Statistical Science
2014, Vol. 29, No. 4, 485–511
DOI: 10.1214/14-STS500

Doubly Robust Policy Evaluation and Optimization[1]

Miroslav Dudík, Dumitru Erhan, John Langford and Lihong Li

Abstract. We study sequential decision making in environments where rewards are only partially observed, but can be modeled as a function of observed contexts and the chosen action by the decision maker. This setting, known as contextual bandits, encompasses a wide variety of applications such as health care, content recommendation and Internet advertising. A central task is evaluation of a new policy given historic data consisting of contexts, actions and received rewards. The key challenge is that the past data typically does not faithfully represent proportions of actions taken by a new policy. Previous approaches rely either on models of rewards or models of the past policy. The former are plagued by a large bias whereas the latter have a large variance.

In this work, we leverage the strengths and overcome the weaknesses of the two approaches by applying the *doubly robust* estimation technique to the problems of policy evaluation and optimization. We prove that this approach yields accurate value estimates when we have *either* a good (but not necessarily consistent) model of rewards *or* a good (but not necessarily consistent) model of past policy. Extensive empirical comparison demonstrates that the doubly robust estimation uniformly improves over existing techniques, achieving both lower variance in value estimation and better policies. As such, we expect the doubly robust approach to become common practice in policy evaluation and optimization.

Key words and phrases: Contextual bandits, doubly robust estimators, causal inference.

1. INTRODUCTION

Contextual bandits (Auer et al., 2002/03; Langford and Zhang, 2008), sometimes known as associative reinforcement learning (Barto and Anandan, 1985), are a natural generalization of the classic multiarmed bandits introduced by Robbins (1952). In a contextual bandit problem, the decision maker observes contextual information, based on which an action is chosen out of a set of candidates; in return, a numerical "reward" signal is observed for the *chosen* action, but not for others. The process repeats for multiple steps, and the goal of the decision maker is to maximize the total rewards in this process. Usually, contexts observed by the decision maker provide useful information to infer the expected reward of each action, thus allowing greater rewards to be accumulated, compared to standard multi-armed bandits, which take no account of the context.

Many problems in practice can be modeled by contextual bandits. For example, in one type of Internet advertising, the decision maker (such as a website) dynamically selects which ad to display to a user who visits the page, and receives a payment from the advertiser if the user clicks on the ad (e.g., Chapelle and

Miroslav Dudík is Senior Researcher and John Langford is Principal Researcher, Microsoft Research, New York, New York, USA (e-mail: mdudik@microsoft.com; jcl@microsoft.com). Dumitru Erhan is Senior Software Engineer, Google Inc., Mountain View, California, USA (e-mail: dumitru@google.com). Lihong Li is Researcher, Microsoft Research, Redmond, Washington, USA (e-mail: lihongli@microsoft.com).

[1] Parts of this paper were presented at the 28th International Conference on Machine Learning (Dudík, Langford and Li, 2011), and the 28th Conference on Uncertainty in Artificial Intelligence (Dudík et al., 2012).

Li, 2012). In this case, the context can be the user's geographical information, the action is the displayed ad and the reward is the payment. Importantly, we find only whether a user clicked on the presented ad, but receive no information about the ads that were not presented.

Another example is content recommendation on Web portals (Agarwal et al., 2013). Here, the decision maker (the web portal) selects, for each user visit, what content (e.g., news, images, videos and music) to display on the page. A natural objective is to "personalize" the recommendations, so that the number of clicks is maximized (Li et al., 2010). In this case, the context is the user's interests in different topics, either self-reported by the user or inferred from the user browsing history; the action is the recommended item; the reward can be defined as 1 if the user clicks on an item, and 0 otherwise.

Similarly, in health care, we only find out the clinical outcome (the reward) of a patient who received a treatment (action), but not the outcomes for alternative treatments. In general, the treatment strategy may depend on the context of the patient such as her health level and treatment history. Therefore, contextual bandits can also be a natural model to describe personalized treatments.

