Off-Policy Evaluation for Large Action Spaces via Conjunct Effect Modeling

Yuta Saito [1] **Qingyang Ren** [1] **Thorsten Joachims** [1]

Abstract

We study off-policy evaluation (OPE) of contextual bandit policies for large discrete action spaces where conventional importance-weighting approaches suffer from excessive variance. To circumvent this variance issue, we propose a new estimator, called *OffCEM*, that is based on the *conjunct effect model* (CEM), a novel decomposition of the causal effect into a cluster effect and a residual effect. OffCEM applies importance weighting only to action clusters and addresses the residual causal effect through model-based reward estimation. We show that the proposed estimator is unbiased under a new condition, called *local correctness*, which only requires that the residual-effect model preserves the relative expected reward differences of the actions within each cluster. To best leverage the CEM and local correctness, we also propose a new two-step procedure for performing model-based estimation that minimizes bias in the first step and variance in the second step. We find that the resulting OffCEM estimator substantially improves bias and variance compared to a range of conventional estimators. Experiments demonstrate that OffCEM provides substantial improvements in OPE especially in the presence of many actions.

1. Introduction

Logged feedback is widely available in intelligent systems for a growing range of applications from media streaming to precision medicine. Many of these systems interact with their users through the *contextual bandit* process, where a policy repeatedly observes a context, takes an action, and observes the reward for the chosen action. A common counterfactual question in this setting is that of *off-policy evaluation* (OPE): how well would a new policy have performed, if it had been deployed instead of a logging policy? The ability to accurately answer this question using previously logged feedback data is of great practical importance, as it avoids costly and time-consuming online experiments (Dudík et al., 2014; Su et al., 2020a).

While we already have practical OPE estimators for small action spaces (Dudík et al., 2014; Swaminathan & Joachims, 2015a; Wang et al., 2017; Farajtabar et al., 2018; Su et al., 2019; 2020a; Metelli et al., 2021), the effectiveness of these estimators degrades for large action spaces, which are common in many potential applications of OPE with thousands or millions of actions (e.g., movies, songs, products). In particular, when the number of actions is large, existing estimators – most of which are based on importance weighting – can collapse due to extreme variance caused by large importance weights (Saito & Joachims, 2022).

To alleviate the variance problem caused by large action spaces, this paper introduces the ***C**onjunct **E**ffect **M**odel (**CEM**)*, which decomposes the expected reward into a *cluster effect* and a *residual effect*. The cluster effect considers the realistic situation where some clustering of the actions already accounts for much of the reward variation (e.g., movie genres). The *residual effect* only needs to model what causal effects remain from individual actions (e.g. movies) that are not already modeled by the cluster effect. This is a substantially more general formulation compared to the Marginalized Inverse Propensity Score (MIPS) estimator of Saito & Joachims (2022), which completely ignores the residual effect and assumes no direct effect for individual actions. Based on our new model of the reward function, we propose a novel estimator, called ***Off**-policy evaluation estimator based on the **C**onjunct **E**ffect **M**odel (**OffCEM**)*, which applies importance weighting over the action clusters to unbiasedly estimate the cluster effect while dealing with the residual effect via model-based estimation to avoid extreme variance. We first show that OffCEM is unbiased under a new condition called *local correctness*, which only requires that the model-based residual-effect estimation preserves the relative value difference of the actions within each action cluster. To best leverage the structure of the

[1]Department of Computer Science, Cornell University, Ithaca, NY, USA. Correspondence to: Yuta Saito <ys552@cornell.edu>, Thorsten Joachims <tj@cs.cornell.edu>.

Proceedings of the 40th International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

CEM, we also provide a new learning procedure for a model-based estimator that directly optimizes the bias and variance of OffCEM. In particular, we propose to optimize regression models via a two-step procedure where the first-step minimizes the bias by optimizing local correctness and the second-step minimizes the variance by optimizing a baseline value function for each action cluster.

Furthermore, we provide a thorough statistical comparison against a range of conventional estimators. In particular, we show that our estimator has a lower variance than Inverse Propensity Score (IPS), MIPS, and Doubly Robust (DR) (Dudík et al., 2014) because OffCEM considers importance weighting with respect to the action cluster space, which is much more compact than the original action space. Moreover, OffCEM often has a lower bias than MIPS because it explicitly addresses the residual effect via model-based estimation. Comprehensive experiments on synthetic and extreme classification data demonstrate that OffCEM can be substantially more effective than conventional estimators, improving the bias-variance trade-off particularly for challenging problems with many actions.

2. Background

We begin with a formal definition of OPE in the contextual bandit setting. We also describe and discuss the limitations of IPS, DR, and MIPS as the benchmark estimators.[1]

2.1. Off-Policy Evaluation

We formulate OPE following the general contextual bandit process, where a decision maker repeatedly observes a context $x \in \mathcal{X}$ drawn i.i.d. from an unknown distribution $p(x)$. Given context x, a possibly stochastic *policy* $\pi(a|x)$ chooses action a from a finite action space denoted as $\mathcal{A}$. The reward $r \in [0, r_{\max}]$ is then sampled from an unknown distribution $p(r|x,a)$, and we use $q(x,a) := \mathbb{E}[r|x,a]$ to denote the expected reward given context x and action a. We define the *value* of π as the key performance measure:

$$V(\pi) := \mathbb{E}_{p(x)\pi(a|x)p(r|x,a)}[r] = \mathbb{E}_{p(x)\pi(a|x)}[q(x,a)]$$

The logged bandit data we use is of the form $\mathcal{D} := \{(x_i, a_i, r_i)\}_{i=1}^n$, which contains n independent observations drawn from the logging policy π_0 as $(x,a,r) \sim p(x)\pi_0(a|x)p(r|x,a)$. In OPE, we aim to design an estimator $\hat{V}$ that can accurately estimate the value of a target policy π (which is different from π_0) using only $\mathcal{D}$. We measure the accuracy of $\hat{V}$ by its mean squared error (MSE)

$$\begin{aligned}\mathrm{MSE}(\hat{V}(\pi;\mathcal{D})) &:= \mathbb{E}_{\mathcal{D}}\Big[\big(V(\pi) - \hat{V}(\pi;\mathcal{D})\big)^2\Big], \\ &= \mathrm{Bias}(\hat{V}(\pi;\mathcal{D}))^2 + \mathbb{V}_{\mathcal{D}}\big[\hat{V}(\pi;\mathcal{D})\big],\end{aligned}$$

where $\mathbb{E}_{\mathcal{D}}[\cdot]$ denotes the expectation over the logged data. The bias and variance of $\hat{V}$ are defined respectively as

$$\begin{aligned}\mathrm{Bias}(\hat{V}(\pi;\mathcal{D})) &:= \mathbb{E}_{\mathcal{D}}[\hat{V}(\pi;\mathcal{D})] - V(\pi), \\ \mathbb{V}_{\mathcal{D}}\big[\hat{V}(\pi;\mathcal{D})\big] &:= \mathbb{E}_{\mathcal{D}}\Big[\big(\hat{V}(\pi;\mathcal{D}) - \mathbb{E}_{\mathcal{D}}[\hat{V}(\pi;\mathcal{D})]\big)^2\Big].\end{aligned}$$

2.2. Limits of IPS and DR

Our first benchmark is the IPS estimator, which forms the basis of many other OPE estimators (Dudík et al., 2014; Wang et al., 2017; Su et al., 2019; 2020a; Metelli et al., 2021). IPS estimates the value of π by re-weighting the observed rewards as

$$\hat{V}_{\mathrm{IPS}}(\pi;\mathcal{D}) := \frac{1}{n}\sum_{i=1}^n \frac{\pi(a_i \,|\, x_i)}{\pi_0(a_i \,|\, x_i)} r_i = \frac{1}{n}\sum_{i=1}^n w(x_i, a_i) r_i,$$

where $w(x,a) := \pi(a \,|\, x)/\pi_0(a \,|\, x)$ is the *(vanilla) importance weight*.

This estimator is unbiased (i.e., $\mathbb{E}_{\mathcal{D}}[\hat{V}_{\mathrm{IPS}}(\pi;\mathcal{D})] = V(\pi)$) under the following common support assumption.

Assumption 2.1. (Common Support) The logging policy π_0 is said to have common support for policy π if $\pi(a \,|\, x) > 0 \rightarrow \pi_0(a \,|\, x) > 0$ for all $a \in \mathcal{A}$ and $x \in \mathcal{X}$.

The unbiasedness of IPS is indeed desirable. However, a critical issue is that its variance can be excessive, particularly when there is a large number of actions. The DR estimator (Dudík et al., 2014) has been a popular technique to reduce the variance by incorporating a model-based reward estimator $\hat{q}(x,a) \approx q(x,a)$ as follows.

$$\begin{aligned}&\hat{V}_{\mathrm{DR}}(\pi;\mathcal{D},\hat{q}) \\ &:= \frac{1}{n}\sum_{i=1}^n \left\{ w(x_i,a_i)(r_i - \hat{q}(x_i,a_i)) + \hat{q}(x_i,\pi) \right\},\end{aligned}$$

where $\hat{q}(x,\pi) := \mathbb{E}_{\pi(a|x)}[\hat{q}(x,a)]$. DR often improves the MSE of IPS by reducing the variance. However, its variance can still be extremely large due to vanilla importance weighting, which leads to substantial accuracy deterioration in large action spaces (Saito & Joachims, 2022). This issue of IPS and DR can be seen by calculating their variance

$$\begin{aligned}n\mathbb{V}_{\mathcal{D}}\big[\hat{V}_{\mathrm{DR}}(\pi;\mathcal{D},\hat{q})\big] &= \mathbb{E}_{p(x)\pi_0(a|x)}[w(x,a)^2\sigma^2(x,a)] \\ &\quad + \mathbb{E}_{p(x)}\left[\mathbb{V}_{\pi_0(a|x)}[w(x,a)\Delta_{q,\hat{q}}(x,a)]\right] \\ &\quad + \mathbb{V}_{p(x)}\left[\mathbb{E}_{\pi(a|x)}[q(x,a)]\right], \qquad (1)\end{aligned}$$

where $\sigma^2(x,a) := \mathbb{V}[r|x,a]$ and $\Delta_{q,\hat{q}}(x,a) := q(x,a) - \hat{q}(x,a)$. Note that the variance of IPS can be obtained by setting $\hat{q}(x,a) = 0$. The variance reduction of DR comes from the second term where $\Delta_{q,\hat{q}}(x,a)$ is smaller than $q(x,a)$ if $\hat{q}(x,a)$ is reasonably accurate. However, we can also see

[1] Appendix A provides an extensive discussion of related work.

that the variance contributed by the first term can be extremely large for both IPS and DR when the reward is noisy and the weights $w(x,a)$ have a wide range, which occurs when π assigns large probabilities to actions that have low probability under π_0. This is often the case when there are many actions and the logging policy π_0 aims to guarantee *universal support* (i.e., $\pi_0(a \mid x) > 0$ for all a and x).

2.3. Limits of The MIPS Estimator

To address the excessive variance of IPS and DR, Saito & Joachims (2022) assume the existence of *action embeddings* in the logged data. There are many cases where we have access to such prior information about the actions. For example, movies are characterized by genres (e.g., adventure, documentary), director, or actors. The idea of MIPS (Saito & Joachims, 2022) is to utilize this auxiliary information about the actions to perform OPE more efficiently.

More formally, suppose we are given a d_e-dimensional *action embedding* $e \in \mathcal{E} \subseteq \mathbb{R}^{d_e}$ for each action a, where the embedding is drawn i.i.d. from some unknown distribution $p(e \mid x, a)$. If the embedding is defined by predefined category information, for instance, then it is independent of the context and is deterministic given the action. However, if the action embedding is, for example, a product price generated by some personalized and randomized pricing algorithm, it becomes continuous, stochastic, and context-dependent.

Leveraging the predefined action embeddings, Saito & Joachims (2022) propose the following MIPS estimator.

$$\hat{V}_{\mathrm{MIPS}}(\pi; \mathcal{D}) := \frac{1}{n}\sum_{i=1}^{n} \frac{\pi(e_i \mid x_i)}{\pi_0(e_i \mid x_i)} r_i = \frac{1}{n}\sum_{i=1}^{n} w(x_i, e_i) r_i,$$

where the logged dataset $\mathcal{D} = \{(x_i, a_i, e_i, r_i)\}_{i=1}^n$ now contains action embeddings for each data and

$$w(x,e) := \frac{\pi(e \mid x)}{\pi_0(e \mid x)} = \frac{\sum_{a \in \mathcal{A}} p(e \mid x, a)\pi(a \mid x)}{\sum_{a \in \mathcal{A}} p(e \mid x, a)\pi_0(a \mid x)}$$

is the *marginal importance weight* based on marginal embedding distribution $\pi(e \mid x) := \sum_{a \in \mathcal{A}} p(e \mid x, a)\pi(a \mid x)$.

This estimator is unbiased (i.e., $\mathbb{E}_{\mathcal{D}}[\hat{V}_{\mathrm{MIPS}}(\pi; \mathcal{D})] = V(\pi)$) under the following assumptions (Saito & Joachims, 2022).

Assumption 2.2. (Common Embedding Support) The logging policy π_0 is said to have common embedding support for policy π if $\pi(e \mid x) > 0 \rightarrow \pi_0(e \mid x) > 0$ for all e and x.

Assumption 2.3. (No Direct Effect) Action a has no direct effect on the reward r given the embedding e if $a \perp r \mid x, e$.

Assumption 2.2 is a weaker version of Assumption 2.1, requiring common support only with respect to the action embedding space. Assumption 2.3 is about the quality of the given action embeddings and requires that every possible effect of a on r be fully mediated by embedding e.

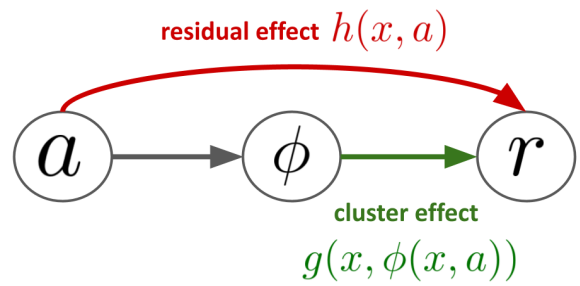

Figure 1. The Conjunct Effect Model (CEM)

Even though MIPS often improves the MSE over existing estimators such as IPS and DR by utilizing action embeddings, we argue that MIPS can still suffer from substantial bias or variance in many realistic situations. First, if the given action embeddings are high-dimensional and fine-grained, MIPS becomes very similar to IPS, producing extreme variance. This is because, in such cases, the embedding distribution $p(e \mid x, a)$ is near-deterministic and thus the marginal importance weight becomes close to the vanilla importance weight, i.e., $w(x,a) \approx w(x,e)$. We can reduce the variance of MIPS by performing action feature selection as described in Saito & Joachims (2022), but this may produce a large amount of bias by violating Assumption 2.3. This bias-variance dilemma of MIPS motivates the development of a new framework and estimator for large action spaces.

3. The Conjunct Effect Model and OffCEM Estimator

The following proposes a new estimator that circumvents the challenges of MIPS. The key idea is to decompose the expected reward into the *cluster effect* and *residual effect* rather than making the no direct effect assumption, which might be unrealistic and thus cause the dilemma. Specifically, given some action clustering function $\phi : \mathcal{X} \times \mathcal{A} \rightarrow \mathcal{C}$, which may be learned from log data, where $\mathcal{C}$ is an action cluster space (typically $|\mathcal{C}| \ll |\mathcal{A}|$), we consider the following *conjunct effect model* (CEM), which decomposes the expected reward function into two separate effects.

