0 \\ &\qquad\qquad\qquad\quad \iff r(x,a) > C + L_a\,, \end{aligned}$$

with C_0 and L_a positive slack variables. We can observe that the solution interpolates between smooth, full support solution and a degenerate policy (with 0 probability mass for some actions) depending on the value of λ. Precisely, $\lambda \to 0$ recovers the IPS solution (as LS converge towards IPS) while λ big enough recovers a smoother solution, linear on the reward and proportional to:

$$\pi_*^{\mathtt{LS}-\lambda}(a|x) \propto \frac{1}{\lambda}\left(\frac{(\lambda+1)r(x,a)}{\mathbb{E}_{\pi_0(\cdot|x)}\left[r(x,a)\right]} - 1\right)\pi_0(a|x)\mathbb{1}\left[r(x,a) > \mathbb{E}_{\pi_0(\cdot|x)}\left[r(x,a)\right]/(\lambda+1)\right].$$

This means that if we are interested in reaching a smooth solution, we need to increase λ.

C Asymptotic Solution Proofs

C.1 OPE-based solutions

(IPS), its Clipping Variant (cIPS) and ES Recall the definition of the (logging propensity) clipped IPS estimator with $\tau \in [0, 1]$:

$$\hat{V}_n^{\mathtt{cIPS}}(\pi) = \frac{1}{n}\sum_{i=1}^{n}\frac{\pi(A_i|X_i)}{\max\{\pi_0(A_i|X_i), \tau\}}R_i\,.$$

Taking $n \to \infty$, one obtains:

$$\begin{aligned} V^{\mathtt{cIPS}}(\pi) &= \mathbb{E}_{X\sim\nu, A\sim\pi_0(\cdot|x)}\left[\frac{\pi(a|x)}{\max\{\pi_0(a|x), \tau\}}r(x,a)\right] \\ &= \mathbb{E}_{X\sim\nu, A\sim\pi(\cdot|x)}\left[\frac{\pi_0(a|x)}{\max\{\pi_0(a|x), \tau\}}r(x,a)\right]. \end{aligned}$$

As the objective is linear in the policy π, the optimal policy should put for any $x \in \mathcal{X}$, all the mass on the action a that maximizes the weighted reward, giving:

$$\pi_*^{\mathtt{cIPS}}(a|x) = \mathbb{1}\Big[a = \operatorname*{argmax}_{a'\in\mathcal{A}}\frac{\pi_0(a'|x)r(x,a')}{\max\{\pi_0(a'|x), \tau\}}\Big].$$

We recover the solution for IPS when we tend $\tau \to 0$:

$$\pi_*^{\mathtt{IPS}}(a|x) = \mathbb{1}\Big[a = \operatorname*{argmax}_{a'\in\mathcal{A}} r(x,a')\mathbb{1}\left[\pi_0(a'|x) > 0\right]\Big].$$

We also recover the solution of ES just by replacing the clipping function by an exponential function of factor α, obtaining:

$$\pi_*^{\mathtt{ES}}(a|x) = \mathbb{1}\Big[a = \operatorname*{argmax}_{a'\in\mathcal{A}} r(x,a')\pi_0(a'|x)^{1-\alpha}\Big].$$

Doubly Robust (DR) The doubly robust estimator converges to the following quantity:

$$\begin{aligned} V^{\mathtt{DR}}(\pi) &= \mathbb{E}_{X\sim\nu, A\sim\pi_0(\cdot|x)}\left[\frac{\pi(a|x)}{\max\{\pi_0(a|x), \tau\}}(r(x,a) - \hat{r}(x,a))\right] + \mathbb{E}_{x\sim\nu, a\sim\pi(\cdot|x)}\left[\hat{r}(x,a)\right] \\ &= \mathbb{E}_{x\sim\nu, a\sim\pi(\cdot|x)}\left[(r(x,a) - \hat{r}(x,a))\frac{\pi_0(a|x)}{\max\{\pi_0(a|x), \tau\}} + \hat{r}(x,a)\right]. \end{aligned}$$

The objective is linear in π and is thus maximized by the following deterministic decision rule:

$$\pi_*^{\mathtt{DR}}(a|x) = \mathbb{1}\left[a = \operatorname*{argmax}_{a'\in\mathcal{A}}\hat{r}(x,a') + (r(x,a') - \hat{r}(x,a'))\frac{\pi_0(a|x)}{\max\{\pi_0(a|x), \tau\}}\right]$$

Marginalized IPS (MIPS) with clusters. We adopt the same approach to look for the minimizer of MIPS. We generalize the clustering function ϕ to take also the context into account. We write down the estimator:

$$\hat{V}_n^{\mathtt{MIPS}}(\pi) = \frac{1}{n}\sum_{i=1}^{n}\frac{\sum_{a'}\mathbb{1}\left[\phi(a', X_i) = \phi(A_i, X_i)\right]\pi(a'|X_i)}{\sum_{a''}\mathbb{1}\left[\phi(a'', X_i) = \phi(A_i, X_i)\right]\pi_0(a''|X_i)}R_i = \frac{1}{n}\sum_{i=1}^{n}\frac{\pi(C_i|X_i)}{\pi_0(C_i|X_i)}R_i\,,$$

with which, we recover when $n \to \infty$:

$$
\begin{aligned}
V^{\texttt{MIPS}}(\pi) &= \mathbb{E}_{X\sim\nu, A\sim\pi_0(\cdot|x)}\left[\frac{\sum_{a'}\mathbb{1}\left[\phi(a',x)=\phi(a,x)\right]\pi(a'|x)}{\sum_{a''}\mathbb{1}\left[\phi(a'',x)=\phi(a,x)\right]\pi_0(a''|x)} r(x,a)\right] \\
&= \mathbb{E}_{x\sim\nu}\left[\sum_a \pi_0(a|x)\frac{\sum_{a'}\mathbb{1}\left[\phi(a',x)=\phi(a,x)\right]\pi(a'|x)}{\sum_{a''}\mathbb{1}\left[\phi(a'',x)=\phi(a,x)\right]\pi_0(a''|x)} r(x,a)\right] \\
&= \mathbb{E}_{x\sim\nu}\left[\sum_{a'}\pi(a'|x)\sum_a \pi_0(a|x)\frac{\mathbb{1}\left[\phi(a',x)=\phi(a,x)\right]}{\sum_{a''}\mathbb{1}\left[\phi(a'',x)=\phi(a,x)\right]\pi_0(a''|x)} r(x,a)\right] \\
&= \mathbb{E}_{x\sim\nu}\left[\sum_{a'}\pi(a'|x)\mathbb{E}_{a\sim\pi_0(\cdot|x)}\left[\frac{\mathbb{1}\left[\phi(a',x)=\phi(a,x)\right] r(x,a)}{\mathbb{E}_{a''\sim\pi_0(\cdot|x)}\left[\mathbb{1}\left[\phi(a'',x)=\phi(a,x)\right]\right]}\right]\right].
\end{aligned}
$$

The objective is linear in π, and depends on the action a' through its cluster $\phi(a,\cdot)$ alone. This means that multiple solutions are maximizers as long as the policy chooses the best cluster c. We thus write down the asymptotic solution for `MIPS` in the cluster level, giving:

$$
\begin{aligned}
\pi_*^{\texttt{MIPS}}(c|x) &= \mathbb{1}\left[c = \operatorname*{argmax}_{c'\in\mathcal{C}}\left\{\mathbb{E}_{a\sim\pi_0(\cdot|x)}\left[\frac{r(x,a)\mathbb{1}[\phi(a,x)=c']}{\mathbb{E}_{a''\sim\pi_0(\cdot|x)}\left[\mathbb{1}\left[\phi(a'',x)=\phi(a,x)\right]\right]}\right]\right\}\right] \\
&= \mathbb{1}\left[c = \operatorname*{argmax}_{c'\in\mathcal{C}}\left\{\mathbb{E}_{a\sim\pi_0(\cdot|x)}\left[\frac{r(x,a)\mathbb{1}[\phi(a,x)=c']}{\mathbb{E}_{a''\sim\pi_0(\cdot|x)}\left[\mathbb{1}\left[\phi(a'',x)=c'\right]\right]}\right]\right\}\right] \\
&= \mathbb{1}\left[c = \operatorname*{argmax}_{c'\in\mathcal{C}}\left\{\frac{\mathbb{E}_{a\sim\pi_0(\cdot|x)}\left[r(x,a)\mathbb{1}[\phi(a,x)=c']\right]}{\mathbb{E}_{a\sim\pi_0(\cdot|x)}\left[\mathbb{1}\left[\phi(a,x)=c'\right]\right]}\right\}\right],
\end{aligned}
$$

which ends the proof.