The behavior of a decision maker in contextual bandits can be described as a *policy*, to be defined precisely in the next sections. Roughly speaking, a policy is a function that maps the decision maker's past observations and the contextual information to a distribution over the actions. This paper considers the *offline* version of contextual bandits: we assume access to historical data, but no ability to gather new data (Langford, Strehl and Wortman, 2008; Strehl et al., 2011). There are two related tasks that arise in this setting: *policy evaluation* and *policy optimization*. The goal of policy evaluation is to estimate the expected total reward of a *given* policy. The goal of policy optimization is to obtain a policy that (approximately) maximizes expected total rewards. The focus of this paper is on policy evaluation, but as we will see in the experiments, the ideas can also be applied to policy optimization. The offline version of contextual bandits is important in practice. For instance, it allows a website to estimate, from historical log data, how much gain in revenue can be achieved by changing the ad-selection policy to a new one (Bottou et al., 2013). Therefore, the website does not have to experiment on *real* users to test a new policy, which can be very expensive and time-consuming. Finally, we note that this problem is a special case of *off-policy* reinforcement learning (Precup, Sutton and Singh, 2000).

Two kinds of approaches address offline policy evaluation. The first, called the *direct method* (DM), estimates the reward function from given data and uses this estimate in place of actual reward to evaluate the policy value on a set of contexts. The second kind, called *inverse propensity score* (IPS) (Horvitz and Thompson, 1952), uses importance weighting to correct for the incorrect proportions of actions in the historic data. The first approach requires an accurate model of rewards, whereas the second approach requires an accurate model of the past policy. In general, it might be difficult to accurately model rewards, so the first assumption can be too restrictive. On the other hand, in many applications, such as advertising, Web search and content recommendation, the decision maker has substantial, and possibly perfect, knowledge of the past policy, so the second approach can be applied. However, it often suffers from large variance, especially when the past policy differs significantly from the policy being evaluated.

In this paper, we propose to use the technique of *doubly robust* (DR) estimation to overcome problems with the two existing approaches. Doubly robust (or doubly protected) estimation (Cassel, Särndal and Wretman, 1976; Robins, Rotnitzky and Zhao, 1994; Robins and Rotnitzky, 1995; Lunceford and Davidian, 2004; Kang and Schafer, 2007) is a statistical approach for estimation from incomplete data with an important property: if *either one* of the two estimators (i.e., DM or IPS) is correct, then the estimation is unbiased. This method thus increases the chances of drawing reliable inference.

We apply the doubly robust technique to policy evaluation and optimization in a contextual bandit setting. The most straightforward policies to consider are *stationary* policies, whose actions depend on the current, observed context alone. *Nonstationary* policies, on the other hand, map the current context and a history of past rounds to an action. They are of critical interest because *online learning* algorithms (also known as adaptive allocation rules), by definition, produce nonstationary policies. We address both stationary and nonstationary policies in this paper.

In Section 2, we describe previous work and connect our setting to the related area of dynamic treatment regimes.

In Section 3, we study stationary policy evaluation, analyzing the bias and variance of our core technique. Unlike previous theoretical analyses, we do not assume that either the reward model or the past policy model are correct. Instead, we show how the deviations of the two models from the truth impact bias and variance of the doubly robust estimator. To our knowledge, this style of analysis is novel and may provide insights into doubly robust estimation beyond the specific setting studied here. In Section 4, we apply this method to both policy evaluation and optimization, finding that this approach can substantially sharpen existing techniques.

In Section 5, we consider nonstationary policy evaluation. The main approach here is to use the historic data to obtain a sample of the run of an evaluated nonstationary policy via rejection sampling (Li et al., 2011). We combine the doubly robust technique with an improved form of rejection sampling that makes better use of data at the cost of small, controllable bias. Experiments in Section 6 suggest the combination is able to extract more information from data than existing approaches.

2. PRIOR WORK

2.1 Doubly Robust Estimation

Doubly robust estimation is widely used in statistical inference (see, e.g., Kang and Schafer, 2007, and the references therein). More recently, it has been used in Internet advertising to estimate the effects of new features for online advertisers (Lambert and Pregibon, 2007; Chan et al., 2010). Most of previous analysis of doubly robust estimation is focused on asymptotic behavior or relies on various modeling assumptions (e.g., Robins, Rotnitzky and Zhao, 1994; Lunceford and Davidian, 2004; Kang and Schafer, 2007). Our analysis is nonasymptotic and makes no such assumptions.