The Conjunct Effect Model (CEM):

$$q(x,a) = \underbrace{g(x, \phi(x,a))}_{\textit{cluster effect}} + \underbrace{h(x,a)}_{\textit{residual effect}} \qquad (2)$$

For example, in a movie recommendation problem, the cluster effect could capture the relevance of each genre to the users, and the residual effect models how each movie

is better or worse than the overall genre preference. In the simplest case, a non-personalized residual effect can model that some movies in each genre are generally better than others, or it can model a personalized effect that a specific user likes a particular actor. Note that Assumption 2.3 requires no residual effect for MIPS, i.e., $h(x, a) = 0$ for all (x, a), so our CEM formulation is strictly more general than that of Saito & Joachims (2022).[2]

Our proposed estimator, which we call the **OffCEM** estimator, overcomes the limitations of MIPS by appropriately selecting the estimation strategies for the cluster and residual effect based on the CEM as follows.

$$\hat{V}_{\mathrm{OffCEM}}(\pi; \mathcal{D}) := \frac{1}{n}\sum_{i=1}^{n}\left\{w(x_i, \phi(x_i, a_i))(r_i - \hat{f}(x_i, a_i)) + \hat{f}(x_i, \pi)\right\}, \tag{4}$$

where $\hat{f}(x, \pi) := \mathbb{E}_{\pi(a|x)}[\hat{f}(x, a)]$ and we define the *cluster importance weight* as

$$w(x, c) := \frac{\pi(c \,|\, x)}{\pi_0(c \,|\, x)} = \frac{\sum_{a\in\mathcal{A}} \mathbb{I}\{\phi(x, a) = c\}\pi(a \,|\, x)}{\sum_{a\in\mathcal{A}} \mathbb{I}\{\phi(x, a) = c\}\pi_0(a \,|\, x)},$$

for $c \in \mathcal{C}$. The first term of OffCEM estimates the cluster effect via cluster importance weighting and the second term deals with the residual effect via the regression model $\hat{f}$.[3]

Intuitively, our estimator is expected to perform better than a range of typical estimators for large action spaces. First, OffCEM performs importance weighting with respect to the action cluster space and thus is expected to have a much lower variance than IPS, DR, and MIPS, which apply importance weighting with respect to the original action space (IPS and DR) or potentially high-dimensional action embeddings (MIPS). Moreover, OffCEM can have a lower bias than MIPS by dealing with the residual effect via model-based estimation (the second term) rather than ignoring it as in MIPS. Note that a seemingly natural choice of the regression model $\hat{f}$ is a direct estimate of the residual effect, i.e., $\hat{f} \approx h(x, a)$, but we later show that there is a more refined *two-step* procedure to optimize the regression model to best leverage the structure of our conjunct effect model.

The following analyzes the statistical properties of OffCEM, showing that it is unbiased under a new condition called local correctness and that it has a much more favorable bias-variance tradeoff compared to the conventional estimators. We will also discuss how to optimize the regression model $\hat{f}$ based on our statistical analysis.

3.1. Theoretical Analysis

First, we formally introduce the *local correctness* assumption and show that OffCEM is unbiased under it.

Assumption 3.1. (Local Correctness) A regression model $\hat{f}(\cdot, \cdot)$ and action clustering function $\phi(\cdot, \cdot)$ satisfy local correctness if the following holds true:

$$\Delta_q(x, a, b) = \Delta_{\hat{f}}(x, a, b), \tag{5}$$

for all $x \in \mathcal{X}$ and $a, b \in \mathcal{A}$ with $\phi(x, a) = \phi(x, b)$, where $\Delta_q(x, a, b) := q(x, a) - q(x, b)$ is the difference in the expected rewards between the actions a and b given context x, which we call the *relative value difference* of the actions. $\Delta_{\hat{f}}(x, a, b) := \hat{f}(x, a) - \hat{f}(x, b)$ is an estimate of the relative value difference between a and b based on $\hat{f}$.

Assumption 3.1 only requires that the regression model $\hat{f}$ should correctly preserve the *relative value difference* $\Delta_q(x, a, b)$ between the actions within each action cluster c. Thus, to satisfy local correctness, we need not be able to accurately estimate the absolute value of the reward function, which is typically required for model-based estimators and control variates. Note also that local correctness does not require any particular relationship across different clusters. For instance, suppose there are three movies $a = 1, 2, 3$ where $a = 1$ and $a = 2$ belong to the first cluster $c = 1$ while $a = 3$ belongs to the second cluster $c = 2$. Then, local correctness only requires a regression model $\hat{f}$ to preserve the relative value difference between $a = 1$ and $a = 2$, but $\hat{f}$ does not need to learn any particular relationship between $a = 1, 2$ (the first cluster) and $a = 3$ (the second cluster). Appendix B provides more specific examples of locally correct regression models.

The following shows that local correctness is indeed a new requirement for an unbiased OPE based on our framework.

Proposition 3.2. *Under Assumptions 2.2 (wrt $\mathcal{E} = \mathcal{C}$) and 3.1, OffCEM is unbiased, i.e., $\mathbb{E}_{\mathcal{D}}[\hat{V}_{\mathrm{OffCEM}}(\pi; \mathcal{D})] = V(\pi)$. See Appendix C.1 for the proof.*

Proposition 3.2 shows that OffCEM is unbiased even when MIPS is biased due to violations of Assumption 2.3, if we can merely learn the relative value difference within

[2]Note that we consider action clusters as a particular type (one-dimensional and discrete) of low-dimensional action embeddings for brevity of exposition and analysis. By doing so, the CEM can be formulated using the standard logged dataset of the form: $\mathcal{D} = \{(x_i, a_i, r_i)\}_{i=1}^n$, which does not include action embeddings e. However, our framework can be extended to a more general formulation, as shown below:

$$q(x, a, e) = \underbrace{g(x, \phi(x, e))}_{\textit{embedding effect}} + \underbrace{h(x, a, e)}_{\textit{residual effect}}, \tag{3}$$

where e is a raw action embedding recorded in the dataset and $\phi : \mathcal{X} \times \mathcal{E} \to \mathbb{R}^{d_\phi}$ is a d_ϕ-dimensional (lower-dimensional) latent representation of the action.

[3]Note that our estimator may look similar to DR at first glance, but ours is derived from applying different estimation strategies between the two terms in the CEM and this design principal is substantially different from that of DR.

each cluster c. Beyond unbiasedness, the following also characterizes the magnitude of the bias of OffCEM when Assumption 3.1 is violated.

Theorem 3.3. *(Bias of OffCEM) If Assumption 2.2 (wrt $\mathcal{E} = \mathcal{C}$) is true, but Assumption 3.1 is violated, OffCEM has the following bias for some given regression model $\hat{f}(\cdot,\cdot)$ and clustering function $\phi(\cdot,\cdot)$.*

$$\begin{aligned}&\mathrm{Bias}(\hat{V}_{\mathrm{OffCEM}}(\pi;\mathcal{D}))\\&= \mathbb{E}_{p(x)\pi(c|x)}\Bigg[\sum_{a<b:\phi(x,a)=\phi(x,b)=c} \pi_0(a\,|\,x,c)\pi_0(b\,|\,x,c)\\&\qquad\times(\Delta_q(x,a,b)-\Delta_{\hat{f}}(x,a,b))\\&\qquad\times\left(\frac{\pi(b\,|\,x,c)}{\pi_0(b\,|\,x,c)}-\frac{\pi(a\,|\,x,c)}{\pi_0(a\,|\,x,c)}\right)\Bigg],\end{aligned}$$

where $a,b\in\mathcal{A}$ and $\pi_0(a\,|\,x,c)=\pi_0(a\,|\,x)\mathbb{I}\{\phi(x,a)=c\}/\pi_0(c\,|\,x)$. See Appendix C.2 for the proof.

Theorem 3.3 shows that three factors contribute to the bias of OffCEM when Assumption 3.1 is violated. The first factor is the *entropy of the logging policy within each action cluster*. When action a is near deterministic given context x and cluster c, $\pi_0(a\,|\,x,c)$ becomes close to zero or one, making $\pi_0(a\,|\,x,c)\pi_0(b\,|\,x,c)$ close to zero. This suggests that even if Assumption 3.1 is violated, a sufficiently large cluster space still enables nearly unbiased estimation with OffCEM. The second factor is the *accuracy of the regression model $\hat{f}$ with respect to the relative value difference*, which is quantified by $\Delta_q(x,a,b)-\Delta_{\hat{f}}(x,a,b)$. When $\hat{f}$ preserves the relative value difference of the actions within each cluster well, the second factor becomes small and so does the bias of OffCEM. This also suggests that, in an ideal case when Assumption 3.1 is satisfied, OffCEM is unbiased for any target policy, recovering Proposition 3.2. The final factor is the *relative similarity between logging and target policy within each action cluster* as quantified by $\frac{\pi(b\,|\,x,c)}{\pi_0(b\,|\,x,c)}-\frac{\pi(a\,|\,x,c)}{\pi_0(a\,|\,x,c)}$, which suggests that the bias of OffCEM becomes small if the target and logging policies are similar within each action cluster. Note that this still allows the target and logging policy to be different in how they choose between clusters for the third factor to diminish and thus drive the overall bias to zero.

Next, we analyze the variance of OffCEM, which provides additional insights as to how we should optimize the regression model $\hat{f}$ to best exploit the CEM from Eq. (2).

Proposition 3.4. *(Variance of OffCEM) Under Assumptions 2.2 (wrt $\mathcal{E}=\mathcal{C}$) and 3.1, OffCEM has the following variance.*

$$\begin{aligned}&n\mathbb{V}_{\mathcal{D}}\left[\hat{V}_{\mathrm{OffCEM}}(\pi;\mathcal{D})\right]\\&=\mathbb{E}_{p(x)\pi_0(a|x)}[w(x,\phi(x,a))^2\sigma^2(x,a)]\\&\quad+\mathbb{E}_{p(x)}\left[\mathbb{V}_{\pi_0(a|x)}[w(x,\phi(x,a))\Delta_{q,\hat{f}}(x,a)]\right]\\&\quad+\mathbb{V}_{p(x)}\left[\mathbb{E}_{\pi(a|x)}[q(x,a)]\right],\end{aligned}\tag{6}$$

where $\Delta_{q,\hat{f}}(x,a):=q(x,a)-\hat{f}(x,a)$ is the estimation error of $\hat{f}(x,a)$ against the expected reward function $q(x,a)$. See Appendix C.3 for the proof.

Proposition 3.4 shows that the variance of OffCEM depends only on the cluster importance weight rather than the vanilla importance weight, implying reduced variance compared to IPS and DR (c.f., Eq. (1)). Moreover, when action embeddings e are high-dimensional and fine-grained, the marginal importance weight of MIPS becomes very similar to the vanilla importance weight, and thus the variance of OffCEM is expected to be smaller than that of MIPS in such a situation. Moreover, Eq. (6) implies that we can improve the variance of OffCEM by minimizing $|\Delta_{q,\hat{f}}(x,a)|$, which suggests an interesting *two-step* strategy for optimizing the regression model $\hat{f}$ as described in the next subsection.

3.2. A Two-Step Regression Procedure

The analysis in the previous section suggests how we should optimize the regression model $\hat{f}$ for a given clustering function ϕ.[4] More specifically, Theorem 3.3 showed that the accuracy of estimating the relative value difference $\Delta_q(x,a,b)$ (i.e., satisfaction of local correctness) characterizes the bias of OffCEM while we should minimize $|\Delta_{q,\hat{f}}(x,a)|$ for minimizing its variance as implied in Proposition 3.4. Therefore, we propose the following two-step procedure for optimizing a regression model where the first-step corresponds to bias minimization while the second-step aims to minimize the variance of the resulting estimator.

1. **Bias Minimization Step**: Optimize the pairwise regression function $\hat{h}_\theta:\mathcal{X}\times\mathcal{A}\rightarrow\mathbb{R}$, parameterized by θ, to approximate $\Delta_q(x,a,b)$ via

$$\min_\theta \sum_{(x,a,b,r_a,r_b)\in\mathcal{D}_{pair}} \ell_h\left(r_a-r_b,\hat{h}_\theta(x,a)-\hat{h}_\theta(x,b)\right).$$

2. **Variance Minimization Step**: Optimize the baseline value function $\hat{g}_\psi:\mathcal{X}\times\mathcal{C}\rightarrow\mathbb{R}$, parameterized by ψ,

[4]This clustering can be produced by a standard clustering algorithm, but the development of a refined clustering procedure that more explicitly optimizes the statistical properties of OffCEM is an interesting direction for future work.

to minimize $\Delta_{q,\hat{f}}(x,a)$ given $\hat{f} = \hat{g}_\psi + \hat{h}_\theta$ via

$$\min_{\psi} \sum_{(x,a,r)\in\mathcal{D}} \ell_g \left(r - \hat{h}_\theta(x,a), \hat{g}_\psi(x,\phi(x,a)) \right).$$

$\ell_h, \ell_g : \mathbb{R} \times \mathbb{R} \to \mathbb{R}_{\geq 0}$ are some appropriate loss functions such as cross-entropy or squared loss. As suggested in our analysis, $\hat{h}_\theta(x,a)$ fully characterizes the bias of OffCEM, and thus the second step can focus entirely on variance minimization by optimizing the baseline value function $\hat{g}_\psi(x,\phi(x,a))$, which does not affect the bias. After performing the two-step regression, we construct our regression model as $\hat{f}_{\theta,\psi}(x,a) = \hat{g}_\psi(x,\phi(x,a)) + \hat{h}_\theta(x,a)$.

Note that $\mathcal{D}_{pair}$ is a dataset augmented for performing the bias minimization step, which is defined as

$$\mathcal{D}_{pair} := \left\{ (x,a,b,r_a,r_b) \,|\, \begin{array}{c} (x_a,a,r_a),(x_b,b,r_b) \in \mathcal{D} \\ x = x_a = x_b, \phi(x,a) = \phi(x,b) \end{array} \right\},$$

where we assume the context space is finite here. If it is difficult to find a pair of observed actions that have the same cluster, we can instead define $\mathcal{D}_{pair}$ as

$$\mathcal{D}_{pair} := \left\{ (x,a,b,r_a,r_b) \,|\, \begin{array}{c} (x_a,a,r_a),(x_b,b,r_b) \in \mathcal{D} \\ x = x_a = x_b \end{array} \right\},$$

which may perform better empirically due to increased sample size. Note that, even if this two-step procedure is infeasible due to insufficient pairwise data, we can still perform a conventional regression for the expected absolute reward to learn the parameterized function $\hat{f}_\omega : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$ via:

$$\min_{\omega} \sum_{(x,a,r)\in\mathcal{D}} \ell_f \left(r, \hat{f}_\omega(x,a) \right), \tag{7}$$

and then use $\hat{f}_\omega$ in Eq. (4). Even for such a conventionally trained regression model $\hat{f}_\omega$, the OffCEM estimator still has advantages over the benchmark estimators. In particular, cluster importance weighting still provides improved variance and $\hat{f}_\omega$ is still preferable over ignoring the residual effect. In the following experiments, we empirically evaluate the effectiveness of our two-step regression procedure, but we also find that OffCEM performs better than existing estimators even for conventionally trained $\hat{f}_\omega$ via Eq. (7).

4. Empirical Evaluation

We first evaluate OffCEM on synthetic data to identify the situations where it enables a more accurate OPE. Second, we validate real-world applicability of OffCEM on extreme classification datasets, which can be converted into bandit problems with large action spaces. Our experiments are conducted using the *OpenBanditPipeline* (OBP)[5], an open-source software for OPE provided by Saito et al. (2021a). The experiment code is publicly available on GitHub: https://github.com/usaito/icml2023-offcem.