Conjunct Effect Modeling (OffCEM). This estimator can be seen as the natural, doubly robust extension of the `MIPS` estimator.

$$
\hat{V}_n^{\texttt{OffCEM}}(\pi) = \frac{1}{n}\sum_{i=1}^n \frac{\pi(C_i|X_i)}{\pi_0(C_i|X_i)}\left(R_i - \hat{r}(A_i,X_i)\right) + \mathbb{E}_{a\sim\pi(\cdot|X_i)}\left[\hat{r}(X_i,a)\right].
$$

We take the sample size n to ∞ and obtain:

$$
\begin{aligned}
V^{\texttt{OffCEM}}(\pi) &= \mathbb{E}_{X\sim\nu, A\sim\pi_0(\cdot|x)}\left[\frac{\sum_{a'}\mathbb{1}\left[\phi(a',x)=\phi(a,x)\right]\pi(a'|x)}{\sum_{a''}\mathbb{1}\left[\phi(a'',x)=\phi(a,x)\right]\pi_0(a''|x)}\left(r(x,a)-\hat{r}(x,a)\right)\right] + \mathbb{E}_{x\sim\nu,a\sim\pi(\cdot|x)}\left[\hat{r}(x,a)\right] \\
&= \mathbb{E}_{x\sim\nu,a'\sim\pi(\cdot|x)}\left[\mathbb{E}_{a\sim\pi_0(\cdot|x)}\left[\frac{\mathbb{1}\left[\phi(a',x)=\phi(a,x)\right]\left(r(x,a')-\hat{r}(x,a')\right)}{\mathbb{E}_{a''\sim\pi_0(\cdot|x)}\left[\mathbb{1}\left[\phi(a'',x)=\phi(a,x)\right]\right]}\right]\right] + \mathbb{E}_{x\sim\nu,a'\sim\pi(\cdot|x)}\left[\hat{r}(x,a')\right] \\
&= \mathbb{E}_{x\sim\nu,a'\sim\pi(\cdot|x)}\left[\mathbb{E}_{a\sim\pi_0(\cdot|x)}\left[\frac{\mathbb{1}\left[\phi(a',x)=\phi(a,x)\right]\left(r(x,a')-\hat{r}(x,a')\right)}{\mathbb{E}_{a''\sim\pi_0(\cdot|x)}\left[\mathbb{1}\left[\phi(a'',x)=\phi(a,x)\right]\right]}\right] + \hat{r}(x,a')\right].
\end{aligned}
$$

The solution depends explicitly on the action a by the reward model $\hat{r}$. We can derive it as the objective is linear on π, obtaining:

$$
\pi_*^{\texttt{OffCEM}}(a|x) = \mathbb{1}\left[a = \operatorname*{argmax}_{a'\in\mathcal{A}}\left\{\hat{r}(x,a') + \frac{\mathbb{E}_{\bar{a}\sim\pi_0(\cdot|x)}\left[r(\bar{a},x)-\hat{r}(\bar{a},x)\mathbb{1}[\phi(x,\bar{a})=\phi(x,a')]\right]}{\pi_0(\phi(x,a')|x)}\right\}\right].
$$

Two Stage Decomposition (POTEC). This is an *optimization strategy* for `OffCEM`. It restricts the policy to a cluster-informed form,

$$
\pi(a \mid x) = \sum_{c\in\mathcal{C}} \pi^{\text{RM}}(a \mid x, c)\pi^{\text{CL}}(c \mid x),
$$

where $\pi^{\text{RM}}(a \mid x, c) = \mathbb{1}[a = \operatorname{argmax}_{a'\in c}\hat{r}(x,a')]$ is fixed, model-based policy that deterministically selects the best action within each cluster. Learning is then simplified to finding the optimal cluster-level policy π^{CL} that maximizes the `OffCEM` objective in Eq. (13):

$$
\hat{V}_n^{\texttt{POTEC}}(\pi^{\text{CL}}) = \frac{1}{n}\sum_{i=1}^n\left(\frac{\pi^{\text{CL}}(C_i \mid X_i)}{\pi_0(C_i \mid X_i)}\left(R_i - \hat{r}(X_i,A_i)\right) + \sum_{c\in\mathcal{C}}\pi^{\text{CL}}(c \mid X_i)\hat{r}_c^*(X_i)\right),
$$

This is exactly the Doubly Robust version of MIPS on the cluster level, the asymptotic solution on the cluster level can be followed in the same fashion:

$$\pi_*^{\text{CL}}(c \mid x) = \mathbb{1}\Big[c = \underset{c' \in \mathcal{C}}{\text{argmax}} \Big\{ \frac{\mathbb{E}_{a \sim \pi_0(\cdot|x)} \left[(r(x,a) - \hat{r}(x,a)) \mathbb{1}[\phi(a,x) = c'] \right]}{\mathbb{E}_{a \sim \pi_0(\cdot|x)} \left[\mathbb{1}\left[\phi(a,x) = c'\right] \right]} + \hat{r}^*_{c'}(x) \Big\} \Big] \,.$$

The optimal policy for the POTEC optimization strategy unfolds as:

$$\pi_*^{\text{POTEC}}(a|x) = \sum_{c \in \mathcal{C}} \pi^{\text{RM}}(a \mid x, c) \pi_*^{\text{CL}}(c \mid x) \,.$$

At first glance, it might be hard to see the connection between POTEC and OffCEM solutions, but they are equivalent. For ease of notation, let us denote by $D_{\hat{r},x}(c)$:

$$D_{\hat{r},x}(c) = \frac{\mathbb{E}_{a \sim \pi_0(\cdot|x)} \left[(r(x,a) - \hat{r}(x,a)) \mathbb{1}[\phi(a,x) = c] \right]}{\mathbb{E}_{a \sim \pi_0(\cdot|x)} \left[\mathbb{1}\left[\phi(a,x) = c\right] \right]} \,.$$

and recall that the optimal policy of OffCEM finds the action a that maximizes:

$$\tilde{V}(x,a) = \hat{r}(x,a) + D_{\hat{r},x}(\phi(a,x)) \,.$$

For any context x, the optimal action a^* of POTEC verifies:

- a^* is in the optimal cluster: $\phi(a^*, x) = c_*(x)$ with $c_*(x) = \text{argmax}_{c \in \mathcal{C}} D_{\hat{r},x}(c) + \hat{r}^*_c(x)$.
- a^* is optimal within that cluster: $a = \text{argmax}_{a \in c_*(x)} \hat{r}(x,a)$.

This means that for all actions a with $\phi(a,x) \neq c_*(x)$, we have:

$$\begin{aligned}
\tilde{V}(x,a) &= D_{\hat{r},x}(\phi(a,x)) + \hat{r}(x,a) \\
&\leq D_{\hat{r},x}(\phi(a,x)) + \hat{r}^*_{\phi(a,x)}(x) \\
&\leq D_{\hat{r},x}(c_*(x)) + \hat{r}^*_{c_*(x)}(x) \\
&= D_{\hat{r},x}(\phi(x,a^*)) + \hat{r}(x,a^*) = \tilde{V}(x,a^*) \,.
\end{aligned}$$

In addition, for all actions a with $\phi(a,x) = c_*(x)$, we have:

$$\begin{aligned}
\tilde{V}(x,a) &= D_{\hat{r},x}(\phi(a,x)) + \hat{r}(x,a) \\
&= D_{\hat{r},x}(c_*(x)) + \hat{r}(x,a) \\
&\leq D_{\hat{r},x}(c_*(x)) + \hat{r}^*_{c_*(x)}(x) = \tilde{V}(x,a^*) \,.
\end{aligned}$$

This means that the optimal action a^* for POTEC is the maximizer of $\tilde{V}(x,a)$, which is exactly the solution of OffCEM.