Several papers in machine learning have used ideas related to the basic technique discussed here, although not with the same language. For *benign bandits*, Hazan and Kale (2009) construct algorithms which use reward estimators to improve regret bounds when the variance of actual rewards is small. Similarly, the Offset Tree algorithm (Beygelzimer and Langford, 2009) can be thought of as using a crude reward estimate for the "offset." The algorithms and estimators described here are substantially more sophisticated.

Our nonstationary policy evaluation builds on the rejection sampling approach, which has been previously shown to be effective (Li et al., 2011). Relative to this earlier work, our nonstationary results take advantage of the doubly robust technique and a carefully introduced bias/variance tradeoff to obtain an empirical order-of-magnitude improvement in evaluation quality.

2.2 Dynamic Treatment Regimes

Contextual bandit problems are closely related to dynamic treatment regime (DTR) estimation/optimization in medical research. A DTR is a set of (possibly randomized) rules that specify what treatment to choose, given current characteristics (including past treatment history and outcomes) of a patient. In the terminology of the present paper, the patient's current characteristics are contextual information, a treatment is an action, and a DTR is a policy. Similar to contextual bandits, the quantity of interest in DTR can be expressed by a numeric reward signal related to the clinical outcome of a treatment. We comment on similarities and differences between DTR and contextual bandits in more detail in later sections of the paper, where we define our setting more formally. Here, we make a few higher-level remarks.

Due to ethical concerns, research in DTR is often performed with observational data rather than on patients. This corresponds to the offline version of contextual bandits, which only has access to past data but no ability to gather new data. Causal inference techniques have been studied to estimate the mean response of a given DTR (e.g., Robins, 1986; Murphy, van der Laan and Robins, 2001), and to optimize DTR (e.g., Murphy, 2003; Orellana, Rotnitzky and Robins, 2010). These two problems correspond to evaluation and optimization of policies in the present paper.

In DTR, however, a treatment typically exhibits a long-term effect on a patient's future "state," while in contextual bandits the contexts are drawn IID with no dependence on actions taken previously. Such a difference turns out to enable statistically more efficient estimators, which will be explained in greater detail in Section 5.2.

Despite these differences, as we will see later, contextual bandits and DTR share many similarities, and in some cases are almost identical. For example, analogous to the results introduced in this paper, doubly robust estimators have been applied to DTR estimation (Murphy, van der Laan and Robins, 2001), and also used as a subroutine for optimization in a family of parameterized policies (Zhang et al., 2012). The connection suggests a broader applicability of DTR techniques beyond the medical domain, for instance, to the Internet-motivated problems studied in this paper.

3. EVALUATION OF STATIONARY POLICIES

3.1 Problem Definition

We are interested in the *contextual bandit* setting where on each round:

1. A vector of covariates (or a *context*) $x \in \mathcal{X}$ is revealed.
2. An action (or *arm*) a is chosen from a given set $\mathcal{A}$.
3. A reward $r \in [0, 1]$ for the action a is revealed, but the rewards of other actions are not. The reward may depend stochastically on x and a.

We assume that contexts are chosen IID from an unknown distribution $D(x)$, the actions are chosen from a finite (and typically not too large) action set $\mathcal{A}$, and the distribution over rewards $D(r|a, x)$ does not change over time (but is unknown).

The input data consists of a finite stream of triples (x_k, a_k, r_k) indexed by $k = 1, 2, \dots, n$. We assume that the actions a_k are generated by some past (possibly nonstationary) policy, which we refer to as the *exploration policy*. The *exploration history* up to round k is denoted

$$z_k = (x_1, a_1, r_1, \dots, x_k, a_k, r_k).$$

Histories are viewed as samples from a probability measure μ. Our assumptions about data generation then translate into the assumption about factoring of μ as

$$\begin{aligned}&\mu(x_k, a_k, r_k|z_{k-1})\\ &\quad = D(x_k)\mu(a_k|x_k, z_{k-1})D(r_k|x_k, a_k),\end{aligned}$$

for any k. Note that apart from the unknown distribution D, the only degree of freedom above is $\mu(a_k|x_k, z_{k-1})$, that is, the unknown exploration policy.