4.1. Synthetic Data

For the first set of experiments, we create synthetic datasets to be able to compare the estimates to the ground-truth value of the target policies. Our setup imitates a recommender system, where each user repeatedly interacts with the system to provide data useful for the two-step regression method from Section 3.2. Specifically, we first define 200 unique users whose 10-dimensional context vectors x are sampled from the standard normal distribution. We then assign 10-dimensional categorical action embedding $e_a \in \mathcal{E}$ to each action a where each dimension of $\mathcal{E}$ has a cardinality of 5.[6] We also cluster the actions based on their embeddings where the cluster assigned to action a is denoted as $c_a (= \phi(a))$. We then synthesize the expected reward function as

$$q(x,a) = g(x,c_a) + h_{c_a}(x,a) \tag{8}$$

Appendix D.2 defines $g(\cdot,\cdot)$ and $h_\cdot(\cdot,\cdot)$ in detail. We then sample the reward r from a normal distribution with mean $q(x,a)$ and standard deviation $\sigma = 3$.

We synthesize the logging policy π_0 by applying the softmax function to the expected reward function $q(x,a)$ as

$$\pi_0(a \,|\, x) = \frac{\exp(\beta \cdot q(x,a))}{\sum_{a'\in\mathcal{A}} \exp(\beta \cdot q(x,a'))}, \tag{9}$$

where β is a parameter that controls the optimality and entropy of the logging policy, and we use $\beta = -0.1$.

In contrast, the target policy π is defined as

$$\pi(a \,|\, x) = (1-\epsilon) \cdot \mathbb{I}\{a = \arg\max_{a'\in\mathcal{A}} q(x,a')\} + \frac{\epsilon}{|\mathcal{A}|}, \tag{10}$$

where the noise $\epsilon \in [0,1]$ controls the quality of π, and we set $\epsilon = 0.2$ in the main text. Appendix D.2 provides additional results for varying values of ϵ.

To summarize, we first sample a user and define the expected reward $q(x,a)$ as in Eq. (8). We then sample discrete action a from π_0 based on Eq. (9) where action a is associated with a cluster c_a and embedding vector e_a. The reward is then sampled from a normal distribution with mean $q(x,a)$. Iterating this procedure n times generates logged data $\mathcal{D}$ with n independent copies of (x,a,e_a,c_a,r) where the same user may appear multiple times in the logged data and thus the two-step procedure from Section 3.2 is applicable.

[5] https://github.com/st-tech/zr-obp

[6] This is higher-dimensional and finer-grained than the embeddings used in the synthetic experiment of Saito & Joachims (2022).

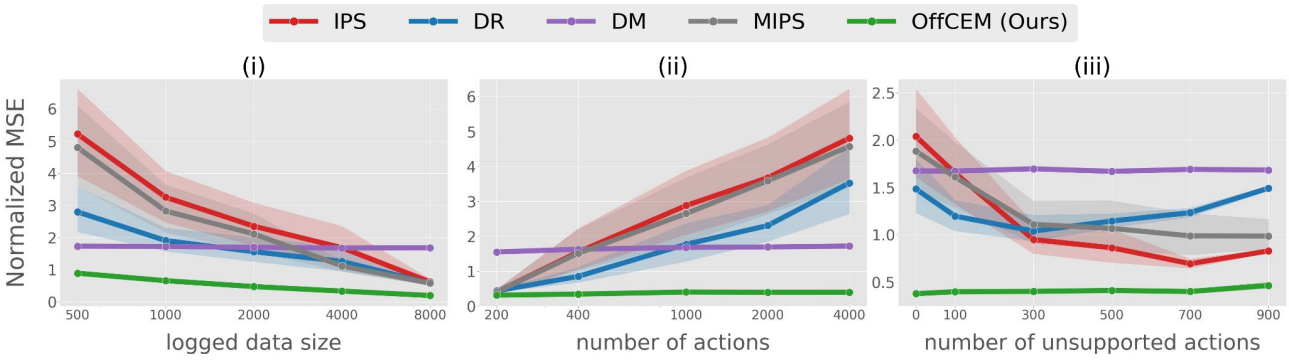

Figure 2. Comparison of the estimators' MSE (normalized by the true value $V(\pi)$) with (i) **varying logged data sizes** (n), (ii) **varying numbers of actions** ($|\mathcal{A}|$), and (iii) **varying numbers of unsupported actions** ($|\mathcal{U}_0|$) in the synthetic experiment.

Table 1. Improvements in MSE provided by OffCEM against the baselines in the synthetic experiment. A smaller value indicates a larger improvement by OffCEM. (default experiment parameters: $n = 3,000$, $|\mathcal{A}| = 1,000$, and $|\mathcal{U}_0| = 0$)

	$n = 500$	$n = 8,000$	$\|\mathcal{A}\| = 200$	$\|\mathcal{A}\| = 4,000$	$\|\mathcal{U}_0\| = 0$	$\|\mathcal{U}_0\| = 900$
$\mathrm{MSE}(\hat{V}_{\mathrm{OffCEM}})/\mathrm{MSE}(\hat{V}_{\mathrm{IPS}})$	0.169	0.314	0.767	0.082	0.185	0.560
$\mathrm{MSE}(\hat{V}_{\mathrm{OffCEM}})/\mathrm{MSE}(\hat{V}_{\mathrm{MIPS}})$	0.184	0.335	0.770	0.086	0.200	0.471
$\mathrm{MSE}(\hat{V}_{\mathrm{OffCEM}})/\mathrm{MSE}(\hat{V}_{\mathrm{DR}})$	0.317	0.339	0.720	0.112	0.254	0.311

4.1.1. Baselines

We compare our estimator with the Direct Method (DM), IPS, DR, and MIPS. We optimize the regression model for OffCEM following the two-step procedure described in Section 3.2. We use a neural network with 3 hidden layers along with 3-fold cross-fitting (Newey & Robins, 2018) to obtain $\hat{q}(x, a)$ for DR and DM, and $(\hat{h}_\theta, \hat{g}_\psi)$ for OffCEM.

4.1.2. Results

Figures 2 and 3 show the MSE (normalized by $V(\pi)$) of the estimators computed over 300 simulations with different random seeds. Table 1 shows the MSE of OffCEM relative to those of the baselines where a smaller value indicates a larger improvement by OffCEM. Note that we use $n = 3,000$, $|\mathcal{A}| = 1,000$, and $|\mathcal{C}| = 50$ as default parameters.

Performance comparisons. First, Figure 2 (i) compares the estimators' performance when we vary the logged data size from 500 to 8,000. We observe that all estimators except for DM improve the MSE with increasing logged data sizes as expected, but OffCEM provides substantial improvements in MSE over IPS, DR, and MIPS, particularly when the logged data size is small. More specifically, Table 1 shows that OffCEM provides approximately 83.0% improvement (reduction) in MSE compared to IPS, 81.5% improvement compared to MIPS, and 68.2% improvement compared to DR when $n = 500$. The improvements of OffCEM when $n = 8,000$ are still substantial but slightly smaller. Note that MIPS performs almost identically to IPS since its marginal importance weight becomes similar to the vanilla importance weight given fine-grained embeddings. OffCEM also performs much better than DM, which suffers from high bias. These strong results of OffCEM are due to its almost zero bias and a variance similar to that of DM.[7]

Next, Figure 2 (ii) reports the estimators' performance when we increase the number of actions from 200 to 4,000. We can see that OffCEM is appealing, particularly for large action spaces, where it outperforms IPS, DR, and MIPS by a larger margin than in small action spaces. Specifically, Table 1 suggests that OffCEM improves IPS and MIPS by approximately 99.1% and DR by 88.7% when $|\mathcal{A}| = 4,000$, which are substantially larger improvements compared to when $|\mathcal{A}| = 200$. We can also see that MIPS suffers from large action spaces similarly to IPS since its marginal importance weight $w(x, e)$ becomes almost identical to the vanilla importance weight $w(x, a)$ of IPS in our synthetic environment. DR is slightly better than IPS and MIPS, but it still suffers from growing variance due to vanilla importance weighting when the number of actions becomes larger. These results suggest that cluster importance weighting is increasingly effective in reducing the variance in larger action spaces, while marginal importance weighting of MIPS fails to improve IPS given fine-grained action embeddings.

[7] Appendix D.2 reports and discusses the bias-variance decomposition of the synthetic results.

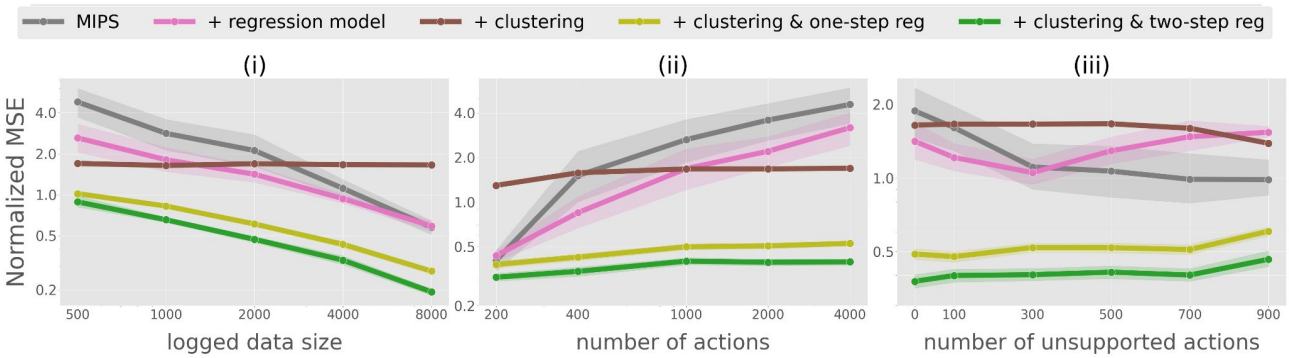

Figure 3. Ablation analysis to evaluate the contribution of the key components of OffCEM with (i) **varying logged data sizes** (n), (ii) **varying numbers of actions** ($|\mathcal{A}|$), and (iii) **varying numbers of unsupported actions** ($|\mathcal{U}_0|$).

Finally, Figure 2 (iii) evaluates the estimators' performance when we increase the number of unsupported actions from 0 to 900, which introduces some bias in IPS, DR, and MIPS by violating Assumption 2.1 (Sachdeva et al., 2020). We can see from the figure that OffCEM is robust to the violation of Assumption 2.1, which is likely to occur in large action spaces, and is consistently more favorable compared to the baseline estimators in various numbers of unsupported actions $|\mathcal{U}_0|$ (where $\mathcal{U}_0 := \{a \in \mathcal{A} \,|\, \pi_0(a \,|\, \cdot) = 0\}$ is the set of unsupported actions). This is due to the fact that OffCEM avoids producing large bias in the cluster effect estimate as long as some actions in each cluster are still supported, while IPS, DR, and MIPS drastically increase their bias. Note that a larger number of unsupported actions introduce a larger bias in IPS, DR, and MIPS, but their MSEs do not necessarily worsen due to reduced variance with a smaller number of supported actions where logging and target policies become relatively similar.

Ablation analysis. In addition to the performance comparisons, we perform an ablation study to investigate how effective each component of OffCEM is. For this purpose, in Figure 3, we gradually add the key components (cluster importance weighting and regression model) of OffCEM to MIPS and compare their MSE in various settings.[8]

We can see from Figure 3 that combining cluster importance weighting and regression model is crucial for improving OPE in large action spaces. More specifically, using only cluster importance weighting (**"+clustering"**) ignores the residual effect and often produces large bias while using only a regression model (**"+regression model"**) produces high variance due to marginal importance weights. The result also suggests that combining only cluster importance weight and one-step regression as in Eq. (7) is much better than **"+clustering"** and **"+regression model"**, and effective enough to greatly improve MIPS. But, we should perform two-step regression when feasible to provide further improvements, which we can see by comparing **"+clustering & one-step reg"** and **"+clustering & two-step reg"**.

[8]The precise definitions of each estimator compared in Figure 3 can be found in Appendix D.1. In particular, **"+clustering"** uses only cluster importance weighting as $\frac{1}{n}\sum_{i=1}^{n} w(x_i, \phi(x_i, a_i))r_i$ while **"+regression model"** adds only a one-step regression model $\hat{q}(x, a)$ to MIPS as $\frac{1}{n}\sum_{i=1}^{n} \{w(x_i, e_i)(r_i - \hat{q}(x_i, a_i)) + \hat{q}(x_i, \pi)\}$. **"+clustering & two-step reg"** and **"+clustering & one-step reg"** are versions of OffCEM w/ or w/o two-step regression.

Table 2. Dataset Statistics

	n_{train}	n_{test}	$\lvert\mathcal{A}\rvert$
EUR-Lex 4K	15,449	3,865	3,956
Wiki10-31K	14,146	6,616	30,938

4.2. Real-World Data

To assess the real-world applicability of OffCEM, we now evaluate OffCEM on extreme classification data. Specifically, we use EUR-Lex 4K and Wiki10-31K from the Extreme Classification Repository (Bhatia et al., 2016), which were used in previous work on off-policy learning in large action spaces (Lopez et al., 2021). Table 2 provides the statistic of these datasets where we can see that both datasets contain several thousands of labels (actions).

Following prior work (Dudík et al., 2014; Wang et al., 2017; Su et al., 2019; 2020a), we convert the extreme classification datasets with L labels into contextual bandit datasets with the same number of actions. We consider stochastic rewards where we first define the base reward function as follows.

$$\tilde{q}(x, a) = \begin{cases} 1 - \eta_a & \text{if } a \text{ is a positive label} \\ \eta_a - 1 & \text{otherwise} \end{cases}$$

η_a is a noise parameter sampled separately for each action

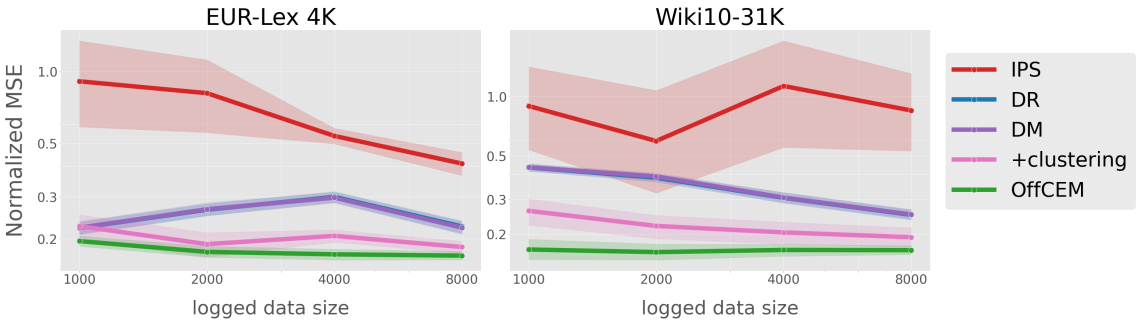

Figure 4. Comparison of the estimators' MSE with varying logged data sizes on the real-world extreme classification datasets.

a from a uniform distribution with range $[0, 0.2]$. Then, for each data, we sample the reward from a Bernoulli distribution with mean $q(x, a) = \sigma(\tilde{q}(x, a))$ where $\sigma(z) = (1 + \exp(-z))^{-1}$ is the sigmoid function.

We define the logging policy π_0 by applying the softmax function to an estimated reward function $\hat{q}(x, a)$ similarly to Eq. (9) where we use $\beta = 30$ and obtain $\hat{q}(x, a)$ by ridge regression with full-information labels. We also define the target policy π following Eq. (10) with $\epsilon = 0.1$.

Results. We compare OffCEM against DM, IPS, and DR. We cannot include MIPS in the real-world experiment since there is no action embeddings available in the extreme classification datasets. Instead, we include "+clustering" from the previous section as the best approximation of MIPS. Note that we obtain $\hat{q}(x, a)$ by simply regressing the observed reward as in Eq. (7) by a neural network with 3 hidden layers and use it for DM and DR. For OffCEM, we perform the two-step regression from Section 3.2 to obtain $\hat{f}(x, a) = \hat{g}_\psi(x, \phi(a)) + \hat{h}_\theta(x, a)$ where $\hat{g}_\psi$ and $\hat{h}_\theta$ are parameterized by a neural network. We use uniform random action clusters for OffCEM with $|\mathcal{C}| = 100$ as a heuristic.