Policy Convolution (PC) This estimator uses a nearest neighbors function to aggregate the propensities of similar actions, making the hypothesis that similar actions will result in similar reward signal. The esimator writes:

$$\hat{V}_n^{\text{PC}}(\pi) = \frac{1}{n} \sum_{i=1}^{n} \frac{\pi(N_\epsilon(A_i) \mid X_i)}{\pi_0(N_\epsilon(A_i) \mid X_i)} R_i, \quad \text{with } \pi(N_\epsilon(a) \mid x) = \sum_{a' \in N_\epsilon(a)} \pi(a' \mid x).$$

This estimator is equivalent to the following when $n \to \infty$:

$$\begin{aligned}
V^{\text{PC}}(\pi) &= \mathbb{E}_{X \sim \nu, A \sim \pi_0(\cdot|x)} \left[\frac{\sum_{a'} \pi(a'|x) \mathbb{1}\left[a' \in N_\epsilon(a)\right]}{\pi_0(N_\epsilon(a)|x)} r(X,A) \right] \\
&= \mathbb{E}_{x \sim \nu, a \sim \pi(\cdot|x)} \left[\mathbb{E}_{\bar{a} \sim \pi_0(\cdot|x)} \left[\frac{r(\bar{a},x) \mathbb{1}\left[a \in N_\epsilon(\bar{a})\right]}{\pi_0(N_\epsilon(\bar{a})|x)} \right] \right] \,.
\end{aligned}$$

The same argument of linearity applies here, giving us the corresponding asymptotic solution:

$$\pi_*^{\text{PC}}(a|x) = \mathbb{1}\Big[a = \underset{a' \in \mathcal{A}}{\text{argmax}} \Big\{ \mathbb{E}_{\bar{a} \sim \pi_0(\cdot|x)} \left[\frac{r(x,\bar{a}) \mathbb{1}[a' \in N_\epsilon(\bar{a})]}{\pi_0(N_\epsilon(\bar{a})|x)} \right] \Big\} \Big] \,.$$

C.2 SURROGATE-BASED SOLUTIONS

Our objectives can be written in the same form, only choosing for each a different function g:

$$\hat{U}_n^g(\pi) = \frac{1}{n}\sum_{i=1}^{n} g(X_i, A_i, R_i) \log \pi(A_i \mid X_i)\,.$$

We solve the maximization of the asymptotic value of this objective, to recover the solutions of our surrogate objectives as a special case. The objective decomposes on the contexts x, we are thus interested in the following maximization problem for each x:

$$\begin{aligned} &\max_{\pi(\cdot|x)} \mathbb{E}_{a\sim\pi_0(\cdot|x)}\left[\mathbb{E}_r\left[g(x,a,r)\right]\log\pi(a|x)\right] \\ &\text{s.t} \quad \sum_{a\in\mathcal{A}} \pi(a|x) = 1. \\ &\qquad \forall a \in \mathcal{A}, \pi(a|x) \geq 0. \end{aligned}$$

This can be solved by setting the Lagrangian to 0. There exists a $\lambda \geq 0$ with which the optimal policy π_*^{g} verifies for all x and a:

$$\begin{aligned} L_\lambda^{\mathrm{g}}(\pi_*^{\mathrm{g}})(x,a) = 0 &\iff \pi_0(a|x)\frac{1}{\pi_*^{\mathrm{g}}(a|x)}\mathbb{E}_r\left[g(x,a,r)\right] - \lambda = 0 \\ &\iff \pi_*^{\mathrm{g}}(a|x) \propto \mathbb{E}_r\left[g(x,a,r)\right]\pi_0(a|x)\,, \end{aligned}$$

which concludes the proof.

D ADDITIONAL PROOFS

In this section, we prove the propositions about the optimization landscape of OPE based and PWLL based learning approaches. We start by stating the following lemmas, that will be helpful to prove our propositions.

Lemma D.1. *For all $\mathbf{r} \in [0,1]^K$, $\boldsymbol{\theta} \mapsto \hat{V}_n(\pi_{\boldsymbol{\theta}})$ is $5/2$-smooth, i.e., for all $\pi_{\boldsymbol{\theta}} :=$ softmax$(\boldsymbol{\theta})$ and $\pi_{\boldsymbol{\theta}'} :=$ softmax$(\boldsymbol{\theta}')$, we have,*

$$\left|\hat{V}_n(\pi_{\boldsymbol{\theta}'}) - \hat{V}_n(\pi_{\boldsymbol{\theta}}) - \left\langle \frac{d\hat{V}_n(\pi_{\boldsymbol{\theta}})}{d\boldsymbol{\theta}}, \boldsymbol{\theta}' - \boldsymbol{\theta} \right\rangle\right| \leq \frac{5}{4}\cdot\|\boldsymbol{\theta}' - \boldsymbol{\theta}\|_2^2.$$

Proof. See the proof in (Mei et al., 2020b, Lemma 2). □

Lemma D.2. *All the action level estimators* `EST` *in (*`IPS, cIPS, DR, PC`*) can be written, for any policy π, in the form:*

$$\hat{V}_n^{EST}(\pi) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}_{a\sim\pi(\cdot|X_i)}\left[\hat{r}_{EST,i}(a, X_i)\right]\,, \tag{28}$$

For the cluster level estimators/approaches `EST-C` *in (*`MIPS, POTEC`*), we also have*

$$\hat{V}_n^{EST\text{-}C}(\pi) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}_{c\sim\pi(\cdot|X_i)}\left[\hat{r}_{EST\text{-}C,i}(c, X_i)\right]\,, \tag{29}$$

meaning that all these estimators are linear in π.

Proof. This is straightforward to prove. We begin by the action level estimators and take `DR` as a representative. For `DR`, we have the following:

$$\hat{r}_{\mathrm{DR},i}(a, X_i) = \hat{r}(a, X_i) + \mathbb{I}[a = A_i]\frac{R_i - \hat{r}(A_i, X_i)}{\max(\tau, \pi_0(A_i|X_i))}$$

verifies the equation. Solutions for cIPS and IPS can be recovered directly, and PC follows the same construction. For the cluster level approaches, we take POTEC as a representative, and we have:

$$\hat{r}_{\texttt{POTEC},i}(c, X_i) = \hat{r}^\star_c(X_i) + \mathbb{I}[c = X_i]\frac{R_i - \hat{r}(A_i, X_i)}{\pi_0(C_i|X_i)} ,$$

The $\hat{r}_{\texttt{MIPS},i}$ follows as a special case when $\hat{r} = 0$. □

Lemma D.3. *For problems with a single state x, and for any EST in (IPS, cIPS, DR, OffCEM, MIPS, PC), there is a problem, defined by r, π_0 for which we obtain:*

$$\hat{r}_{EST}(a) = \mathbb{1}\left[a = a_K\right] , \tag{30}$$

with a_K being the optimal action. We can also find a problem, for any cluster level approach (POTEC and also MIPS) with:

$$\hat{r}_{EST\text{-}C}(c) = \mathbb{1}\left[c = c_{|\mathcal{C}|}\right] , \tag{31}$$

with $c_{|\mathcal{C}|}$ being the optimal cluster.

Proof. Let us prove this for the cIPS for the action level estimators and POTEC for the cluster level approach. The result can be adapted for other estimators as they follow the same construction. To simplify the analysis, we suppose that $\pi_0(a) > 0$ for all actions. For cIPS, we have for any $\tau \in [0, 1]$, for n large enough:

$$\hat{r}_{\texttt{EST}}(a) = \frac{\pi_0(a)}{\max(\pi_0(a), \tau)} r(a) .$$

We consider the problem where actual rewards are of the following form:

$$r(a) = \mathbb{1}\left[a = a_K\right] \frac{\max(\pi_0(a), \tau)}{\pi_0(a)} \in \mathbb{R}^+,$$

However, as the rewards need to be in $[0, 1]$, we choose $\tau < \max_a \pi(a)$. We can choose $a_K = \operatorname{argmax}_a \frac{\pi_0(a)}{\max(\pi_0(a),\tau)}$, and obtain $r(a_K) = 1$ and $r(a) = 0$ otherwise, giving $r \in [0, 1]$, and proving the existence of the problem. The same construction follows for IPS, ES and DR.

For the POTEC approach, we get also, for n large enough:

$$\hat{r}_{\texttt{POTEC}}(c) = \max_{a \in c} \hat{r}(a) + \frac{\sum_{a \in c} \pi_0(a)(r(a) - \hat{r}(a))}{\pi_0(c)}$$

We define the problem where action a_K, the optimal action is really far from all the other actions in the cluster space, to get the action a_K with its own cluster, i.e. $c_{|\mathcal{C}|} = \phi(a_K) = \{a_K\}$, and $\forall a \neq a_K , a_K \notin \phi(a)$. If we choose $r(a_K) = 1$ and $r(a) = 0$ otherwise, and additionally $\hat{r}(a_K) = 1 - \epsilon$ and $\hat{r}(a) = \epsilon$ otherwise, we get:

$$\hat{r}_{\texttt{POTEC}}(c) = \mathbb{1}\left[c = c_{|\mathcal{C}|}\right] .$$

This proves that there is a configuration in which the estimator has a one-hot reward. The same constructions can be done for all other estimators, based on the same ideas. □

Now we restate Proposition 2.1 and proceed to its proof.