When z_{k-1} is clear from the context, we use a shorthand μ_k for the conditional distribution over the kth triple

$$\mu_k(x, a, r) = \mu(x_k = x, a_k = a, r_k = r|z_{k-1}).$$

We also write $\mathbf{P}_k^\mu$ and $\mathbf{E}_k^\mu$ for $\mathbf{P}_\mu[\,\cdot\,|z_{k-1}]$ and $\mathbf{E}_\mu[\,\cdot\,|z_{k-1}]$.

Given input data z_n, we study the *stationary policy evaluation* problem. A stationary randomized policy ν is described by a conditional distribution $\nu(a|x)$ of choosing an action on each context. The goal is to use the history z_n to estimate the *value* of ν, namely, the expected reward obtained by following ν:

$$V(\nu) = \mathbf{E}_{x\sim D}\mathbf{E}_{a\sim\nu(\,\cdot\,|x)}\mathbf{E}_{r\sim D(\,\cdot\,|x,a)}[r].$$

In content recommendation on Web portals, for example, $V(\nu)$ measures the average click probability per user visit, one of the major metrics with critical business importance.

In order to have unbiased policy evaluation, we make a standard assumption that if $\nu(a|x) > 0$ then $\mu_k(a|x) > 0$ for all k (and all possible histories z_{k-1}). This clearly holds for instance if $\mu_k(a|x) > 0$ for all a. Since ν is fixed in our paper, we will write V for $V(\nu)$. To simplify notation, we extend the conditional distribution ν to a distribution over triples (x, a, r)

$$\nu(x, a, r) = D(x)\nu(a|x)D(r|a, x)$$

and hence $V = \mathbf{E}_\nu[r]$.

The problem of stationary policy evaluation, defined above, is slightly more general than DTR analysis in a typical cross-sectional observational study, where the exploration policy (known as "treatment mechanism" in the DTR literature) is stationary; that is, the conditional distribution $\mu(a_k|x_k, z_{k-1})$ is independent of z_{k-1} and identical across all k, that is, $\mu_k = \mu_1$ for all k.

3.2 Existing Approaches

The key challenge in estimating policy value in contextual bandits is that rewards are *partially* observable: in each round, only the reward for the chosen action is revealed; we do not know what the reward would have been if we chose a different action. Hence, the data collected in a contextual bandit process cannot be used directly to estimate a new policy's value: if in a context x the new policy selects an action a' different from the action a chosen during data collection, we simply do *not* have the reward signal for a'.

There are two common solutions for overcoming this limitation (see, e.g., Lambert and Pregibon, 2007, for an introduction to these solutions). The first, called the *direct method* (DM), forms an estimate $\hat{r}(x, a)$ of the expected reward conditioned on the context *and* action. The policy value is then estimated by

$$\hat{V}_{\mathrm{DM}} = \frac{1}{n}\sum_{k=1}^{n}\sum_{a\in\mathcal{A}}\nu(a|x_k)\hat{r}(x_k, a).$$

Clearly, if $\hat{r}(x, a)$ is a good approximation of the true expected reward $\mathbf{E}_D[r|x, a]$, then the DM estimate is close to V. A problem with this method is that the estimate $\hat{r}$ is typically formed without the knowledge of ν, and hence might focus on approximating expected reward in the areas that are irrelevant for ν and not sufficiently in the areas that are important for ν (see, e.g., the analysis of Beygelzimer and Langford, 2009).

The second approach, called *inverse propensity score* (IPS), is typically less prone to problems with bias. Instead of approximating the reward, IPS forms an approximation $\hat{\mu}_k(a|x)$ of $\mu_k(a|x)$, and uses this estimate to correct for the shift in action proportions between the exploration policy and the new policy:

$$\hat{V}_{\mathrm{IPS}} = \frac{1}{n}\sum_{k=1}^{n} \frac{\nu(a_k|x_k)}{\hat{\mu}_k(a_k|x_k)} \cdot r_k.$$

If $\hat{\mu}_k(a|x) \approx \mu_k(a|x)$, then the IPS estimate above will be, approximately, an unbiased estimate of V. Since we typically have a good (or even accurate) understanding of the data-collection policy, it is often easier to obtain good estimates $\hat{\mu}_k$, and thus the IPS estimator is in practice less susceptible to problems with bias compared with the direct method. However, IPS typically has a much larger variance, due to the increased range of the random variable $\nu(a_k|x_k)/\hat{\mu}_k(a_k|x_k)$. The issue becomes more severe when $\hat{\mu}_k(a_k|x_k)$ gets smaller in high probability areas under ν. Our approach alleviates the large variance problem of IPS by taking advantage of the estimate $\hat{r}$ used by the direct method.