Figure 4 shows the MSEs of the estimators with varying logged data sizes averaged over 100 OPE simulations performed with different random seeds. The figure demonstrates that our OffCEM estimator outperforms all baselines on both datasets due to its substantially reduced variance against the baselines and its small bias. "+clustering" also performs better than DM, IPS, and DR, but it produces larger bias and variance than OffCEM. The results suggest the real-world applicability of OffCEM even with heuristically generated and likely suboptimal action clusters, however, we can still expect a further improvement of OffCEM with a more refined clustering procedure.

5. Conclusion and Future Work

In this paper, we studied the problem of OPE for large action spaces and proposed the conjunct effect model, as well as its associated OffCEM estimator. OffCEM performs importance weighting over action clusters and deals with the remaining bias via model-based estimation. We characterize the bias and variance of the new estimator and show that it has superior statistical properties compared to IPS, DR, and MIPS. We also describe how to optimize the regression model to minimize the bias and variance of OffCEM, leveraging its local correctness condition via the two-step procedure. Extensive experiments on synthetic and real-world data demonstrate that OffCEM provides a substantial reduction in MSE particularly for large actions spaces.

The findings in our paper raise several intriguing questions for future work. In particular, we relied on a heuristic procedure to find a clustering of the actions in the real-world experiment. While OffCEM performed better than the benchmarks even with heuristic clustering, we conjecture that there are more refined clustering procedures as concurrently explored by Peng et al. (2023). For example, it would be valuable to consider an iterative procedure to optimize clustering and regression model simultaneously to satisfy local correctness better. It may also be possible to learn more general types of action representations to bring in further improvements beyond mere clustering. Moreover, this paper only considered the statistical problem of estimating the value of a fixed new policy, so it would be interesting to use our estimator to enable a more efficient off-policy learning in the presence of many actions.

Acknowledgements

This research was supported in part by NSF Awards IIS-1901168 and IIS-2008139. Yuta Saito was supported by Funai Overseas Scholarship. All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

References

Athey, S., Chetty, R., Imbens, G. W., and Kang, H. The surrogate index: Combining short-term proxies to estimate

long-term treatment effects more rapidly and precisely. Technical report, National Bureau of Economic Research, 2019.

Athey, S., Chetty, R., and Imbens, G. Combining experimental and observational data to estimate treatment effects on long term outcomes. *arXiv preprint arXiv:2006.09676*, 2020.

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

Borisov, A., Markov, I., De Rijke, M., and Serdyukov, P. A neural click model for web search. In *Proceedings of the 25th International Conference on World Wide Web*, pp. 531–541, 2016.

Chandak, Y., Theocharous, G., Kostas, J., Jordan, S., and Thomas, P. Learning action representations for reinforcement learning. In *International Conference on Machine Learning*, pp. 941–950. PMLR, 2019.

Chen, J. and Ritzwoller, D. M. Semiparametric estimation of long-term treatment effects. *arXiv preprint arXiv:2107.14405*, 2021.

Chuklin, A., Markov, I., and Rijke, M. d. Click models for web search. *Synthesis lectures on information concepts, retrieval, and services*, 7(3):1–115, 2015.

Dudík, M., Erhan, D., Langford, J., and Li, L. Doubly robust policy evaluation and optimization. *Statistical Science*, 29(4):485–511, 2014.

Dulac-Arnold, G., Denoyer, L., Preux, P., and Gallinari, P. Fast reinforcement learning with large action sets using error-correcting output codes for mdp factorization. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 180–194. Springer, 2012.

Dulac-Arnold, G., Evans, R., van Hasselt, H., Sunehag, P., Lillicrap, T., Hunt, J., Mann, T., Weber, T., Degris, T., and Coppin, B. Deep reinforcement learning in large discrete action spaces. *arXiv preprint arXiv:1512.07679*, 2015.

Farajtabar, M., Chow, Y., and Ghavamzadeh, M. More robust doubly robust off-policy evaluation. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pp. 1447–1456. PMLR, 2018.

Felicioni, N., Ferrari Dacrema, M., Restelli, M., and Cremonesi, P. Off-policy evaluation with deficient support using side information. *Advances in Neural Information Processing Systems*, 35, 2022.

Gu, P., Zhao, M., Chen, C., Li, D., Hao, J., and An, B. Learning pseudometric-based action representations for offline reinforcement learning. In *International Conference on Machine Learning*, pp. 7902–7918. PMLR, 2022.

Guo, F., Liu, C., and Wang, Y. M. Efficient multiple-click models in web search. In *Proceedings of the 2nd ACM International Conference on Web Search and Data Mining*, pp. 124–131, 2009.

Guo, R., Zhao, X., Henderson, A., Hong, L., and Liu, H. Debiasing grid-based product search in e-commerce. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2852–2860, 2020.

Jiang, N. and Li, L. Doubly robust off-policy value evaluation for reinforcement learning. In *Proceedings of the 33rd International Conference on Machine Learning*, volume 48, pp. 652–661. PMLR, 2016.

Kallus, N. and Uehara, M. Double reinforcement learning for efficient off-policy evaluation in markov decision processes. *J. Mach. Learn. Res.*, 21:167–1, 2020.

Kallus, N., Saito, Y., and Uehara, M. Optimal off-policy evaluation from multiple logging policies. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pp. 5247–5256. PMLR, 2021.

Kiyohara, H., Saito, Y., Matsuhiro, T., Narita, Y., Shimizu, N., and Yamamoto, Y. Doubly robust off-policy evaluation for ranking policies under the cascade behavior model. In *Proceedings of the 15th International Conference on Web Search and Data Mining*, 2022.

Lee, J. J., Arbour, D., and Theocharous, G. Off-policy evaluation in embedded spaces. *arXiv preprint arXiv:2203.02807*, 2022.

Li, S., Abbasi-Yadkori, Y., Kveton, B., Muthukrishnan, S., Vinay, V., and Wen, Z. Offline evaluation of ranking policies with click models. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1685–1694, 2018.

Liu, Q., Li, L., Tang, Z., and Zhou, D. Breaking the curse of horizon: infinite-horizon off-policy estimation. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pp. 5361–5371, 2018.

Liu, Y., Bacon, P.-L., and Brunskill, E. Understanding the curse of horizon in off-policy evaluation via conditional importance sampling. In *International Conference on Machine Learning*, pp. 6184–6193. PMLR, 2020.

Lopez, R., Dhillon, I. S., and Jordan, M. I. Learning from extreme bandit feedback. *Proc. Association for the Advancement of Artificial Intelligence*, 2021.

McInerney, J., Brost, B., Chandar, P., Mehrotra, R., and Carterette, B. Counterfactual evaluation of slate recommendations with sequential reward interactions. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1779–1788, 2020.

Metelli, A. M., Russo, A., and Restelli, M. Subgaussian and differentiable importance sampling for off-policy evaluation and learning. *Advances in Neural Information Processing Systems*, 34, 2021.

Narita, Y., Yasui, S., and Yata, K. Efficient counterfactual learning from bandit feedback. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 4634–4641, 2019.

Newey, W. K. and Robins, J. R. Cross-fitting and fast remainder rates for semiparametric estimation. *arXiv preprint arXiv:1801.09138*, 2018.

Pazis, J. and Parr, R. Generalized value functions for large action sets. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, pp. 1185–1192, 2011.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Édouard Duchesnay. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Peng, J., Zou, H., Liu, J., Li, S., Jiang, Y., Pei, J., and Cui, P. Offline policy evaluation in large action spaces via outcome-oriented action grouping. In *Proceedings of the ACM Web Conference 2023*, pp. 1220–1230, 2023.

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

Saito, Y. Doubly robust estimator for ranking metrics with post-click conversions. In *14th ACM Conference on Recommender Systems*, pp. 92–100, 2020.

Saito, Y. and Joachims, T. Counterfactual learning and evaluation for recommender systems: Foundations, implementations, and recent advances. In *Proceedings of the 15th ACM Conference on Recommender Systems*, pp. 828–830, 2021.

Saito, Y. and Joachims, T. Off-policy evaluation for large action spaces via embeddings. In *International Conference on Machine Learning*, pp. 19089–19122. PMLR, 2022.

Saito, Y., Aihara, S., Matsutani, M., and Narita, Y. Open bandit dataset and pipeline: Towards realistic and reproducible off-policy evaluation. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2021a.

Saito, Y., Udagawa, T., Kiyohara, H., Mogi, K., Narita, Y., and Tateno, K. Evaluating the robustness of off-policy evaluation. In *Proceedings of the 15th ACM Conference on Recommender Systems*, pp. 114–123, 2021b.

Su, Y., Wang, L., Santacatterina, M., and Joachims, T. Cab: Continuous adaptive blending for policy evaluation and learning. In *International Conference on Machine Learning*, volume 84, pp. 6005–6014, 2019.

Su, Y., Dimakopoulou, M., Krishnamurthy, A., and Dudík, M. Doubly robust off-policy evaluation with shrinkage. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pp. 9167–9176. PMLR, 2020a.

Su, Y., Srinath, P., and Krishnamurthy, A. Adaptive estimator selection for off-policy evaluation. In *International Conference on Machine Learning*, pp. 9196–9205. PMLR, 2020b.

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

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

Swaminathan, A. and Joachims, T. The self-normalized estimator for counterfactual learning. *Advances in Neural Information Processing Systems*, 28, 2015c.

Swaminathan, A., Krishnamurthy, A., Agarwal, A., Dudik, M., Langford, J., Jose, D., and Zitouni, I. Off-policy evaluation for slate recommendation. In *Advances in Neural Information Processing Systems*, volume 30, pp. 3632–3642, 2017.

Tennenholtz, G. and Mannor, S. The natural language of actions. In *International Conference on Machine Learning*, pp. 6196–6205. PMLR, 2019.

Thomas, P. and Brunskill, E. Data-efficient off-policy policy evaluation for reinforcement learning. In *Proceedings of the 33rd International Conference on Machine Learning*, volume 48, pp. 2139–2148. PMLR, 2016.

Udagawa, T., Kiyohara, H., Narita, Y., Saito, Y., and Tateno, K. Policy-adaptive estimator selection for off-policy evaluation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 2023.

Van Hasselt, H. and Wiering, M. A. Using continuous action spaces to solve discrete problems. In *2009 International Joint Conference on Neural Networks*, pp. 1149–1156. IEEE, 2009.

Vlassis, N., Chandrashekar, A., Gil, F. A., and Kallus, N. Control variates for slate off-policy evaluation. *Advances in Neural Information Processing Systems*, 34, 2021.

Voloshin, C., Le, H. M., Jiang, N., and Yue, Y. Empirical study of off-policy policy evaluation for reinforcement learning. *arXiv preprint arXiv:1911.06854*, 2019.

Wang, T., Gupta, T., Mahajan, A., Peng, B., Whiteson, S., and Zhang, C. Rode: Learning roles to decompose multi-agent tasks. In *International Conference on Learning Representations*, 2020.

Wang, Y.-X., Agarwal, A., and Dudık, M. Optimal and adaptive off-policy evaluation in contextual bandits. In *International Conference on Machine Learning*, pp. 3589–3597. PMLR, 2017.

Xie, T., Ma, Y., and Wang, Y.-X. Towards optimal off-policy evaluation for reinforcement learning with marginalized importance sampling. In *Advances in Neural Information Processing Systems*, pp. 9665–9675, 2019.

A. Related Work

Off-Policy Evaluation in Contextual Bandits and Reinforcement Learning: Off-policy evaluation of fixed decision-making policies is of great practical relevance, providing a safe and cost-effective alternative for online A/B testing (Saito & Joachims, 2021), and thus has gained particular attention in both contextual bandits (Dudík et al., 2014; Wang et al., 2017; Liu et al., 2018; Farajtabar et al., 2018; Su et al., 2019; 2020a; Kallus et al., 2021; Metelli et al., 2021) and reinforcement learning (RL) (Jiang & Li, 2016; Thomas & Brunskill, 2016; Xie et al., 2019; Kallus & Uehara, 2020; Liu et al., 2020). There are three benchmark estimators in this area. The first estimator is DM, which estimates the expected reward function by some off-the-self supervised machine learning methods and then estimates the value of new policies based on the estimated rewards. DM has a lower variance than IPS, but it is susceptible to misspecification of the expected reward function, which is often the case in highly complex real-world data (Farajtabar et al., 2018; Voloshin et al., 2019). The second benchmark estimator is IPS, which applies importance weighting to the observed rewards to estimate the policy value. Under some identification assumptions such as common support and unconfoundedness, IPS enables unbiased and consistent OPE. However, a drawback is that it can produce excessive bias and variance in large action spaces. First, it can have a high bias when the logging policy fails to satisfy the common support condition, which is expected when there are many actions (Sachdeva et al., 2020). There is also the critical variance issue, as the importance weights are likely to be extremely large in the presence of many actions. The weight clipping (Swaminathan & Joachims, 2015b; Su et al., 2019; 2020a) and normalization (Swaminathan & Joachims, 2015c) might be applied to deal with the variance issue, but these simple techniques often produce additional bias. Thus, DR has gained particular attention as a hybrid of DM and IPS, which can achieve a lower bias than DM and a lower variance than IPS (Dudík et al., 2014; Wang et al., 2017; Farajtabar et al., 2018; Su et al., 2020a). Though there are a number of extensions of DR both in contextual bandits (Dudík et al., 2014; Wang et al., 2017; Farajtabar et al., 2018; Su et al., 2020a; Metelli et al., 2021) and RL (Jiang & Li, 2016; Thomas & Brunskill, 2016; Kallus & Uehara, 2020), it may still degrade drastically due to extreme variance in large action spaces (as shown in Eq. (1)). This is because DR relies on the vanilla importance weight defined with respect to the distributions over large action spaces, causing the critical variance issue as in IPS. To overcome this fundamental variance issue of the typical OPE estimators for large action spaces, Saito & Joachims (2022) recently propose a new framework and estimator called MIPS, which utilize some prior information about the actions (i.e., action embeddings or action features) that are widely available in practice and provide structure in the action space. More specifically, MIPS is based on the marginal importance weight, which is defined with respect to the marginal distributions of the action embeddings induced by the target and logging policies. By doing so, it was shown that MIPS has increasingly lower variance than IPS in larger action spaces while remaining unbiased under the no direct effect assumption, which requires that the given action embeddings should be informative enough to be able to mediate every causal effect of the actions on the rewards (Assumption 2.3). A similar assumption is also often utilized when performing causal inference of long-term outcomes via short-term surrogates (Athey et al., 2019; 2020; Chen & Ritzwoller, 2021) and to deal with the deficient support problem in OPE (Felicioni et al., 2022). Moreover, Lee et al. (2022) propose a method based on normalizing flow to estimate the marginal importance weight when the logging and target policies are unknown. Note that, in order to improve the MSE of MIPS, we can perform some data-driven action feature selection in a way that minimizes the MSE of the resulting estimator based on existing estimator selection methods for OPE (Su et al., 2020b; Udagawa et al., 2023). However, a caveat is that MIPS may still have a very large variance similarly to IPS when the given action embeddings are high-dimensional and fined-grained. Moreover, it may produce a large bias if the no direct effect is violated and there is much direct effect of the actions that is not captured by the action embeddings. This bias issue is particularly expected when we perform action feature selection of high-dimensional and granular action embeddings aiming for variance reduction. **Therefore, there remains a critical bias-variance dilemma regarding MIPS – high-dimensional action embeddings should be avoided to guarantee sufficient variance reduction, however, naively doing so via action feature selection might produce large bias by violating no direct effect**. The goal of our work is thus to overcome this bias-variance dilemma of MIPS by generalizing the formulation and proposing a refined estimator. Specifically, rather than making no direct effect, which is often stringent, we decompose the expected reward function into what we call the cluster effect and residual effect. We then apply model-free estimation based on cluster importance weight to estimate the cluster effect with no bias and apply model-based estimation based on the pairwise regression procedure to estimate the residual effect with small variance rather than applying marginal importance weight to both terms as in MIPS, possibly producing excessive variance. Our OffCEM estimator thus achieves much smaller variance than IPS, DR, and MIPS in the presence of many actions or high-dimensional action embeddings, while often reducing the bias of MIPS without ignoring the residual effect.