Proposition D.4. *Let $\hat{V}_n$ an OPE estimator linear in π, π_n its maximizer. Let $\eta \in (0, 1]$ a learning rate. Even for a single context x, and a linear softmax policy $\pi_\theta(a) = \exp(\theta_a)/\sum_{a' \in \mathcal{A}_{\text{EFF}}} \exp(\theta_{a'})$ with $\mathcal{A}_{\text{EFF}}$ its effective action space, there exist a problem such that gradient descent **cannot escape a suboptimal region** before $t_0 = C.K_{\text{EFF}} = \mathcal{O}(K_{\text{EFF}})$ as we have:*

$$\forall t \leq t_0 : \hat{V}_n(\pi_n) - \hat{V}_n(\pi_{\theta_t}) \geq 0.9$$

Proof. The proof of this result follows the same technique as (Mei et al., 2020a, Theorem 1). We adapt it here and derive it for the sake of completeness. Let EST be one of the estimators with action level policies considered before and let $\hat{V}_n$ be that estimator. Consider the single context case where:

$$\hat{r}_{\text{EST}}(a) = \mathbb{1}\left[a = a_K\right] . \tag{32}$$

This case exists per Lemma D.3. This means that for a policy π:

$$\hat{V}_n(\pi) = \pi(a_K).$$

If $a_K \in \mathcal{A}_{\text{EFF}}$, this means that the maximizer $\pi_n = \text{argmax}_{\pi_\theta} \hat{V}_n(\pi_\theta)$, reaches $\hat{V}_n(\pi_n) = 1$ and we have for any $\pi_{\boldsymbol{\theta}}$:

$$\begin{aligned} r(a_K) - \hat{V}_n(\pi_{\boldsymbol{\theta}}) &= 1 - \pi_{\boldsymbol{\theta}}(a_K). \\ r(a) - \hat{V}_n(\pi_{\boldsymbol{\theta}}) &= -\pi_{\boldsymbol{\theta}}(a_K)\,, \forall a \neq a_K. \end{aligned}$$

The condition $a_K \in \mathcal{A}_{\text{EFF}}$ is important, as it ensures us that the considered family of parametrized policies include the optimal policy. This is the reason why $\mathcal{A}_{\text{EFF}}$ should be constructed using information from the optimal policy for the estimator optimized. For our policy, parametrized by a softmax $\pi_{\boldsymbol{\theta}}(a) \propto \exp(\theta_a)\mathbb{I}[a \in \mathcal{A}_{\text{EFF}}]$, the ℓ_2 norm of the gradient is upper bounded by:

$$\begin{aligned} \left\| \frac{d\hat{V}_n(\pi_{\boldsymbol{\theta}})}{d\boldsymbol{\theta}} \right\|_2 &= \sqrt{\pi_{\boldsymbol{\theta}}(a_K)^2(1-\pi_{\boldsymbol{\theta}}(a_K))^2 + \pi_{\boldsymbol{\theta}}(a_K)^2 \sum_{a \neq K} \pi_{\boldsymbol{\theta}}(a)^2} \\ &= \pi_{\boldsymbol{\theta}}(a_K) \cdot \sqrt{(1-\pi_{\boldsymbol{\theta}}(a_K))^2 + \sum_{a \neq K} \pi_{\boldsymbol{\theta}}(a)^2} \\ &\leq \pi_{\boldsymbol{\theta}}(a_K) \cdot \sqrt{(1-\pi_{\boldsymbol{\theta}}(a_K))^2 + \left(\sum_{a \neq K} \pi_{\boldsymbol{\theta}}(a) \right)^2} \\ &= \pi_{\boldsymbol{\theta}}(a_K) \cdot \sqrt{(1-\pi_{\boldsymbol{\theta}}(a_K))^2 + (1-\pi_{\boldsymbol{\theta}}(a_K))^2} \\ &= \sqrt{2} \cdot \pi_{\boldsymbol{\theta}}(a_K) \cdot (1-\pi_{\boldsymbol{\theta}}(a_K)). \end{aligned}$$

Let $\boldsymbol{\theta}_{t+1} \leftarrow \boldsymbol{\theta}_t + \eta_t \cdot \frac{d\hat{V}_n(\pi_{\boldsymbol{\theta}_t})}{d\boldsymbol{\theta}_t}$, and $\pi_{\boldsymbol{\theta}_{t+1}} = \text{softmax}(\boldsymbol{\theta}_{t+1})$ be the next policy after one step gradient update. Define the following two kinds of iterations:

$$t_{\text{good}} := \left\{ t \geq 1 : \pi_{\boldsymbol{\theta}_{t+1}}(a_K) > \pi_{\boldsymbol{\theta}_t}(a_K) \right\},$$

$$t_{\text{bad}} := \left\{ t \geq 1 : \pi_{\boldsymbol{\theta}_{t+1}}(a_K) \leq \pi_{\boldsymbol{\theta}_t}(a_K) \right\}.$$

For all $t \in t_{\text{bad}}$, we have,

$$\frac{1}{\pi_{\boldsymbol{\theta}_t}(a_K)} - \frac{1}{\pi_{\boldsymbol{\theta}_{t+1}}(a_K)} = \frac{1}{\pi_{\boldsymbol{\theta}_{t+1}}(a_K) \cdot \pi_{\boldsymbol{\theta}_t}(a_K)} \cdot \left(\pi_{\boldsymbol{\theta}_{t+1}}(a_K) - \pi_{\boldsymbol{\theta}_t}(a_K) \right) \leq 0.$$

For all $t \in t_{\text{good}}$, we have,

$$\begin{aligned}
\pi_{\boldsymbol{\theta}_{t+1}}(a_K) - \pi_{\boldsymbol{\theta}_t}(a_K) &= \left[\hat{V}_n(\pi_{\boldsymbol{\theta}_{t+1}}) - \hat{V}_n(\pi_{\boldsymbol{\theta}_t})\right] \\
&= \left[\hat{V}_n(\pi_{\boldsymbol{\theta}_{t+1}}) - \hat{V}_n(\pi_{\boldsymbol{\theta}_t}) - \left\langle \frac{d\hat{V}_n(\pi_{\boldsymbol{\theta}_t})}{d\boldsymbol{\theta}_t}, \boldsymbol{\theta}_{t+1} - \boldsymbol{\theta}_t \right\rangle + \left\langle \frac{d\hat{V}_n(\pi_{\boldsymbol{\theta}_t})}{d\boldsymbol{\theta}_t}, \boldsymbol{\theta}_{t+1} - \boldsymbol{\theta}_t \right\rangle\right] \\
&\leq \left[\frac{5}{4} \cdot \|\boldsymbol{\theta}_{t+1} - \boldsymbol{\theta}_t\|_2^2 + \left\langle \frac{d\hat{V}_n(\pi_{\boldsymbol{\theta}_t})}{d\boldsymbol{\theta}_t}, \boldsymbol{\theta}_{t+1} - \boldsymbol{\theta}_t \right\rangle\right] \quad \text{(by Lemma D.1)} \\
&= \left(\frac{5\eta_t^2}{4} + \eta_t\right) \cdot \left\| \frac{d\hat{V}_n(\pi_{\boldsymbol{\theta}_t})}{d\boldsymbol{\theta}_t} \right\|_2^2 \quad \left(\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \eta_t \cdot \frac{d\hat{V}_n(\pi_{\boldsymbol{\theta}_t})}{d\boldsymbol{\theta}_t}\right) \\
&\leq \left(\frac{5\eta_t^2}{4} + \eta_t\right) \cdot 2 \cdot \pi_{\boldsymbol{\theta}_t}(a_K)^2 \cdot (1 - \pi_{\boldsymbol{\theta}_t}(a_K))^2 \\
&\leq \frac{9}{2} \cdot \pi_{\boldsymbol{\theta}_t}(a_K)^2 \cdot (1 - \pi_{\boldsymbol{\theta}_t}(a_K))^2 \quad (\eta_t \in (0, 1]) \\
&\leq \frac{9}{2} \cdot \pi_{\boldsymbol{\theta}_t}(a_K)^2. \quad (\pi_{\boldsymbol{\theta}_t}(a_K) \in [0, 1])
\end{aligned}$$