3.3 Doubly Robust Estimator

Doubly robust estimators take advantage of both the estimate of the expected reward $\hat{r}$ and the estimate of action probabilities $\hat{\mu}_k(a|x)$. A similar idea has been suggested earlier by a number of authors for different estimation problems (Cassel, Särndal and Wretman, 1976; Rotnitzky and Robins, 1995; Robins and Rotnitzky, 1995; Murphy, van der Laan and Robins, 2001; Robins, 1998). For the setting in this section, the estimator of Murphy, van der Laan and Robins (2001) can be reduced to

$$\hat{V}_{\mathrm{DR}} = \frac{1}{n}\sum_{k=1}^{n}\biggl[\hat{r}(x_k,\nu) + \frac{\nu(a_k|x_k)}{\hat{\mu}_k(a_k|x_k)} \cdot \bigl(r_k - \hat{r}(x_k,a_k)\bigr)\biggr], \tag{3.1}$$

where

$$\hat{r}(x,\nu) = \sum_{a\in\mathcal{A}} \nu(a|x)\hat{r}(x,a)$$

is the estimate of $\mathbf{E}_\nu[r|x]$ derived from $\hat{r}$. Informally, the doubly robust estimator uses $\hat{r}$ as a baseline and if there is data available, a correction is applied. We will see that our estimator is unbiased if *at least one* of the estimators, $\hat{r}$ and $\hat{\mu}_k$, is accurate, hence the name *doubly robust*.

In practice, quite often neither $\mathbf{E}_D[r|x,a]$ or μ_k is accurate. It should be noted that, although μ_k tends to be much easier to estimate than $\mathbf{E}_D[r|x,a]$ in applications that motivate this study, it is rare to be able to get a perfect estimator, due to engineering constraints in complex applications like Web search and Internet advertising. Thus, a basic question is: How does the estimator $\hat{V}_{\mathrm{DR}}$ perform as the estimates $\hat{r}$ and $\hat{\mu}_k$ deviate from the truth? The following section analyzes bias and variance of the DR estimator as a function of errors in $\hat{r}$ and $\hat{\mu}_k$. Note that our DR estimator encompasses DM and IPS as special cases (by respectively setting $\hat{\mu}_k \equiv \infty$ and $\hat{r} \equiv 0$), so our analysis also encompasses DM and IPS.

3.4 Analysis

We assume that $\hat{r}(x,a) \in [0,1]$ and $\hat{\mu}_k(a|x) \in (0,\infty]$, but in general $\hat{\mu}_k$ does not need to represent conditional probabilities (our notation is only meant to indicate that $\hat{\mu}_k$ estimates μ_k, but no probabilistic structure). In general, we allow $\hat{r}$ and $\hat{\mu}_k$ to be random variables, as long as they satisfy the following independence assumptions:

- $\hat{r}$ is independent of z_n.
- $\hat{\mu}_k$ is conditionally independent of $\{(x_\ell, a_\ell, r_\ell)\}_{\ell\geq k}$, conditioned on z_{k-1}.

The first assumption means that $\hat{r}$ can be assumed fixed and determined before we see the input data z_n, for example, by initially splitting the input dataset and using the first part to obtain $\hat{r}$ and the second part to evaluate the policy. In our analysis, we condition on $\hat{r}$ and ignore any randomness in its choice.