Off-Policy Evaluation for Slate Recommendations and Rankings: Another line of related work is OPE of slate recommendations and ranking policies where the estimators have to handle combinatorial action spaces, which could be exponentially large (Swaminathan et al., 2017; Li et al., 2018; McInerney et al., 2020; Saito, 2020; Su et al., 2020a; Vlassis et al., 2021; Lopez et al., 2021; Kiyohara et al., 2022). A conventional problem setting in this direction is OPE for slate bandit policies, where it is assumed that only a single, slate-wise reward is observed for each data. Swaminathan et al. (2017) tackle this setting by positing a linearity assumption on the reward function. The proposed pseudoinverse (PI) estimator was shown to provide an exponential gain in the sample complexity over IPS. Following this seminal work, Vlassis et al. (2021) improve the PI estimator by optimizing a set of control variates and Su et al. (2020a) did so by defining some shrinkage estimators. Although these variants of PI are compelling, applications of this class of estimators are limited to the specific problem of slate bandits. On the other hand, our framework is more general and applicable not only to slate bandits, but also to other problem instances including OPE for ranking policies with observable slot-wise rewards (described below) or general contextual bandits with large action spaces. In addition, all estimators for slate bandits rely on the linearity assumption, which is not necessary for our framework.

There is also much work in OPE for ranking policies where it is assumed that the rewards for every slot in a ranking (slot-wise rewards) are observed in the logged data. PI and its variants may be applied to this setting, but it was empirically verified that the PI estimators do not work well, as they do not utilize the slot-wise rewards (McInerney et al., 2020). Thus, some ranking-specific estimators have been proposed to leverage slot-wise rewards based on some assumptions on user behavior from the click modeling literature (Guo et al., 2009; Chuklin et al., 2015). For example, Li et al. (2018) assume that users interact with the items presented in a ranking independently, while McInerney et al. (2020) and Kiyohara et al. (2022) assume that users examine the items one by one from the top position to the bottom one, namely the cascade assumption. However, the reliability of these assumptions depends highly on a ranking interface and real user behavior. If the assumption fails to explain real user behavior well, this approach can produce large bias. For example, the cascade model is only applicable when a ranking interface is vertical, however, real-world ranking interfaces are often more complex such as grid (Guo et al., 2020). Moreover, real-world user behaviors are often highly diverse and heterogeneous and cannot be captured by a single assumption such as cascade (Borisov et al., 2016). In contrast, our framework and estimator are applicable to any ranking interfaces without assuming any particular user behavior via performing clustering of the actions. Moreover, ours is more general in that its application is not limited to information retrieval and recommender systems, but includes robotics, education, conversational agents, or personalized medicine where click models are not relevant

Reinforcement Learning for Large Action Spaces: Although we focus on OPE, there have been several attempts to enable high-performance policy learning for large action spaces. A typical approach is to factorize the action space into binary sub-spaces (Pazis & Parr, 2011; Dulac-Arnold et al., 2012). By contrast, Van Hasselt & Wiering (2009) and Dulac-Arnold et al. (2015) assume the existence of some continuous representations of discrete actions as prior knowledge and utilize continuous policy gradients for policy optimization. Some recent studies also tackle how to learn useful action representations from only available data. Chandak et al. (2019) perform supervised learning to predict the state transitions and obtain action representations with no prior knowledge. Tennenholtz & Mannor (2019) achieve this from trajectories of expert demonstrations. Wang et al. (2020) learn action representations that focus on accurate reconstruction of the rewards and next observations. Gu et al. (2022) perform action representation learning based on a pseudometric with no auxiliary data and perform offline reinforcement learning (offline RL) within the learned metric space. Compared to these works, we consider the conjunct effect model of the reward function and perform cluster importance weighting (combined with a locally correct regression model) to improve OPE in large action spaces rather than the off-policy learning task. Thus, as a future work, it would be interesting to use our estimator to enable a more efficient off-policy learning in the presence of many actions. It would also be valuable to consider extending our idea of the CEM to deal with large state and action spaces in off-policy evaluation of RL policies or offline RL beyond mere contextual bandits.

Table 3. Examples of locally correct regression models

a	a_0	a_1	a_2	a_3
$\phi(x_0, a)$	0		1	
$q(x_0, a)$	4	1	3	2
$\hat{f}_1(x_0, a)$	3	0	1	0
$\Delta_{\cdot,\cdot}(x_0, a, b)$	3		1	

a	a_0	a_1	a_2	a_3
$\phi(x_0, a)$	0		1	
$q(x_0, a)$	4	1	3	2
$\hat{f}_2(x_0, a)$	50	47	-30	-31
$\Delta_{\cdot,\cdot}(x_0, a, b)$	3		1	

a	a_0	a_1	a_2	a_3
$\phi(x_0, a)$	0		1	
$q(x_0, a)$	4	1	3	2
$\hat{f}_3(x_0, a)$	4	1	3	2
$\Delta_{\cdot,\cdot}(x_0, a, b)$	3		1	

B. Examples: Locally Correct Regression Models

This section provides some examples of regression model $\hat{f}$ that satisfy Assumption 3.1 (local correctness). Suppose that there is only a single context $\mathcal{X} = \{x_0\}$ and four actions $\mathcal{A} = \{a_0, a_1, a_2, a_3\}$. The expected reward function $q(x, a)$ and clustering function $\phi(x, a)$ are given as follows.

$$q(x_0, a_0) = 4,\ q(x_0, a_1) = 1,\ q(x_0, a_2) = 3,\ q(x_0, a_3) = 2,$$
$$\phi(x_0, a_0) = 0,\ \phi(x_0, a_1) = 0,\ \phi(x_0, a_2) = 1,\ \phi(x_0, a_3) = 1.$$

Then, Table 3 provides three locally correct regression models ($\hat{f}_1$ to $\hat{f}_3$). More specifically, these example models succeed in preserving the relative value difference of the actions within each action cluster. In fact, we can see that $\Delta_q(x_0, a_0, a_1) = \Delta_{\hat{f}_1}(x_0, a_0, a_1) = \Delta_{\hat{f}_2}(x_0, a_0, a_1) = \Delta_{\hat{f}_3}(x_0, a_0, a_1) = 3$ and $\Delta_q(x_0, a_2, a_3) = \Delta_{\hat{f}_1}(x_0, a_2, a_3) = \Delta_{\hat{f}_2}(x_0, a_2, a_3) = \Delta_{\hat{f}_3}(x_0, a_2, a_3) = 1$ where $\phi(x_0, a_0) = \phi(x_0, a_1)$ and $\phi(x_0, a_2) = \phi(x_0, a_3)$.

C. Omitted Proofs

First, we generalize the formulation in the main content and will be based on action clustering given deterministic action embedding $e \in \mathcal{E}$ and context x, i.e., $\phi : \mathcal{X} \times \mathcal{E} \to \mathcal{C}$ to prove the theorems. Under this generalization, OffCEM is refined as

$$\hat{V}_{\mathrm{OffCEM}}(\pi; \mathcal{D}) := \frac{1}{n} \sum_{i=1}^{n} \left\{ w(x_i, \phi(x_i, e_{a_i}))(r_i - \hat{f}(x_i, a_i, e_{a_i})) + \hat{f}(x_i, \pi) \right\}, \tag{11}$$

where $\hat{f}(x, \pi) := \mathbb{E}_{\pi(a|x)}[\hat{f}(x, a, e_a)]$ and

$$w(x, \phi(x, e_a)) := \frac{\pi(\phi(x, e_a) \,|\, x)}{\pi_0(\phi(x, e_a) \,|\, x)} = \frac{\sum_{a' \in \mathcal{A}} \mathbb{I}\{\phi(x, e_a) = \phi(x, e_{a'})\} \pi(a' \,|\, x)}{\sum_{a' \in \mathcal{A}} \mathbb{I}\{\phi(x, e_a) = \phi(x, e_{a'})\} \pi_0(a' \,|\, x)}.$$

Moreover, under this setup, the local correctness assumption in Assumption 3.1 is generalized as follows.

Assumption C.1. (Generalized Local Correctness) A regression model $\hat{f}(\cdot, \cdot)$ and action clustering function $\phi(\cdot, \cdot)$ satisfy local correctness if the following holds true:

$$\Delta_q(x, a, b) = \Delta_{\hat{f}}(x, a, b),$$

for all $x \in \mathcal{X}$ and $a, b \in \mathcal{A}$ with $\phi(x, e_a) = \phi(x, e_b)$, where $\Delta_q(x, a, b) := q(x, a, e_a) - q(x, b, e_b)$ is the difference in the expected rewards between the actions a and b given context x, which we call the *relative value difference* of the actions.

Note that, given the fact: $\Delta_q(x, a, b) = \Delta_{\hat{f}}(x, a, b) \Rightarrow \Delta_{q,\hat{f}}(x, a) = \Delta_{q,\hat{f}}(x, b)$, the local correctness assumption can equivalently be described as:

A regression model $\hat{f}(\cdot, \cdot)$ and action clustering function $\phi(\cdot, \cdot)$ satisfy local correctness if there exists a cluster value function $g : \mathcal{X} \times \mathcal{C} \to \mathbb{R}$ that satisfies the following:

$$\Delta_{q,\hat{f}}(x, a) = g(x, \phi(x, e_a)) \left(\Rightarrow q(x, a, e_a) = g(x, \phi(x, e_a)) + \hat{f}(x, a, e_a) \right), \tag{12}$$

for all $x \in \mathcal{X}$ and $a \in \mathcal{A}$ where $\Delta_{q,\hat{f}}(x, a, e) := q(x, a, e) - \hat{f}(x, a, e)$ is an estimation error of the regression model $\hat{f}$ given context x, action a, and embedding e.

Note that it is straightforward to see that Assumption C.1 is reduced to a simpler version used in the main text (Assumption 3.1) when given is an action clustering function that depends directly on the action, i.e., $\phi : \mathcal{X} \times \mathcal{A} \to \mathcal{C}$.

C.1. Proof of Proposition 3.2

Proof. We can calculate the expectation of OffCEM under Assumption C.1 below.

$$
\begin{aligned}
&\mathbb{E}_{\mathcal{D}}[\hat{V}_{\mathrm{OffCEM}}(\pi;\mathcal{D})] \\
&= \mathbb{E}_{p(x)\pi_0(a|x)p(r|x,a,e_a)}\left[w(x,\phi(x,e_a))(r-\hat{f}(x,a,e_a))+\hat{f}(x,\pi)\right] \\
&= \mathbb{E}_{p(x)\pi_0(a|x)}\left[w(x,\phi(x,e_a))(q(x,a,e_a)-\hat{f}(x,a,e))+\hat{f}(x,\pi)\right] \\
&= \mathbb{E}_{p(x)\pi_0(a|x)}\left[w(x,\phi(x,e_a))\Delta_{q,\hat{f}}(x,a,e_a)+\hat{f}(x,\pi)\right] \\
&= \mathbb{E}_{p(x)}\left[\hat{f}(x,\pi)+\sum_{a\in\mathcal{A}}\pi_0(a\,|\,x)\frac{\pi(\phi(x,e_a)\,|\,x)}{\pi_0(\phi(x,e_a)\,|\,x)}\Delta_{q,\hat{f}}(x,a,e_a)\right] \\
&= \mathbb{E}_{p(x)}\left[\hat{f}(x,\pi)+\sum_{a\in\mathcal{A}}\pi_0(a\,|\,x)\frac{\pi(\phi(x,e_a)\,|\,x)}{\pi_0(\phi(x,e_a)\,|\,x)}g(x,\phi(x,e_a))\right] \quad \because \text{Eq. (12)} \\
&= \mathbb{E}_{p(x)}\left[\hat{f}(x,\pi)+\sum_{a\in\mathcal{A}}\pi_0(a\,|\,x)\sum_{c\in\mathcal{C}}\frac{\pi(c\,|\,x)}{\pi_0(c\,|\,x)}g(x,c)\mathbb{I}\{\phi(x,e_a)=c\}\right] \\
&= \mathbb{E}_{p(x)}\left[\hat{f}(x,\pi)+\sum_{c\in\mathcal{C}}\frac{\pi(c\,|\,x)}{\pi_0(c\,|\,x)}g(x,c)\sum_{a\in\mathcal{A}}\pi_0(a\,|\,x)\mathbb{I}\{\phi(x,e_a)=c\}\right] \\
&= \mathbb{E}_{p(x)}\left[\hat{f}(x,\pi)+\sum_{c\in\mathcal{C}}\frac{\pi(c\,|\,x)}{\pi_0(c\,|\,x)}g(x,c)\pi_0(c\,|\,x)\right] \quad \because \pi_0(c\,|\,x)=\sum_{a\in\mathcal{A}}\pi_0(a\,|\,x)\mathbb{I}\{\phi(x,e_a)=c\} \\
&= \mathbb{E}_{p(x)}\left[\hat{f}(x,\pi)+\sum_{c\in\mathcal{C}}\pi(c\,|\,x)g(x,c)\right] \\
&= \mathbb{E}_{p(x)}\left[\hat{f}(x,\pi)+\sum_{c\in\mathcal{C}}g(x,c)\sum_{a\in\mathcal{A}}\pi(a\,|\,x)\mathbb{I}\{\phi(x,e_a)=c\}\right] \quad \because \pi(c\,|\,x)=\sum_{a\in\mathcal{A}}\pi(a\,|\,x)\mathbb{I}\{\phi(x,e_a)=c\} \\
&= \mathbb{E}_{p(x)}\left[\hat{f}(x,\pi)+\sum_{a\in\mathcal{A}}\pi(a\,|\,x)\sum_{c\in\mathcal{C}}g(x,c)\mathbb{I}\{\phi(x,e_a)=c\}\right] \\
&= \mathbb{E}_{p(x)}\left[\sum_{a\in\mathcal{A}}\pi(a\,|\,x)\left\{g(x,\phi(x,e_a))+\hat{f}(x,a,e_a)\right\}\right] \\
&= \mathbb{E}_{p(x)\pi(a|x)}\left[q(x,a,e_a)\right] \quad \because \text{Eq. (12)} \\
&= V(\pi)
\end{aligned}
$$

and thus OffCEM is unbiased under Assumption C.1. □

C.2. Proof of Theorem 3.3

Proof. We first derive the bias of the general form of OffCEM below.[9]

$$
\begin{aligned}
&\mathrm{Bias}(\hat{V}_{\mathrm{OffCEM}}(\pi;\mathcal{D})) \\
&= \mathbb{E}_{p(x)\pi_0(a|x)p(e|x,a)p(r|x,a,e)}[w(x,\phi(x,e))(r-\hat{f}(x,a,e))+\hat{f}(x,\pi)]-V(\pi) \\
&= \mathbb{E}_{p(x)\pi_0(a|x)p(e|x,a)}[w(x,\phi(x,e))(q(x,a,e)-\hat{f}(x,a,e))+\hat{f}(x,\pi)]-\mathbb{E}_{p(x)\pi(a|x)p(e|x,a)}[q(x,a,e)]
\end{aligned}
$$