Dividing both sides with $\pi_{\boldsymbol{\theta}_{t+1}}(a_K) \cdot \pi_{\boldsymbol{\theta}_t}(a_K)$, we have,

$$\frac{1}{\pi_{\boldsymbol{\theta}_t}(a_K)} - \frac{1}{\pi_{\boldsymbol{\theta}_{t+1}}(a_K)} \leq \frac{9}{2} \cdot \frac{\pi_{\boldsymbol{\theta}_t}(a_K)}{\pi_{\boldsymbol{\theta}_{t+1}}(a_K)} \leq \frac{9}{2}. \quad \left(\pi_{\boldsymbol{\theta}_{t+1}}(a_K) \geq \pi_{\boldsymbol{\theta}_t}(a_K) > 0\right)$$

Therefore, we have,

$$\begin{aligned}
\frac{1}{\pi_{\boldsymbol{\theta}_1}(a_K)} - \frac{1}{\pi_{\boldsymbol{\theta}_t}(a_K)} &= \sum_{s=1}^{t-1} \left[\frac{1}{\pi_{\boldsymbol{\theta}_s}(a_K)} - \frac{1}{\pi_{\boldsymbol{\theta}_{s+1}}(a_K)}\right] \\
&= \sum_{s=1, s\in t_{\text{good}}}^{t-1} \left[\frac{1}{\pi_{\boldsymbol{\theta}_s}(a_K)} - \frac{1}{\pi_{\boldsymbol{\theta}_{s+1}}(a_K)}\right] + \sum_{s=1, s\in t_{\text{bad}}}^{t-1} \left[\frac{1}{\pi_{\boldsymbol{\theta}_s}(a_K)} - \frac{1}{\pi_{\boldsymbol{\theta}_{s+1}}(a_K)}\right] \\
&\leq \sum_{s=1, s\in t_{\text{good}}}^{t-1} \left[\frac{1}{\pi_{\boldsymbol{\theta}_s}(a_K)} - \frac{1}{\pi_{\boldsymbol{\theta}_{s+1}}(a_K)}\right] \\
&\leq \sum_{s=1, s\in t_{\text{good}}}^{t-1} \left[\frac{9}{2}\right] \\
&\leq \frac{9}{2} \cdot t.
\end{aligned}$$

In the majority of the scenarios, the parameters are initialized randomly. It means that at initialization, we have $\pi_{\boldsymbol{\theta}_1}(a_K) = 1/K_{\text{EFF}}$. Once K_{EFF} large enough, we have $\pi_{\boldsymbol{\theta}_1}(a_K) \leq \frac{1}{c}$, for some constant $c = 11$. If $t \leq \frac{2}{9c} \cdot K_{\text{EFF}}$, then we have,

$$\begin{aligned}
\frac{1}{\pi_{\boldsymbol{\theta}_t}(a_K)} &\geq \frac{1}{\pi_{\boldsymbol{\theta}_1}(a_K)} - \frac{9}{2} \cdot t \\
&\geq \frac{1}{\pi_{\boldsymbol{\theta}_1}(a_K)} \cdot \left(1 - \frac{1}{c}\right) \geq c \cdot \left(1 - \frac{1}{c}\right) = c - 1 \geq 10,
\end{aligned}$$

which implies $\pi_{\boldsymbol{\theta}_t}(a_K) \leq \frac{1}{10}$. Therefore, for all $t \leq \frac{2}{9c} \cdot K_{\text{EFF}}$, we have,

$$\hat{V}_n(\pi_n) - \hat{V}_n(\pi_{\theta_t}) = (1 - \pi_{\boldsymbol{\theta}_t}(a_K)) \geq 0.9\,.$$

The same exact proof can be done using Lemma D.3 for cluster level estimators, where $\mathcal{A}_{\text{EFF}} = \mathcal{C}$ the cluster space. □

Proposition D.5. *Even for a single context x, deterministic rewards, there is problem where OPE-based learning with a linear softmax policy $\pi_\theta(a) \propto \exp(\langle\theta, \phi(x,a)\rangle)\mathbb{I}[a \in \mathcal{A}_{\text{EFF}}]$ can have a number of local maxima* ***exponential in the number of effective actions*** *K_{EFF}.*

Proof. Let EST an off-policy estimators considered in the paper with an action-level policy. By Lemma D.2, we have:

$$\hat{V}_n^{\text{EST}}(\pi) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}_{a\sim\pi(\cdot|x_i)}\left[\hat{r}_{\text{EST},i}(a, x_i)\right], \tag{33}$$

In a single context setting, it becomes:

$$\hat{V}_n^{\text{EST}}(\pi_\theta) = \mathbb{E}_{a\sim\pi_\theta(\cdot)}\left[\frac{1}{n}\sum_{i=1}^{n} \hat{r}_{\text{EST},i}(a)\right], \tag{34}$$

$$= \langle \frac{1}{n}\sum_{i=1}^{n} \hat{r}_{\text{EST},i}, \pi_\theta\rangle\,. \tag{35}$$

This also holds for estimators with policies in the cluster level, as we still have:

$$\hat{V}_n^{\text{EST-C}}(\pi_\theta) = \mathbb{E}_{c\sim\pi_\theta(\cdot)}\left[\frac{1}{n}\sum_{i=1}^{n} \hat{r}_{\text{EST-C},i}(c)\right], \tag{36}$$

$$= \langle \frac{1}{n}\sum_{i=1}^{n} \hat{r}_{\text{EST-C},i}, \pi_\theta\rangle\,. \tag{37}$$

These softmax policies are all defined on the effective action space $\mathcal{A}_{\text{EFF}}$, be it a subset of the action space $\mathcal{A}$ or the discrete cluster space $\mathcal{C}$. Using the linearity of the objective, we can directly apply Theorem 1 from Chen et al. (2019) and obtain our result. □

Finally, we also restate Proposition 3.1, and provide its proof.

Proposition D.6. *For an ℓ_2 regularized (adding $\lambda||\theta||^2$, with $\lambda > 0$), linear softmax policy π_θ, the PWLL objective $\hat{U}_n^g(\pi_\theta)$ defined as:*

$$\hat{U}_n^g(\pi) = \frac{1}{n}\sum_{i=1}^{n} g(R_i, \pi_0(A_i \mid X_i)) \log \pi(A_i \mid X_i)\,,$$

is λ-strongly concave. Without regularization, the objective is concave.

Proof. For any x and $a \in \mathcal{A}_{\text{EFF}}(x)$, we have:

$$\pi_\theta(a|x) = \frac{\exp(\langle\theta, \phi(x,a)\rangle)}{\sum_{a'\in\mathcal{A}_{\text{EFF}}(x)} \exp(\langle\theta, \phi(x,a')\rangle)}\,,$$

optimizing an ℓ_2 regularized linear softmax, giving:

$$\hat{L}_n^{g,\lambda}(\pi) = \hat{U}_n^{\text{g}}(\pi) - \lambda||\theta||^2\,,$$

with $\lambda > 0$ and recall that $g \geq 0$. For strong concavity, we need to show that the Hessian $\nabla_\theta^2 \hat{U}_n^{\text{g}}(\pi_\theta)$ is negative definite with eigenvalues bounded away from zero.