The second assumption means that $\hat{\mu}_k$ is not allowed to depend on future. A simple way to satisfy this assumption is to split the dataset to form an estimator (and potentially also include data z_{k-1}). If we have some control over the exploration process, we might also have access to "perfect logging", that is, recorded probabilities $\mu_k(a_k|x_k)$. With perfect logging, we can achieve $\hat{\mu}_k = \mu_k$, respecting our assumptions.[2]

Analogous to $\hat{r}(x,a)$, we define the population quantity $r^*(x,a)$

$$r^*(x,a) = \mathbf{E}_D[r|x,a],$$

and define $r^*(x,\nu)$ similarly to $\hat{r}(x,\nu)$:

$$r^*(x,\nu) = \mathbf{E}_\nu[r|x].$$

[2] As we will see later in the paper, in order to reduce the variance of the estimator it might still be advantageous to use a slightly inflated estimator, for example, $\hat{\mu}_k = c\mu_k$ for $c > 1$, or $\hat{\mu}_k(a|x) = \max\{c, \mu_k(a|x)\}$ for some $c > 0$.

Let $\Delta(x,a)$ and $\varrho_k(x,a)$ denote, respectively, the *additive* error of $\hat r$ and the *multiplicative* error of $\hat\mu_k$:

$$\Delta(x,a) = \hat r(x,a) - r^*(x,a),$$
$$\varrho_k(x,a) = \mu_k(a|x)/\hat\mu_k(a|x).$$

We assume that for some $M \ge 0$, with probability one under μ:

$$\nu(a_k|x_k)/\hat\mu_k(a_k|x_k) \le M$$

which can always be satisfied by enforcing $\hat\mu_k \ge 1/M$.

To bound the error of $\hat V_{\mathrm{DR}}$, we first analyze a single term:

$$\hat V_k = \hat r(x_k,\nu) + \frac{\nu(a_k|x_k)}{\hat\mu_k(a_k|x_k)}\cdot\big(r_k - \hat r(x_k,a_k)\big).$$

We bound its range, bias, and conditional variance as follows (for proofs, see Appendix A):

LEMMA 3.1. *The range of $\hat V_k$ is bounded as*

$$|\hat V_k| \le 1 + M.$$

LEMMA 3.2. *The expectation of the term $\hat V_k$ is*

$$\mathbf{E}_k^\mu[\hat V_k] = \mathop{\mathbf{E}}_{(x,a)\sim\nu}\big[r^*(x,a) + \big(1-\varrho_k(x,a)\big)\Delta(x,a)\big].$$

LEMMA 3.3. *The variance of the term $\hat V_k$ can be decomposed and bounded as follows*:

(i) $$\mathbf{V}_k^\mu[\hat V_k] = \mathop{\mathbf{V}}_{x\sim D}\Big[\mathop{\mathbf{E}}_{a\sim\nu(\cdot|x)}\big[r^*(x,a) + \big(1-\varrho_k(x,a)\big)\cdot\Delta(x,a)\big]\Big] - \mathop{\mathbf{E}}_{x\sim D}\Big[\mathop{\mathbf{E}}_{a\sim\nu(\cdot|x)}\big[\varrho_k(x,a)\Delta(x,a)\big]^2\Big] + \mathop{\mathbf{E}}_{(x,a)\sim\nu}\bigg[\frac{\nu(a|x)}{\hat\mu_k(a|x)}\cdot\varrho_k(x,a)\cdot\mathop{\mathbf{V}}_{r\sim D(\cdot|x,a)}[r]\bigg] + \mathop{\mathbf{E}}_{(x,a)\sim\nu}\bigg[\frac{\nu(a|x)}{\hat\mu_k(a|x)}\cdot\varrho_k(x,a)\Delta(x,a)^2\bigg].$$

(ii) $$\mathbf{V}_k^\mu[\hat V_k] \le \mathop{\mathbf{V}}_{x\sim D}\big[r^*(x,\nu)\big] + 2\mathop{\mathbf{E}}_{(x,a)\sim\nu}\big[\big|\big(1-\varrho_k(x,a)\big)\Delta(x,a)\big|\big] + M\mathop{\mathbf{E}}_{(x,a)\sim\nu}\Big[\varrho_k(x,a)\cdot\mathop{\mathbf{E}}_{r\sim D(\cdot|x,a)}\big[\big(r-\hat r(x,a)\big)^2\big]\Big].$$

The range of $\hat V_k$ is controlled by the worst-case ratio $\nu(a_k|x_k)/\hat\mu_k(a_k|x_k)$. The bias of $\hat V_k$ gets smaller as Δ and ϱ_k become more accurate, that is, as $\Delta \approx 0$ and $\varrho_k \approx 1$. The expression for variance is more complicated. Lemma 3.3(i) lists four terms. The first term represents the variance component due to the randomness over x. The second term can contribute to the decrease in the variance. The final two terms represent the penalty due to the importance weighting. The third term scales with the conditional variance of rewards (given contexts and actions), and it vanishes if rewards are deterministic. The fourth term scales with the magnitude of Δ, and it captures the potential improvement due to the use of a good estimator $\hat r$.