[9]Here, we work on a slightly more general case with stochastic action embeddings based on $p(e\,|\,x,a)$ as in Saito & Joachims (2022).

$$
\begin{aligned}
&= \mathbb{E}_{p(x)}\left[\sum_{a\in\mathcal{A}}\pi_0(a\,|\,x)\sum_{e\in\mathcal{E}}p(e\,|\,x,a)\Delta_{q,\hat{f}}(x,a,e)w(x,\phi(x,e))\right] \\
&\quad + \mathbb{E}_{p(x)}\left[\sum_{a\in\mathcal{A}}\sum_{e\in\mathcal{E}}\pi(a,e\,|\,x)\hat{f}(x,a,e)\right] - \mathbb{E}_{p(x)}\left[\sum_{a\in\mathcal{A}}\pi(a\,|\,x)\sum_{e\in\mathcal{E}}\frac{\pi_0\left(e\,|\,x\right)\pi_0(a\,|\,x,e)}{\pi_0(a\,|\,x)}q(x,a,e)\right] \\
&= \mathbb{E}_{p(x)}\left[\sum_{a\in\mathcal{A}}\pi_0(a\,|\,x)\sum_{e\in\mathcal{E}}\frac{\pi_0(e\,|\,x)\pi_0(a\,|\,x,e)}{\pi_0(a\,|\,x)}\Delta_{q,\hat{f}}(x,a,e)\sum_{c\in\mathcal{C}}w(x,c)\mathbb{I}\{\phi(x,e)=c\}\right] \\
&\quad + \mathbb{E}_{p(x)}\left[\sum_{a\in\mathcal{A}}\sum_{e\in\mathcal{E}}\pi_0(a,e\,|\,x)\frac{\pi(a,e\,|\,x)}{\pi_0(a,e\,|\,x)}\hat{f}(x,a,e)\right] - \mathbb{E}_{p(x)}\left[\sum_{e\in\mathcal{E}}\pi_0(e\,|\,x)\sum_{a\in\mathcal{A}}w(x,a)\pi_0(a\,|\,x,e)q(x,a,e)\right] \\
&= \mathbb{E}_{p(x)}\left[\sum_{c\in\mathcal{C}}\pi_0(c\,|\,x)w(x,c)\sum_{a\in\mathcal{A}}\sum_{e\in\mathcal{E}}\frac{\pi_0(e\,|\,x)\pi_0(a\,|\,x,e)\mathbb{I}\{\phi(x,e)=c\}}{\pi_0(c\,|\,x)}\Delta_{q,\hat{f}}(x,a,e)\right] \\
&\quad + \mathbb{E}_{p(x)}\left[\sum_{a\in\mathcal{A}}\sum_{e\in\mathcal{E}}\pi_0(a,e\,|\,x)\frac{\pi(a\,|\,x)p(e\,|\,x,a)}{\pi_0(a\,|\,x)p(e\,|\,x,a)}\hat{f}(x,a,e)\right] \\
&\quad - \mathbb{E}_{p(x)}\left[\sum_{e\in\mathcal{E}}\pi_0(e\,|\,x)\sum_{a\in\mathcal{A}}w(x,a)q(x,a,e)\frac{\sum_{c\in\mathcal{C}}\pi_0(c\,|\,x)\pi_0(a,e\,|\,x,c)}{\pi_0(e\,|\,x)}\right] \\
&= \mathbb{E}_{p(x)\pi_0(c|x)}\left[w(x,c)\sum_{a\in\mathcal{A}}\sum_{e\in\mathcal{E}}\pi_0(a,e\,|\,x,c)\Delta_{q,\hat{f}}(x,a,e)\right] \\
&\quad + \mathbb{E}_{p(x)\pi_0(c|x)}\left[\sum_{a\in\mathcal{A}}\sum_{e\in\mathcal{E}}w(x,a)\pi_0(a,e\,|\,x,c)\hat{f}(x,a,e)\right] \\
&\quad - \mathbb{E}_{p(x)}\left[\sum_{c\in\mathcal{C}}\pi_0(c\,|\,x)\sum_{a\in\mathcal{A}}\sum_{e\in\mathcal{E}}w(x,a)\pi_0(a,e\,|\,x,c)q(x,a,e)\right] \\
&= \mathbb{E}_{p(x)\pi_0(c|x)}\left[\sum_{a\in\mathcal{A}}\sum_{e\in\mathcal{E}}w(x,a)\pi_0(a,e\,|\,x,c)\sum_{a'\in\mathcal{A}}\sum_{e'\in\mathcal{E}}\pi_0(a',e'\,|\,x,c)\Delta_{q,\hat{f}}(x,a',e')\right] \\
&\quad - \mathbb{E}_{p(x)\pi_0(c|x)}\left[\sum_{a\in\mathcal{A}}\sum_{e\in\mathcal{E}}w(x,a)\pi_0(a,e\,|\,x,c)\Delta_{q,\hat{f}}(x,a,e)\right] \quad \because w(x,c)=\mathbb{E}_{\pi_0(a,e|x,c)}[w(x,a)] \\
&= \mathbb{E}_{p(x)\pi_0(c|x)}\left[\sum_{a\in\mathcal{A}}\sum_{e\in\mathcal{E}}w(x,a)\pi_0(a,e\,|\,x,c)\left(\Big(\sum_{a'\in\mathcal{A}}\sum_{e'\in\mathcal{E}}\pi_0(a',e'\,|\,x,c)\Delta_{q,\hat{f}}(x,a',e')\Big)-\Delta_{q,\hat{f}}(x,a,e)\right)\right] \qquad (13)
\end{aligned}
$$

where we used

$$
\begin{aligned}
w(x,c) &= \frac{\pi(c\,|\,x)}{\pi_0(c\,|\,x)} \\
&= \frac{1}{\pi_0(c\,|\,x)}\sum_{a\in\mathcal{A}}\sum_{e\in\mathcal{E}}\pi(a\,|\,x)p(e\,|\,x,a)\mathbb{I}\{\phi(x,e)=c\} \\
&= \frac{1}{\pi_0(c\,|\,x)}\sum_{a\in\mathcal{A}}\sum_{e\in\mathcal{E}}\pi(a\,|\,x)\frac{\pi_0(e\,|\,x)\pi_0(a\,|\,x,e)}{\pi_0(a\,|\,x)}\mathbb{I}\{\phi(x,e)=c\} \\
&= \sum_{a\in\mathcal{A}}\sum_{e\in\mathcal{E}}\frac{\pi(a\,|\,x)}{\pi_0(a\,|\,x)}\pi_0(a\,|\,x,e)\frac{\pi_0(e\,|\,x)\mathbb{I}\{\phi(x,e)=c\}}{\pi_0(c\,|\,x)} \\
&= \sum_{a\in\mathcal{A}}\sum_{e\in\mathcal{E}}w(x,a)\pi_0(a\,|\,x,e)p(e\,|\,x,c) \\
&= \mathbb{E}_{\pi_0(a,e|x,c)}[w(x,a)] \quad \because \pi_0(a\,|\,x,e)=\pi_0(a\,|\,x,e,c)
\end{aligned}
$$

By applying Lemma B.1 of Saito & Joachims (2022) to Eq. (13), we obtain the following expression of the bias.

$$\begin{aligned}&\mathrm{Bias}(\hat{V}_{\mathrm{OffCEM}}(\pi;\mathcal{D}))\\&=\mathbb{E}_{p(x)\pi_0(c|x)}\left[\sum_{(a,e)\neq(a',e')}\pi_0(a,e\,|\,x,c)\pi_0(a',e'\,|\,x,c)(\Delta_{q,\hat{f}}(x,a,e)-\Delta_{q,\hat{f}}(x,a',e'))(w(x,a')-w(x,a))\right],\end{aligned}\tag{14}$$

where we use $w(x,a)=\pi(a\,|\,x)/\pi_0(a\,|\,x)=\pi(a,e\,|\,x)/\pi_0(a,e\,|\,x)$.

Note that if we simplify the problem setting as in the main text, i.e., $\phi:\mathcal{X}\times\mathcal{A}\to\mathcal{C}$, Eq. (14) is reduced to

$$\begin{aligned}&\mathrm{Bias}(\hat{V}_{\mathrm{OffCEM}}(\pi;\mathcal{D}))\\&=\mathbb{E}_{p(x)\pi_0(c|x)}\left[\sum_{a<b}\pi_0(a\,|\,x,c)\pi_0(b\,|\,x,c)(\Delta q(x,a,b)-\Delta\hat{f}(x,a,b))(w(x,b)-w(x,a))\right]\\&=\mathbb{E}_{p(x)\pi_0(c|x)}\left[\sum_{a<b}\frac{\pi_0(a\,|\,x)\mathbb{I}\{\phi(x,a)=c\}}{\pi_0(c\,|\,x)}\frac{\pi_0(b\,|\,x)\mathbb{I}\{\phi(x,b)=c\}}{\pi_0(c\,|\,x)}(w(x,b)-w(x,a))(\Delta q(x,a,b)-\Delta\hat{f}(x,a,b))\right]\\&=\mathbb{E}_{p(x)\pi_0(c|x)}\left[w(x,c)\sum_{a<b:\phi(x,a)=\phi(x,b)=c}\pi_0(a\,|\,x,c)\pi_0(b\,|\,x,c)\left(\frac{\pi(b\,|\,x,c)}{\pi_0(b\,|\,x,c)}-\frac{\pi(a\,|\,x,c)}{\pi_0(a\,|\,x,c)}\right)(\Delta q(x,a,b)-\Delta\hat{f}(x,a,b))\right]\\&=\mathbb{E}_{p(x)\pi(c|x)}\left[\sum_{a<b:\phi(x,a)=\phi(x,b)=c}\pi_0(a\,|\,x,c)\pi_0(b\,|\,x,c)\left(\frac{\pi(b\,|\,x,c)}{\pi_0(b\,|\,x,c)}-\frac{\pi(a\,|\,x,c)}{\pi_0(a\,|\,x,c)}\right)(\Delta q(x,a,b)-\Delta\hat{f}(x,a,b))\right],\end{aligned}$$

where we use $w(x,a)=\frac{\pi(a|x)}{\pi_0(a|x)}=\frac{\pi(a|x,c)\pi(c|x)}{\pi_0(a|x,c)\pi_0(c|x)}=\frac{\pi(a|x,c)}{\pi_0(a|x,c)}w(x,c)$ for $\phi(x,a)=c$.

Theorem 3.3 implies that OffCEM is unbiased under Assumption 3.1, as $\Delta q(x,a,b)-\Delta\hat{f}(x,a,b)=0$ for all x,a,b with $\phi(x,a)=\phi(x,b)=c$ for a given clustering function $\phi(\cdot,\cdot)$. □

C.3. Proof of Proposition 3.4

Proof. We apply the law of total variance several times to obtain the variance of OffCEM.

$$\begin{aligned}&\mathbb{V}_{p(x)\pi_0(a|x)p(r|x,a,e_a)}\left[w(x,\phi(x,e_a))(r-\hat{f}(x,a,e_a))+\hat{f}(x,\pi)\right]\\&=\mathbb{E}_{p(x)\pi_0(a|x)}\left[\mathbb{V}_{p(r|x,a,e_a)}\left[w(x,\phi(x,e_a))(r-\hat{f}(x,a,e_a))+\hat{f}(x,\pi)\right]\right]\\&\quad+\mathbb{V}_{p(x)\pi_0(a|x)}\left[\mathbb{E}_{p(r|x,a,e_a)}\left[w(x,\phi(x,e_a))(r-\hat{f}(x,a,e_a))+\hat{f}(x,\pi)\right]\right]\\&=\mathbb{E}_{p(x)\pi_0(a|x)}\left[w(x,\phi(x,e_a))^2\sigma^2(x,a,e_a)\right]+\mathbb{V}_{p(x)\pi_0(a|x)}\left[w(x,\phi(x,e_a))(q(x,a,e_a)-\hat{f}(x,a,e_a))+\hat{f}(x,\pi)\right]\\&=\mathbb{E}_{p(x)\pi_0(a|x)}\left[w(x,\phi(x,e_a))^2\sigma^2(x,a,e_a)\right]+\mathbb{V}_{p(x)\pi_0(a|x)}\left[w(x,\phi(x,e_a))\Delta_{q,\hat{f}}(x,a)+\hat{f}(x,\pi)\right]\\&=\mathbb{E}_{p(x)\pi_0(a|x)}\left[w(x,\phi(x,e_a))^2\sigma^2(x,a,e_a)\right]+\mathbb{E}_{p(x)}\left[\mathbb{V}_{\pi_0(a|x)}\left[w(x,\phi(x,e_a))\Delta_{q,\hat{f}}(x,a)+\hat{f}(x,\pi)\right]\right]\\&\quad+\mathbb{V}_{p(x)}\left[\mathbb{E}_{\pi_0(a|x)}\left[w(x,\phi(x,e_a))g(x,\phi(x,e_a))+\hat{f}(x,\pi)\right]\right]\quad\because\text{ Assumption C.1}\\&=\mathbb{E}_{p(x)\pi_0(a|x)}\left[w(x,\phi(x,e_a))^2\sigma^2(x,a,e_a)\right]+\mathbb{E}_{p(x)}\left[\mathbb{V}_{\pi_0(a|x)}\left[w(x,\phi(x,e_a))\Delta_{q,\hat{f}}(x,a)\right]\right]+\mathbb{V}_{p(x)}\left[\mathbb{E}_{\pi(a|x)}[q(x,a,e_a)]\right]\end{aligned}\tag{15}$$

where $\Delta_{q,\hat{f}}(x,a,e)=q(x,a,e)-\hat{f}(x,a,e)$ and $\mathbb{E}_{\pi_0(a|x)}[w(x,\phi(x,e_a))g(x,\phi(x,e_a))+\hat{f}(x,\pi)]=\mathbb{E}_{\pi(a|x)}[q(x,a,e_a)]$ as in Section C.1 under Assumption C.1.

If we simplify the problem setting as in the main text, i.e., $\phi:\mathcal{X}\times\mathcal{A}\to\mathcal{C}$, Eq. (15) is reduced to the following.

$$\begin{aligned}&\mathbb{V}_{p(x)\pi_0(a|x)p(r|x,a)}\left[w(x,\phi(x,a))(r-\hat{f}(x,a))+\hat{f}(x,\pi)\right]\\&=\mathbb{E}_{p(x)\pi_0(a|x)}\left[w(x,\phi(x,a))^2\sigma^2(x,a)\right]+\mathbb{E}_{p(x)}\left[\mathbb{V}_{\pi_0(a|x)}\left[w(x,\phi(x,a))\Delta_{q,\hat{f}}(x,a)\right]\right]+\mathbb{V}_{p(x)}\left[\mathbb{E}_{\pi(a|x)}[q(x,a)]\right].\end{aligned}$$

□

D. Detailed Experiment Settings and Results

D.1. Baseline Estimators

Below, we define and describe the baseline estimators compared in our experiments in detail.

Direct Method (DM). DM is defined as follows.

$$\hat{V}_{\mathrm{DM}}(\pi;\mathcal{D},\hat{q}) := \frac{1}{n}\sum_{i=1}^{n}\hat{q}(x_i,\pi) = \frac{1}{n}\sum_{i=1}^{n}\sum_{a\in\mathcal{A}}\pi(a\,|\,x_i)\hat{q}(x_i,a),$$

where $\hat{q}(x,a)$ estimates $q(x,a)$ based on logged bandit data. The accuracy of DM depends on the quality of $\hat{q}(x,a)$. If $\hat{q}(x,a)$ is accurate, so is DM. However, if $\hat{q}(x,a)$ fails to estimate the expected reward accurately, the final estimator is no longer consistent. As discussed in Appendix A, the misspecification issue is challenging, as it cannot be easily detected from observed data (Farajtabar et al., 2018; Voloshin et al., 2019). This is why DM is often described as a high bias estimator.