The gradient with respect to θ is: $\nabla_\theta \hat{U}_n^{\text{g}}(\pi_\theta) = \frac{1}{n}\sum_{i=1}^{n} g(R_i, \pi_0(A_i \mid X_i))\nabla_\theta \log \pi_\theta(A_i|X_i) - \lambda\theta$

For the softmax policy:

$$\nabla_\theta \log \pi_\theta(a|x) = \phi(x,a) - \sum_{a'} \pi_\theta(a'|x)\phi(x,a') = \phi(x,a) - \mathbb{E}_{\pi_\theta(\cdot|x)}[\phi(x,\cdot)]$$

Therefore: $\nabla_\theta \hat{U}_n^{\text{g}}(\pi_\theta) = \frac{1}{n}\sum_{i=1}^{n} g(R_i, \pi_0(A_i \mid X_i))\left(\phi(X_i, A_i) - \mathbb{E}_{\pi_\theta(\cdot|X_i)}[\phi(X_i,\cdot)]\right) - \lambda\theta$

Table 1: Statistics of Post Processed Datasets

Dataset	Num. of actions	Num. of samples
MovieLens	60,000	132,744
Twitch	200,000	400,000
GoodReads	1,000,000	400,000

Taking the second derivative: $\nabla^2_\theta \hat{U}^{\mathrm{g}}_n(\pi_\theta) = -\frac{1}{n}\sum_{i=1}^n g(R_i, \pi_0(A_i \mid X_i))\nabla_\theta \mathbb{E}_{\pi_\theta(\cdot|X_i)}[\phi(X_i,\cdot)] - \lambda I_d$, where I_d is the $d \times d$ identity matrix. The gradient of the expectation is:

$$\nabla_\theta \mathbb{E}_{\pi_\theta(\cdot|x)}[\phi(x,\cdot)] = \sum_a \nabla_\theta \pi_\theta(a|x)\phi(x,a)$$

Using $\nabla_\theta \pi_\theta(a|x) = \pi_\theta(a|x)(\phi(x,a) - \mathbb{E}_{\pi_\theta(\cdot|x)}[\phi(x,\cdot)])$:

$$\nabla_\theta \mathbb{E}_{\pi_\theta(\cdot|x)}[\phi(x,\cdot)] = \sum_a \pi_\theta(a|x)(\phi(x,a) - \mathbb{E}_{\pi_\theta(\cdot|x)}[\phi(x,\cdot)])\phi(x,a)^\top$$

This simplifies to:

$$\nabla_\theta \mathbb{E}_{\pi_\theta(\cdot|x)}[\phi(x,\cdot)] = \mathrm{Cov}_{\pi_\theta(\cdot|x)}[\phi(x,\cdot)]$$

where $\mathrm{Cov}_{\pi_\theta(\cdot|x)}[\phi(x,\cdot)] = \mathbb{E}_{\pi_\theta(\cdot|x)}[\phi(x,\cdot)\phi(x,\cdot)^\top] - \mathbb{E}_{\pi_\theta(\cdot|x)}[\phi(x,\cdot)]\mathbb{E}_{\pi_\theta(\cdot|x)}[\phi(x,\cdot)]^\top$

Therefore:

$$\nabla^2_\theta \hat{U}^{\mathrm{g}}_n(\pi_\theta) = -\frac{1}{n}\sum_{i=1}^n g(R_i, \pi_0(A_i \mid X_i))\mathrm{Cov}_{\pi_\theta(\cdot|X_i)}[\phi(X_i,\cdot)] - \lambda I_d$$

We can write this as: $\nabla^2_\theta \hat{U}^{\mathrm{g}}_n(\pi_\theta) = -H - \lambda I_d$

where $H = \frac{1}{n}\sum_{i=1}^n g(R_i, \pi_0(A_i \mid X_i))\mathrm{Cov}_{\pi_\theta(\cdot|X_i)}[\phi(X_i,\cdot)]$ is positive semi-definite. To see this explicitly, for any vector $v \in \mathbb{R}^d$:

$$v^\top \mathrm{Cov}_{\pi_\theta(\cdot|X_i)}[\phi(X_i,\cdot)]v = \mathrm{Var}_{\pi_\theta(\cdot|X_i)}[v^\top \phi(X_i,\cdot)] \geq 0\,,$$

with the positivity of g, this ensures H is positive semi-definite. Then we have:

$$v^\top \nabla^2_\theta \hat{U}^{\mathrm{g}}_n(\pi_\theta)v = -v^\top Hv - \lambda v^\top v = -v^\top Hv - \lambda\|v\|^2\,,$$

meaning that when $v \neq 0$, we get $v^\top \nabla^2_\theta \hat{U}^{\mathrm{g}}_n(\pi_\theta)v \leq -\lambda\|v\|^2 < 0\,.$

This shows the Hessian is negative definite with all eigenvalues bounded above by $-\lambda < 0$. Therefore, ℓ_2 regularized $\hat{U}^{\mathrm{g}}_n(\pi_\theta)$ is λ-strongly concave. In addition, when $\lambda = 0$, the hessian is negative semi-definite, giving simple concavity. □

E Additional Experiments

E.1 Detailed Experimental Setting

Experimental Setting. Our experimental setup is designed to study the behavior of the different policy learning paradigms in large action spaces. To this end, we use three large action spaces collaborative filtering datasets: Movielens (Lam & Herlocker, 2016), Twitch (Rappaz et al., 2021) and GoodReads (Wan et al., 2019) that are preprocessed to obtain a user-item interaction matrix[2]. The statistics of these datasets are described in Table 1. The large action space scenario restricts the policies used to the inner product parametrization (Aouali et al., 2022). This parametrization is

[2]Code and datasets are heavy, both will be released upon acceptance.

essential to leverage Maximum Inner Product Search algorithms (Shrivastava & Li, 2014) for fast query response. In particular, we adopt policies of the following form:

$$\pi_\theta(a|x) \propto \exp(\langle h_\Gamma(x), \beta_a \rangle)\,,$$

with the learnable parameter $\theta = [\Gamma, \beta]$, $h_\Gamma : \mathcal{X} \to \mathbb{R}^\ell$ defines the context embedding function in $\mathbb{R}^\ell$ and β the actions embeddings of size $K \times \ell$. In all our experiments and unless it is explicitly stated, we follow the procedure of Sakhi et al. (2023b) to define our policies. We start by extracting action embeddings β_0 using an SVD decomposition of the user-item matrix. These embeddings help us define the context embedding function h_Γ and our logging policy π_0. h_0 is set to the average embeddings of the observed actions in the contexts and is fixed for the logging policy π_0. Using the SVD action embeddings β_0, we define our logging policy π_0 as:

$$\pi_0(a|x) \propto \exp\left(\frac{1}{t}\langle h_0(x), \beta_{0,a}\rangle\right) \mathbb{I}\left[a \in \textsc{top}^{k_0}(x)\right]\,,$$

with t the temperature of the logging policy, and k_0 define the support of the logging policy, concentrating on the top k_0 actions with: $\textsc{top}^{k_0}(x) = \operatorname{argsort}_{a_1,\cdots,a_{k_0}} \langle h_0(x), \beta_{0,a}\rangle$.

If not explicitly stated, k_0 is set to 100 and the temperature at $t = 1$ in all experiments. This policy is used to collect the offline dataset $\mathcal{D}_n = \{X_i, A_i, R_i\}_{i\in[n]}$ on which all trainings are conducted.

Trained Policies Parameterizations. We adopt two parameterizations of the trained policies. The first one is a **heavyweight** parametrization, and focuses on learning the embeddings of the actions β (be it $\mathcal{A}$ of size K or $\mathcal{C}$ of size $|\mathcal{C}|$), meaning that θ in this case is β. For action-level policies, this gives $\beta \in \mathbb{R}^{K\times\ell}$ and for any x:

$$\pi_\beta(a|x) = \frac{\exp(\langle h_0(x), \beta_a\rangle)}{\sum_{a'\in\mathcal{A}_{\text{EFF}}(x)} \exp(\langle h_0(x), \beta_{a'}\rangle)}\,,$$

with $\mathcal{A}_{\text{EFF}}(x) \subset \mathcal{A}$, which depends on the choice of the practitioner, for example $\mathcal{A}_{\text{EFF}}(x) = S_0(x)$, the support of π_0 for context x when we optimize IPS objectives. For cluster-level policies, this gives a $\beta \in \mathbb{R}^{|\mathcal{C}|\times\ell}$ and for any x:

$$\pi_\beta(c|x) = \frac{\exp(\langle h_0(x), \beta_c\rangle)}{\sum_{c'\in\mathcal{C}} \exp(\langle h_0(x), \beta_{c'}\rangle)}\,.$$

This is used by default if nothing is explicitly stated.

We have also define a **lightweight** parametrization, where only a small projection $W \in \mathbb{R}^{\ell\times\ell}$ is learned, giving in action level policies:

$$\pi_W(a|x) = \frac{\exp(\langle h_0(x)W, \beta_{a,0}\rangle)}{\sum_{a'\in\mathcal{A}_{\text{EFF}}(x)} \exp(\langle h_0(x)W, \beta_{a',0}\rangle)}\,,$$

using β_0, the embeddings of π_0. For cluster level policies, we first define $\bar\beta_0 \in \mathbb{R}^{|\mathcal{C}|\times\ell}$ with $\bar\beta_{0,c} = \frac{1}{|c|}\sum_{a\in c}\beta_{0,a}$, and use it to define the cluster level policy:

$$\pi_W(c|x) = \frac{\exp(\langle h_0(x)W, \bar\beta_{c,0}\rangle)}{\sum_{c'\in\mathcal{C}} \exp(\langle h_0(x)W, \bar\beta_{c',0}\rangle)}\,.$$

Reward Model. The reward model used $\hat r$ is learned using regularized linear regression the collected interaction data, with $\hat r(x, a) = \langle h(x), \theta_a\rangle$.