Inverse Propensity Score (IPS). IPS is defined as follows.

$$\hat{V}_{\mathrm{IPS}}(\pi;\mathcal{D}) := \frac{1}{n}\sum_{i=1}^{n}w(x_i,a_i)r_i$$

where $w(x,a) := \pi(a\,|\,x)/\pi_0(a\,|\,x)$ is the vanilla importance weight. IPS estimates the value of the target policy by simply re-weighting the observed rewards and is unbiased under common support (Assumption 2.1). Its critical issue, however, is that the variance can be excessive in large action spaces due to the fact that the importance weight inflates.

Doubly Robust (DR) (Dudík et al., 2014). DR is defined as follows.

$$\hat{V}_{\mathrm{DR}}(\pi;\mathcal{D},\hat{q}) := \frac{1}{n}\sum_{i=1}^{n}\left\{w(x_i,a_i)(r_i-\hat{q}(x_i,a_i))+\hat{q}(x_i,\pi)\right\},$$

which combines DM and IPS in a way that reduces the variance. More specifically, DR utilizes the estimated reward function $\hat{q}$ as a control variate. If the expected reward is correctly specified, DR is *semiparametric efficient* meaning that it achieves the minimum possible asymptotic variance among regular estimators (Narita et al., 2019). A problem is that, if the expected reward is misspecified, this estimator can have a larger asymptotic MSE compared to IPS. Moreover, as shown in Eq. (1) and empirically verified by Saito & Joachims (2022), DR can have a very large variance in the presence of many actions.

Marginalized Inverse Propensity Score (MIPS) (Saito & Joachims, 2022). MIPS is defined as follows.

$$\hat{V}_{\mathrm{MIPS}}(\pi;\mathcal{D}) := \frac{1}{n}\sum_{i=1}^{n}w(x_i,e_i)r_i,$$

where $w(x,e) := \pi(e\,|\,x)/\pi_0(e\,|\,x)$ is the marginal importance weight defined based on the marginal embedding distribution induced by the target and logging policies. MIPS is unbiased under common embedding support (Assumption 2.2) and no direct effect (Assumption 2.3). It was also shown by Saito & Joachims (2022) that MIPS provides the following variance reduction in comparison to IPS:

$$n\left(\mathbb{V}_{\mathcal{D}}[\hat{V}_{\mathrm{IPS}}(\pi;\mathcal{D})]-\mathbb{V}_{\mathcal{D}}[\hat{V}_{\mathrm{MIPS}}(\pi;\mathcal{D})]\right) = \mathbb{E}_{p(x)\pi_0(e|x)}\left[\mathbb{E}_{p(r|x,e)}\left[r^2\right]\mathbb{V}_{\pi_0(a|x,e)}\left[w(x,a)\right]\right], \tag{16}$$

which is always non-negative and can be substantial when the variance of the vanilla importance weights is large, which is likely for large action spaces. However, it is still possible that the variance of MIPS can be extremely large when action embeddings are high-dimensional and fine-grained. Specifically, the variance of MIPS (under no direct effect from Assumption 2.3) is given as

$$n\mathbb{V}_{\mathcal{D}}\left[\hat{V}_{\mathrm{MIPS}}(\pi;\mathcal{D})\right] = \mathbb{E}_{p(x)\pi_0(e|x)}[w(x,e)^2\sigma^2(x,a)] + \mathbb{E}_{p(x)}\left[\mathbb{V}_{\pi_0(e|x)}[w(x,e)\Delta_{q,\hat{q}}(x,e)]\right] + \mathbb{V}_{p(x)}\left[\mathbb{E}_{\pi(e|x)}[q(x,e)]\right],$$

which may become almost identical to the variance of IPS when $w(x,e) \approx w(x,a)$ due to high-dimensional and fine-grained embeddings. To reduce the variance of MIPS, Saito & Joachims (2022) describe a heuristic procedure to perform action feature selection based on logged data, but this produces bias by violating the no direct effect assumption. Thus, the critical bias-variance dilemma still remains in this estimator, particularly in the presence of high-dimensional action embeddings.

Definitions of the estimators compared in Figure 3 of Section 4.1. The following defines the estimators compared in Figure 3 (ablation study) of Section 4.1.

$$\hat{V}_{+\text{clustering}}(\pi;\mathcal{D}) := \frac{1}{n}\sum_{i=1}^{n} w(x_i,\phi(x_i,a_i))r_i,$$

$$\hat{V}_{+\text{regression model}}(\pi;\mathcal{D}) := \frac{1}{n}\sum_{i=1}^{n} \left\{w(x_i,e_i)(r_i-\hat{q}(x_i,a_i))+\hat{q}(x_i,\pi)\right\},$$

$$\hat{V}_{\text{clustering+1step reg}}(\pi;\mathcal{D}) := \frac{1}{n}\sum_{i=1}^{n} \left\{w(x_i,\phi(x_i,a_i))(r_i-\hat{q}(x_i,a_i))+\hat{q}(x_i,\pi)\right\},$$

$$\hat{V}_{\text{clustering+2step reg}}(\pi;\mathcal{D}) := \frac{1}{n}\sum_{i=1}^{n} \left\{w(x_i,\phi(x_i,a_i))(r_i-\hat{f}(x_i,a_i))+\hat{f}(x_i,\pi)\right\},$$

where $\hat{f}(x,a) = \hat{g}_\psi(x,\phi(x,a)) + \hat{h}_\theta(x,a)$ is obtained by performing the two-step regression procedure described in Section 3.2 while $\hat{q}(x,a)$ is obtained by simply performing a one-step reward regression as in Eq. (7).

$\hat{V}_{+\text{clustering}}$ uses cluster importance weighting instead of vanilla and marginal importance weighting and thus reduces the variance from IPS and MIPS. However, it may produce a large bias as it does not deal with the residual effect at all. $\hat{V}_{+\text{regression model}}$ uses a regression model $\hat{q}$ and somewhat reduces the variance of MIPS, but it may still produce a very large variance as it depends on the marginal importance weight. $\hat{V}_{\text{clustering+1step reg}}$ is a simple version of our proposed estimator combining cluster importance weighting and a regression model $\hat{q}(x,a)$, but it does not perform the two-step regression procedure and thus may not be optimal in terms of bias. In contrast, $\hat{V}_{\text{clustering+2step reg}}$ fully leverages the reward function decomposition of Eq. (2) and thus is the ideal version of our proposed estimator.

D.2. Additional Synthetic Experiment Setup and Results

D.2.1. Additional Experiment Setup

This section describes how we defined the synthetic reward function in detail. Recall that, in our synthetic experiments, we synthesized the expected reward function as

$$q(x,a) = g(x,c_a) + h_{c_a}(x,a),$$

Specifically, we used the following functions as $g(\cdot,\cdot)$ (cluster effect) and $h_\cdot(\cdot,\cdot)$ (residual effect), respectively.

$$\begin{aligned} g(x,c_a) &= g_{base}(x,\text{one_hot}_{c_a}) + u_1\mathbb{I}\{(\sum_{d=1}^{3} x_d) < 1.5\} \\ &\quad + u_2\mathbb{I}\{(\sum_{d=3}^{8} x_d) < -0.5\} + u_3\mathbb{I}\{(\sum_{d=2}^{3} x_d) > 3.0\} + u_4\mathbb{I}\{(\sum_{d=5}^{10} x_d) < 1.0\}, \\ h_{c_a}(x,a) &= x^\top M_{c_a}\text{one_hot}_a + \theta_{x,c_a}^\top x + \theta_{a,c_a}^\top \text{one_hot}_a, \end{aligned}$$

where one_hot_c is the one-hot encoding of the action cluster c and x_d is the d-th dimension of the context vector x. We use **obp.dataset.polynomial_reward_function** as $g_{base}(\cdot,\cdot)$ and $u_1,\ldots,u_4$ are sampled from a uniform distribution with range $[-3,3]$. M_{c_a}, θ_{x,c_a}, and θ_{a,c_a} are parameter matrices or vectors sampled from a uniform distribution with range $[-1,1]$ separately for each given action cluster c_a.

D.2.2. Additional Results

This section reports and discusses additional synthetic experiment results regarding varying levels of noise on the rewards, varying target policies, and varying numbers of action clusters. In particular, we demonstrate that OffCEM works particularly

better than other baselines when the reward is noisy, and the target and logging policies differ greatly. We also observe that OffCEM is appealing in particular when there is a fewer number of action clusters. The following also reports and discusses the bias-variance decomposition of the results shown in the main text, providing additional insights.

How does OffCEM perform with varying noise levels, target policies, numbers of action embedding dimensions, and numbers of clusters? First, Figure 5 evaluates how the level of noise on the rewards affects the comparison of the estimators. To this end, we vary the noise level $\sigma \in \{1.0, 2.0, 3.0, 4.0, 5.0\}$, where σ is the standard deviation of Gaussian reward noise, i.e., $r \sim \mathcal{N}(q(x, a), \sigma^2)$. We can see from the figures that the variance of IPS, DR, and MIPS grows with increased noise as expected, and OffCEM performs increasingly better than the baselines with larger noise levels. Note that DM is relatively more robust to the increased noise compared to IPS, DR, and MIPS, but it is highly biased in all situations, and our OffCEM performs much better than DM in a range of reward noise settings.

Second, Figure 6 compares the estimators with a range of target policies ($\epsilon \in \{0.0, 0.2, 0.4, 0.6, 0.8, 1.0\}$ in Eq. (10)). Note that a larger value of ϵ introduces a larger entropy for the target policy, making it closer to the logging policy given that we use $\beta = -0.1$ as default (an extreme case with $\epsilon = 1.0$ produces a uniform random target policy). Figure 6 shows that all estimators perform worse for smaller ϵ as expected, where OPE becomes harder in general. IPS, DR, and MIPS perform worse as their variance increases with decreasing ϵ, while DM performs worse as it produces a larger bias. The variance of OffCEM also increases ever so slightly with decreasing ϵ, but it is often much smaller than those of the baselines. In particular, our estimator becomes increasingly effective against the baseline estimators given a near-deterministic target policy (small ϵ), which is highly prevalent in real-world scenarios given that it is often easier-to-implement than stochastic policies and that the optimal policy is always deterministic (except for the case with non-unique optimal actions).

Finally, Figure 7 shows the estimators' performance when we vary the number of action clusters from 10 to 200. We can see from the figure that the variance of OffCEM becomes larger with increasing number of clusters, but it is generally smaller than those of IPS, DR, and MIPS. When there are many action clusters underlying the data-generating process (e.g., $|\mathcal{C}| = 100, 200$), however, we observe that the improvement provided by our estimator becomes slightly smaller due to increased bias. Nonetheless, there is no reason to avoid using OffCEM because the baseline estimators do not outperform ours in any of the experiment settings.

Discussion on the bias-variance decomposition of the synthetic results. Next, Figures 8 to 13 report and discuss the bias-variance decomposition of the synthetic results reported in the main text. First, we can see in Figure 8 that the bias of IPS of MIPS are very small as expected while their variance become extremely large especially for small sample sizes. We can also see that DR provides some reduction in MSE due to reduced variance over IPS and MIPS, however, the variance of our estimator is even smaller and is similar to that of DM while its bias is similar to that of DR, which is very small. We can also confirm from the figure that DM remains highly biased even with increased sample sizes. Figure 9 reports the bias-variance decomposition of the ablation study with varying sample sizes. In particular, it suggests that using only cluster importance weighting (**"+clustering"**) reduces the variance of MIPS while it produces a large bias as it ignores the residual effect. In contrast, using only a regression model provides some variance reduction while not producing a large bias, but there is still much room for improvement in terms of variance due to its use of marginal importance weighting. Simply combining these techniques (**"+clustering and one-step reg"**) is effective enough to outperform MIPS in a rage of sample sizes. However, we also observe that performing two-step regression is beneficial in further reducing the bias. Next, Figure 10 demonstrates that the variance of IPS, DR, and MIPS become larger for larger numbers of actions while that of OffCEM is much more robust, contributing to its greatly reduced MSEs in large action spaces. In Figure 12, we observe that the bias of IPS, DR, and MIPS become larger when there exists a larger number of unsupported actions as expected while the bias of OffCEM remains very small by separately accounting for the cluster and residual effects. More specifically, it can unbiasedly estimate the cluster effect and dealing with the residual effect to some extent via the model-based estimation. Thus, the small bias of OffCEM is the main reason for its much better MSE than those of IPS, DR, and MIPS when there are many unsupported actions. Note that the variance of IPS and DR decrease with increased unsupported actions, because the deficiency between logging and target policies become relatively small in that setup. Note also that we can clearly see the benefit of combining the two key techniques (cluster importance weighting and regression model) and that of using two-step regression under varying numbers of actions and numbers of unsupported actions in Figures 11 and 13.

D.3. Additional Real-World Experiment Setup

Following previous studies (Farajtabar et al., 2018; Dudík et al., 2014; Wang et al., 2017; Kallus et al., 2021; Su et al., 2020a; Saito et al., 2021b), we transform the extreme classification datasets (namely EUR-Lex 4K and Wiki10-31K) to contextual bandit feedback data. In a classification dataset $\{(x_i, a_i)\}_{i=1}^n$, we have some feature vector $x_i \in \mathcal{X}$ and ground-truth label $a_i \in \mathcal{A}$, which will be considered an action. Since the original datasets have very high-dimensional bag-of-words features, we apply feature dimension reduction based on PCA implemented in scikit-learn (Pedregosa et al., 2011). We also drop the labels/actions that have less than 1 (EUR-Lex 4K) or 9 (Wiki10-31K) positive labels during dataset pre-processing.

We consider stochastic binary rewards where we first define the base reward function as

$$\tilde{q}(x, a) = \begin{cases} 1 - \eta_a & \text{if } a \text{ is a positive label} \\ \eta_a - 1 & \text{otherwise} \end{cases} \tag{17}$$

where η_a is a noise parameter sampled separately for each action a from a uniform distribution with range $[0, 0.2]$. Then, for each data, we sample the reward from a Bernoulli distribution with mean $q(x, a) = \sigma(\tilde{q}(x, a))$ where $\sigma(z) = (1 + \exp(-z))^{-1}$ is the sigmoid function.

We define the logging policy π_0 by applying the softmax function to an estimated reward function $\hat{q}(x, a)$ as

$$\pi_0(a \,|\, x) = \frac{\exp(\beta \cdot \hat{q}(x, a))}{\sum_{a' \in \mathcal{A}} \exp(\beta \cdot \hat{q}(x, a'))}, \tag{18}$$

where we use $\beta = 30$ for both datasets. We obtain $\hat{q}(x, a)$ by performing a ridge regression with full-information labels. Note that we use the test data recorded in the original datasets for obtaining a logging policy while we use the training data for performing OPE to make them independent. We also define the target policy π as follows.

$$\pi(a \,|\, x) = (1 - \epsilon) \cdot \mathbb{I}\big\{a = \arg\max_{a' \in \mathcal{A}} q(x, a')\big\} + \frac{\epsilon}{|\mathcal{A}|},$$

where we set $\epsilon = 0.05$ in the real-world experiment.

To summarize, we first define the expected reward $q(x, a)$ as in Eq. (17) on the training dataset. We then sample discrete action a from π_0 based on Eq. (18). Finally, we sample the reward from a normal distribution with mean $q(x, a)$ and standard deviation $\sigma = 1$. Iterating this n times generates logged bandit data $\mathcal{D}$ with n independent copies of (x, a, r).

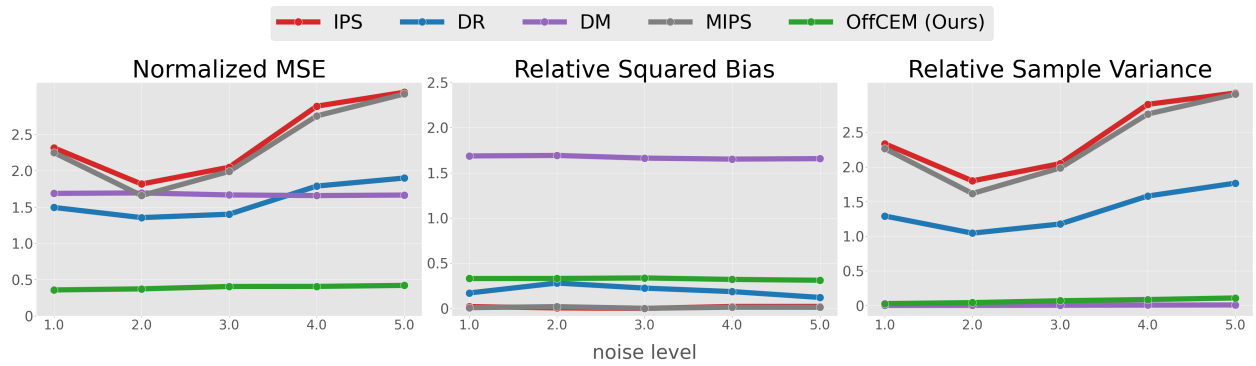

Figure 5. Comparison of the estimators' MSE (normalized by the true value $V(\pi)$) with **varying noise levels** (σ).

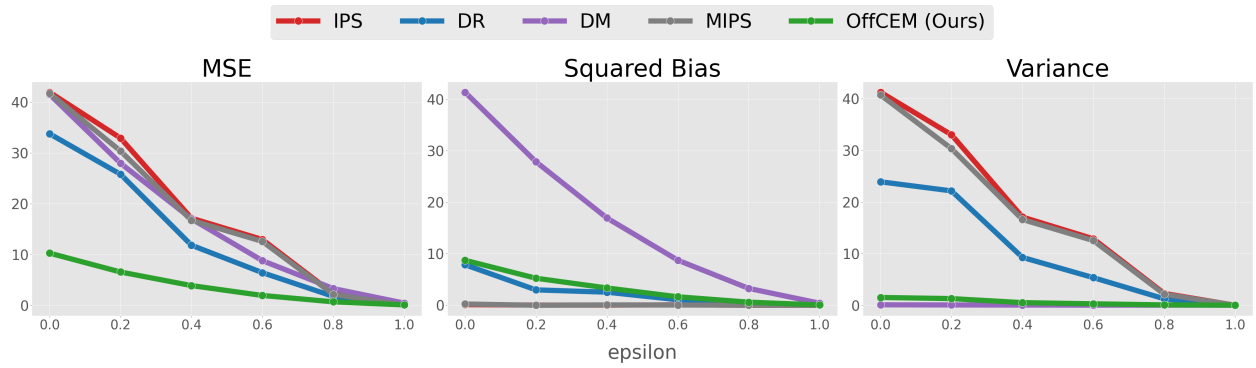

Figure 6. Comparison of the estimators' MSE with **varying target policies** (ϵ).

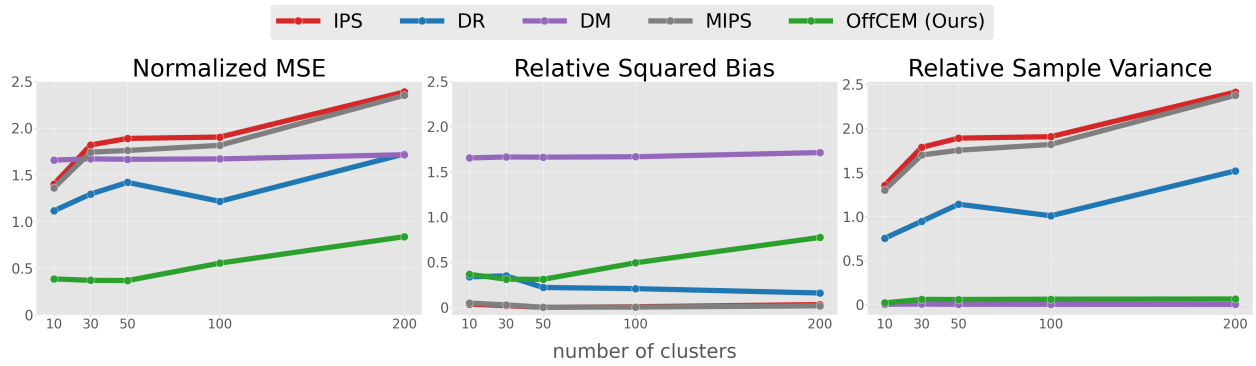

Figure 7. Comparison of the estimators' MSE (normalized by the true value $V(\pi)$) with **varying numbers of clusters** ($|\mathcal{C}|$).

Note: We set $|\mathcal{A}| = 1,000$, $|\mathcal{C}| = 50$, $\epsilon = 0.2$, $\beta = -0.1$, and $\sigma = 3.0$ as default experiment parameters. The results are averaged over 300 different sets of synthetic logged data replicated with different random seeds. The shaded regions in the MSE plots represent the 95% confidence intervals estimated with bootstrap.

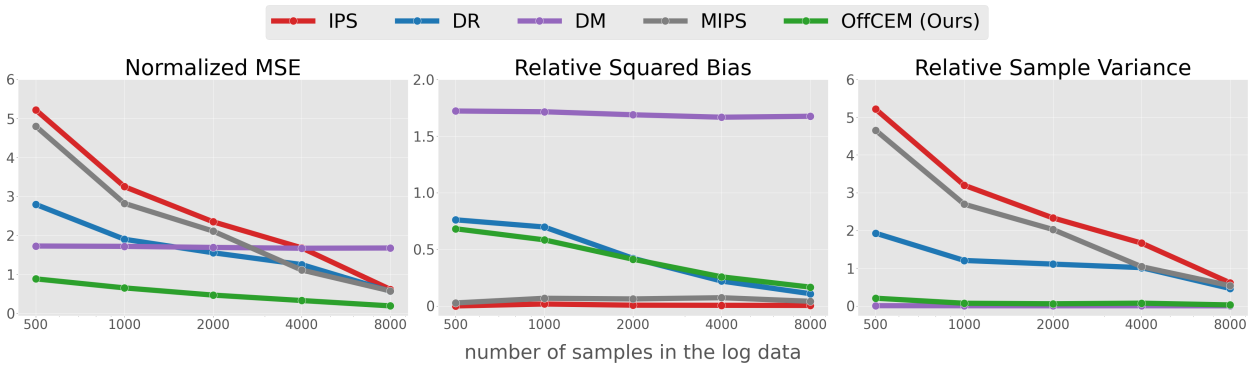

Figure 8. MSE, Squared Bias, and Variance with **varying sample sizes** (n).

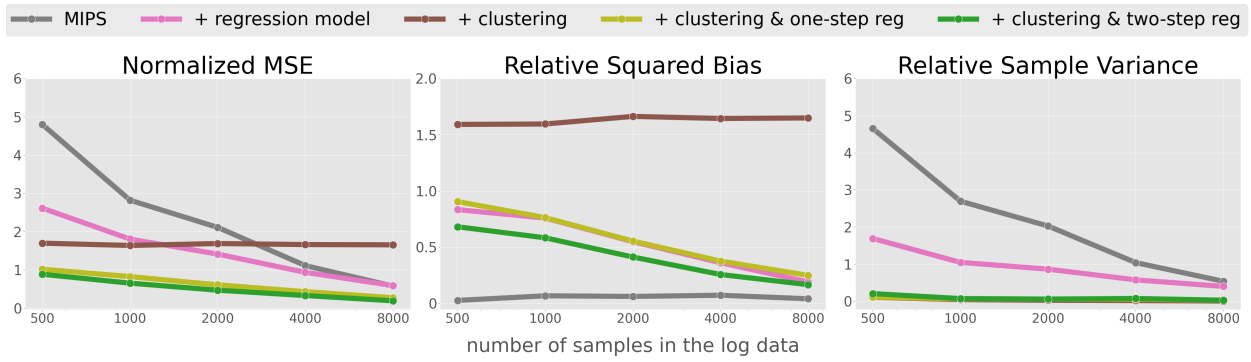

Figure 9. MSE, Squared Bias, and Variance (ablation) with **varying sample sizes** (n).

Note: We set $|\mathcal{A}| = 1,000$, $|\mathcal{C}| = 50$, $\epsilon = 0.2$, $\beta = -0.1$, and $\sigma = 3.0$ as default experiment parameters. The results are averaged over 300 different sets of synthetic logged data replicated with different random seeds. The shaded regions in the MSE plots represent the 95% confidence intervals estimated with bootstrap.

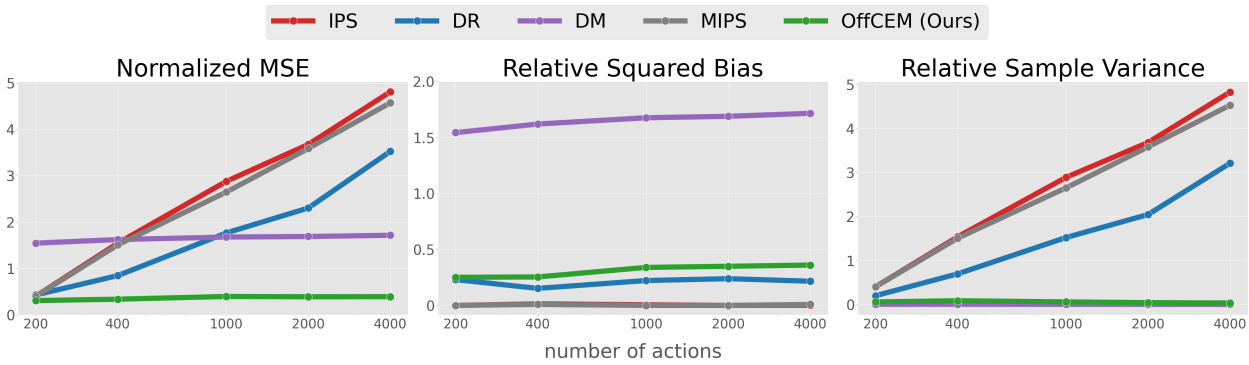

Figure 10. MSE, Squared Bias, and Variance with **varying numbers of actions** ($|\mathcal{A}|$).

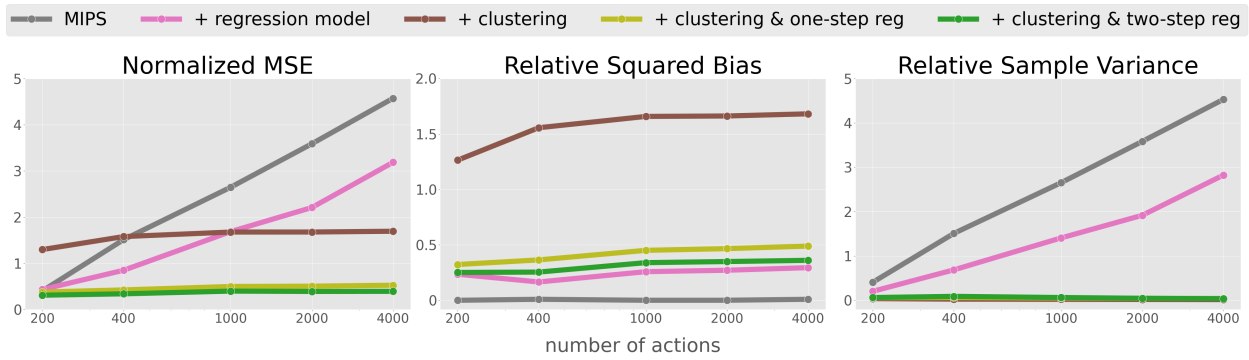

Figure 11. MSE, Squared Bias, and Variance (ablation) with **varying numbers of actions** ($|\mathcal{A}|$).

Note: We set $n = 3,000$, $|\mathcal{C}| = 50$, $\epsilon = 0.2$, $\beta = -0.1$, and $\sigma = 3.0$ as default experiment parameters. The results are averaged over 300 different sets of synthetic logged data replicated with different random seeds. The shaded regions in the MSE plots represent the 95% confidence intervals estimated with bootstrap.

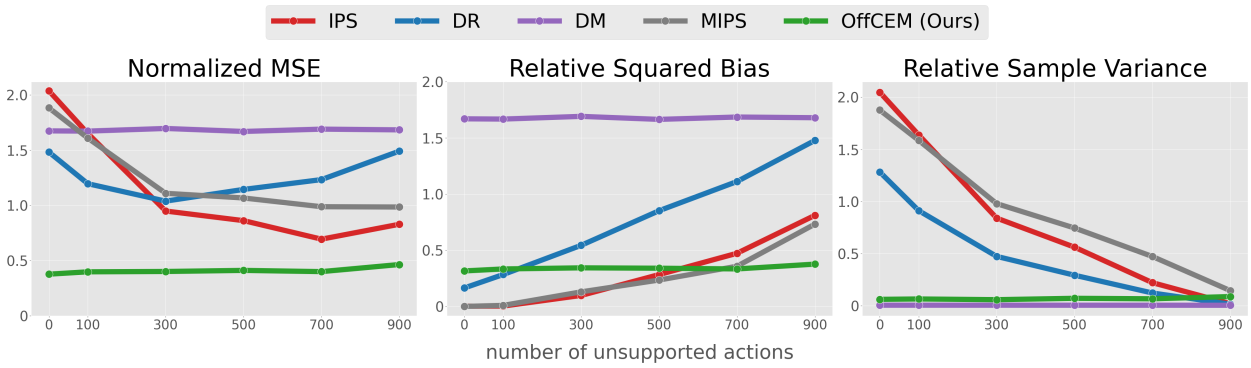

Figure 12. MSE, Squared Bias, and Variance with **varying numbers of unsupported actions**.

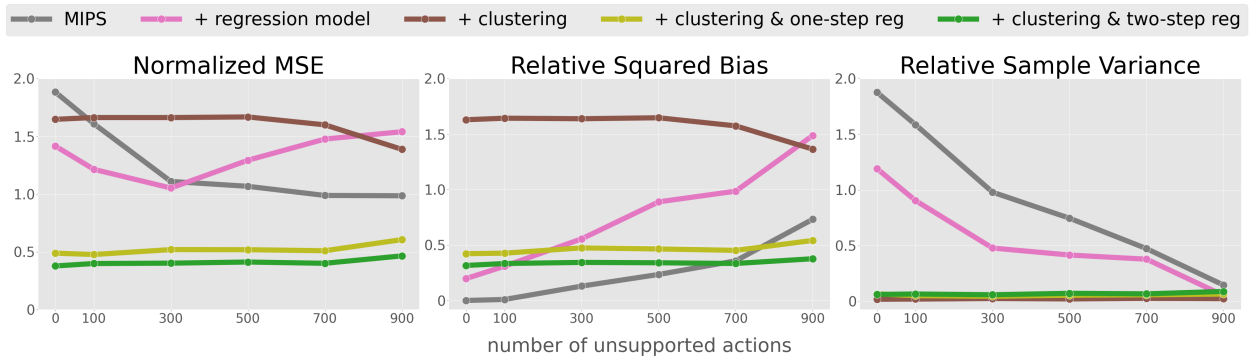

Figure 13. MSE, Squared Bias, and Variance (ablation) with **varying numbers of unsupported actions**.

Note: We set $n = 3,000$, $|\mathcal{A}| = 1,000$, $\epsilon = 0.2$, $\beta = -0.1$, and $\sigma = 3.0$ as default experiment parameters. The results are averaged over 300 different sets of synthetic logged data replicated with different random seeds. The shaded regions in the MSE plots represent the 95% confidence intervals estimated with bootstrap.