Clustering and ϵ used. We use the embeddings β_0, combined with K-means clustering to find our clusters. The number of clusters is set to 2000 for all datasets and experiments. For `PC`, the ℓ_2 threshold ϵ is set to 0.1.

E.2 Additional results

Training progress using two different parametrizations. Fig. 4 shows the training progress over 10 epochs on all three datasets, comparing heavyweight vs. lightweight policies.

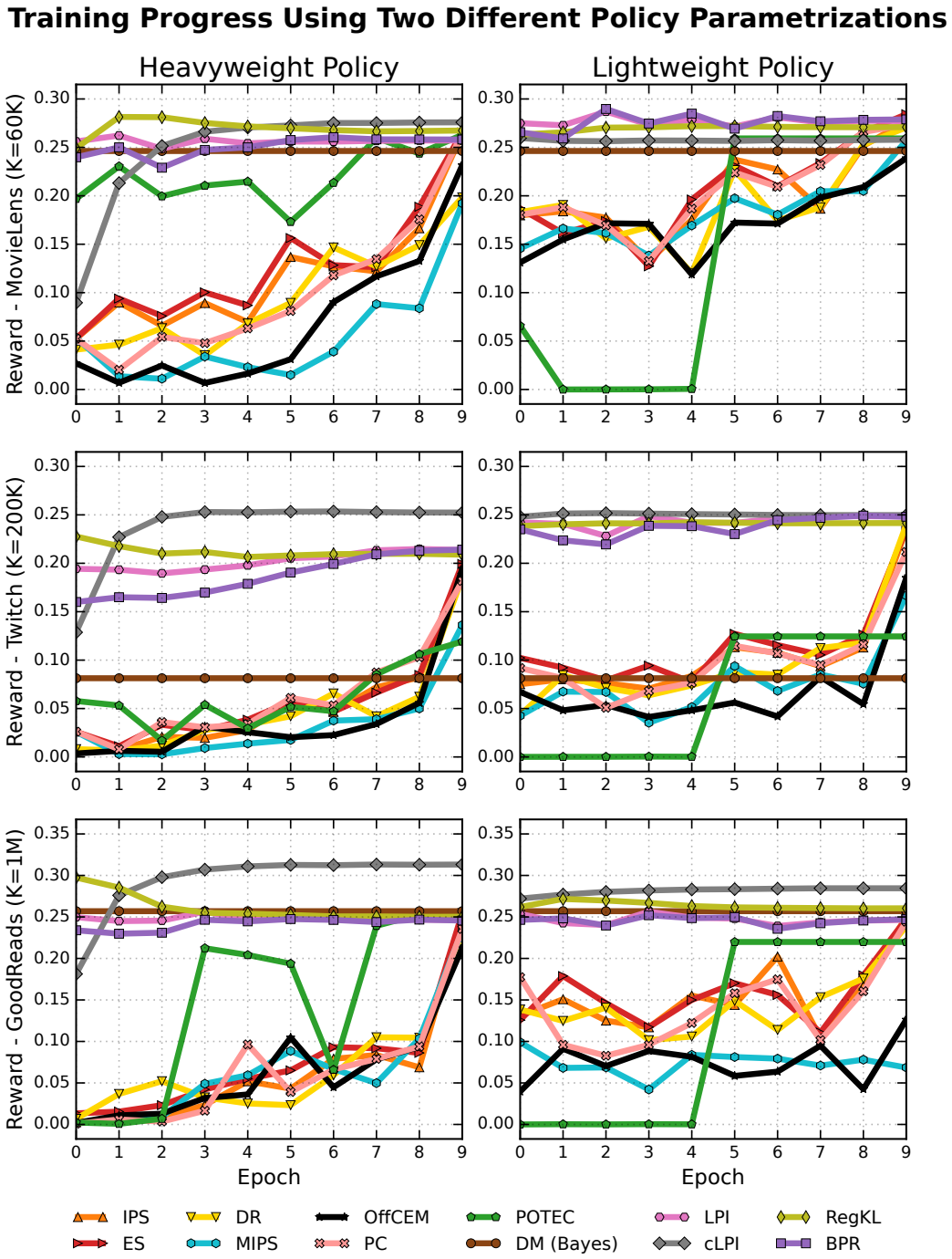

Figure 4: Training progress over 10 epochs on all three datasets, comparing heavyweight vs. lightweight policies.

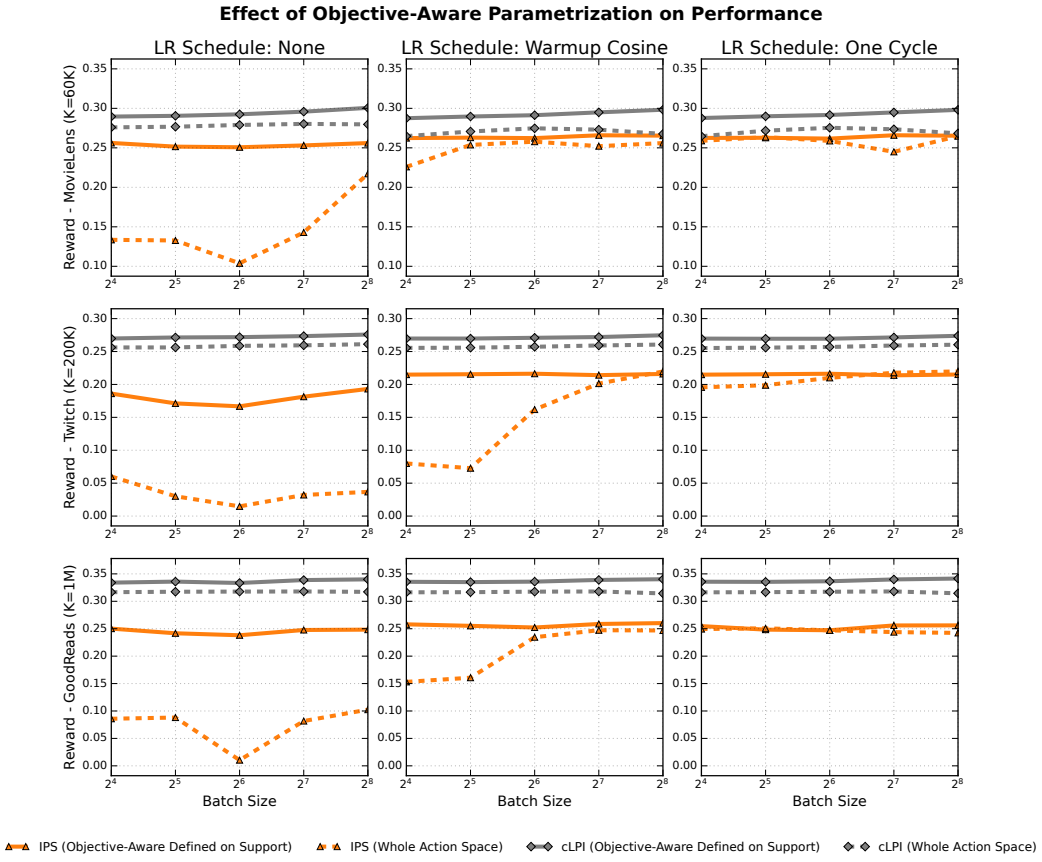

Figure 5: The effect of objective-aware parametrization for `IPS` and `cLPI` on three large-scale datasets

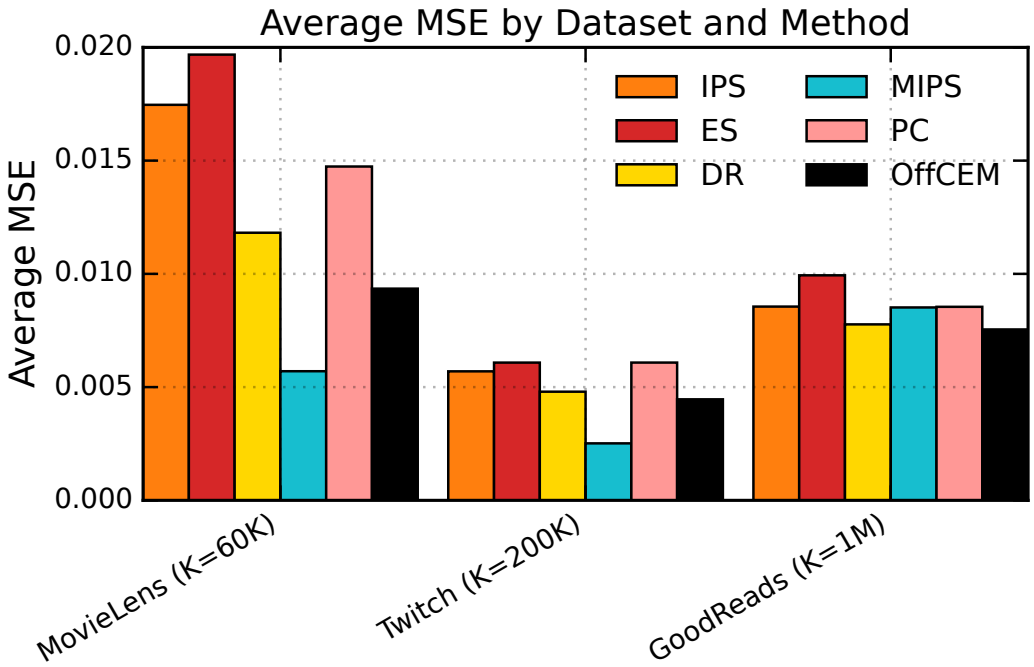

Figure 6: Average MSE by Dataset and Method. Several methods are excluded from the figure, as their high MSE values would distort the scale and obscure the comparison.

Benefits of objective-aware parametrization. Fig. 5 shows the effect of objective-aware policy parameterizations for two different objectives and three large action space datasets.

Average MSE. Fig. 6 shows the average MSE by dataset and method. Several methods are excluded from the figure, as their high MSE values would distort the scale and obscure the comparison.

MSE progress during training. Figs. 7 to 9 show the progress of the MSE over 10 epochs on all three datasets. Several methods are excluded from the figure, as their high MSE values would distort the scale and obscure the comparison.

LLM USAGE

We acknowledge the use of Large Language Models (LLMs) for writing assistance in the preparation of this manuscript. Their role was strictly limited to improving sentence-level grammar, phrasing, and readability. The core scientific ideas, experimental design, analysis, and conclusions presented herein are entirely the work of the authors.

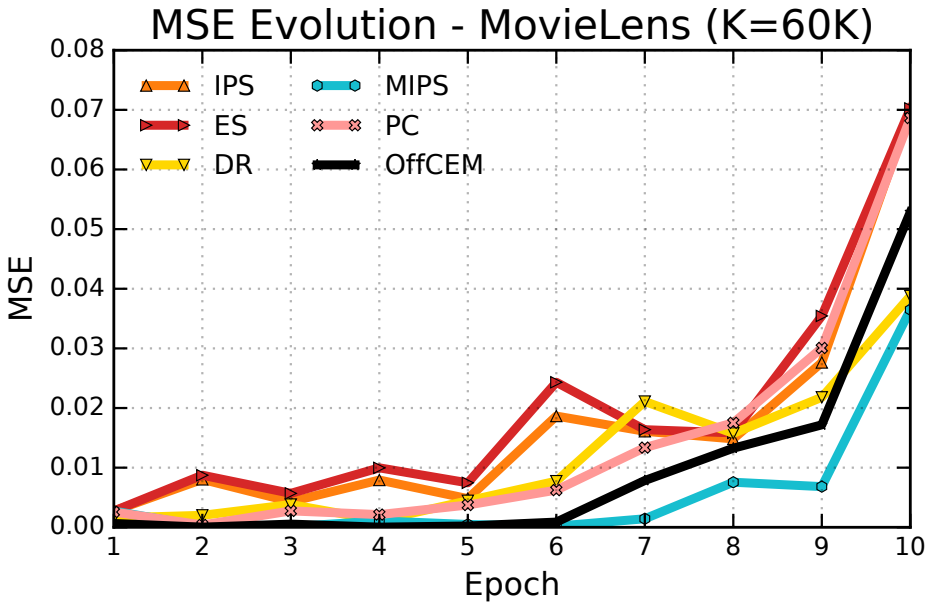

Figure 7: MSE progress over 10 epochs on `MovieLens`.

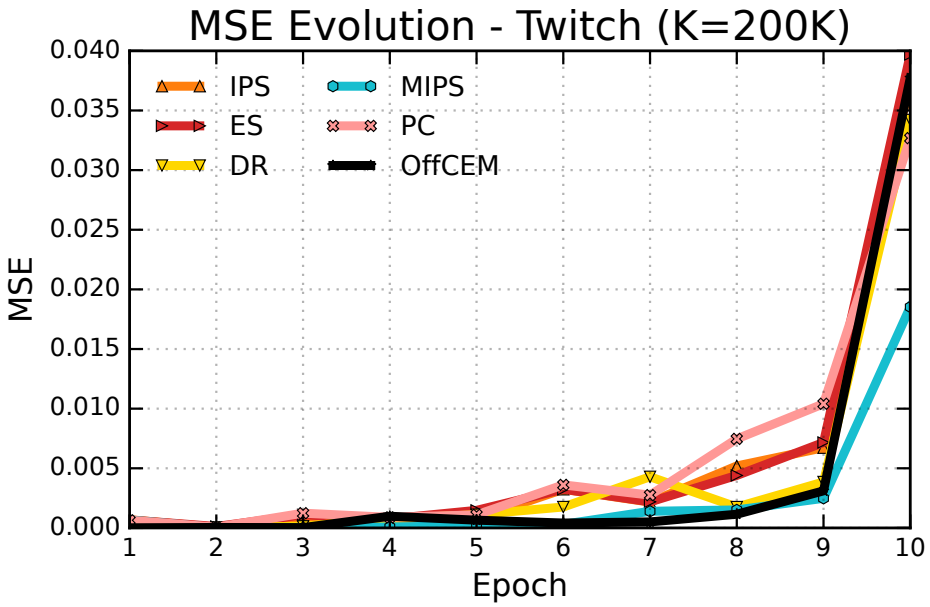

Figure 8: MSE progress over 10 epochs on `Twitch`.

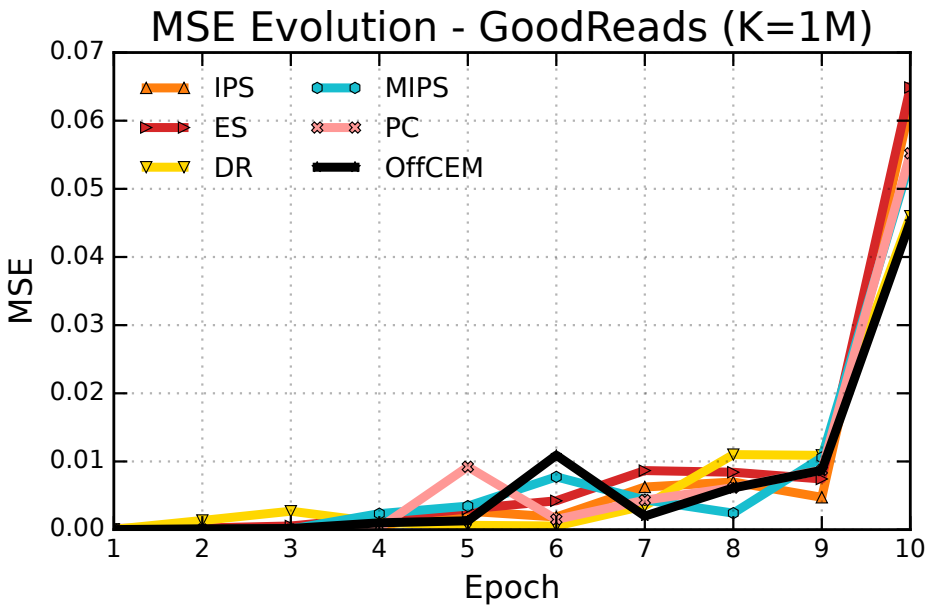

Figure 9: MSE progress over 10 epochs on `GoodReads`.