The upper bound on the variance [Lemma 3.3(ii)] is easier to interpret. The first term is the variance of the estimated variable over x. The second term measures the quality of the estimators $\hat\mu_k$ and $\hat r$—it equals zero if either of them is perfect (or if the union of regions where they are perfect covers the support of ν over x and a). The final term represents the importance weighting penalty. It vanishes if we do not apply importance weighting (i.e., $\hat\mu_k \equiv \infty$ and $\varrho_k \equiv 0$). With nonzero ϱ_k, this term decreases with a better quality of $\hat r$—but it does not disappear even if $\hat r$ is perfect (unless the rewards are deterministic).

3.4.1 *Bias analysis.* Lemma 3.2 immediately yields a bound on the bias of the doubly robust estimator, as stated in the following theorem. The special case for stationary policies (second part of the theorem) has been shown by Vansteelandt, Bekaert and Claeskens (2012).

THEOREM 3.4. *Let Δ and ϱ_k be defined as above. Then the bias of the doubly robust estimator is*

$$|\mathbf{E}_\mu[\hat V_{\mathrm{DR}}] - V| = \frac{1}{n}\bigg|\mathbf{E}_\mu\bigg[\sum_{k=1}^{n}\mathop{\mathbf{E}}_{(x,a)\sim\nu}\big[\big(1-\varrho_k(x,a)\big)\Delta(x,a)\big]\bigg]\bigg|.$$

If the exploration policy μ and the estimator $\hat\mu_k$ are stationary (*i.e.*, *$\mu_k = \mu_1$ and $\hat\mu_k = \hat\mu_1$ for all k*), *the expression simplifies to*

$$|\mathbf{E}_\mu[\hat V_{\mathrm{DR}}] - V| = \big|\mathbf{E}_\nu\big[\big(1-\varrho_1(x,a)\big)\Delta(x,a)\big]\big|.$$

PROOF. The theorem follows immediately from Lemma 3.2. □

In contrast, we have (for simplicity, assuming stationarity of the exploration policy and its estimate)

$$|\mathbf{E}_\mu[\hat{V}_{\mathrm{DM}}] - V| = |\mathbf{E}_\nu[\Delta(x,a)]|,$$
$$|\mathbf{E}_\mu[\hat{V}_{\mathrm{IPS}}] - V| = |\mathbf{E}_\nu[r^*(x,a)(1-\varrho_1(x,a))]|,$$

where the first equality is based on the observation that DM is a special case of DR with $\hat{\mu}_k(a|x) \equiv \infty$ (and hence $\varrho_k \equiv 0$), and the second equality is based on the observation that IPS is a special case of DR with $\hat{r}(x,a) \equiv 0$ (and hence $\Delta \equiv r^*$).

In general, neither of the estimators dominates the others. However, if *either* $\Delta \approx 0$, *or* $\varrho_k \approx 1$, the expected value of the doubly robust estimator will be close to the true value, whereas DM requires $\Delta \approx 0$ and IPS requires $\varrho_k \approx 1$. Also, if $\|\varrho_k - 1\|_{p,\nu} \ll 1$ [for a suitable $L_p(\nu)$ norm], we expect that DR will outperform DM. Similarly, if $\varrho_k \approx 1$ but $\|\Delta\|_{p,\nu} \ll \|r^*\|_{p,\nu}$, we expect that DR will outperform IPS. Thus, DR can effectively take advantage of both sources of information to lower the bias.

3.4.2 *Variance analysis.* We argued that the expected value of $\hat{V}_{\mathrm{DR}}$ compares favorably with IPS and DM. We next look at the variance of DR. Since large-deviation bounds have a primary dependence on variance; a lower variance implies a faster convergence rate. To contrast DR with IPS and DM, we study a simpler setting with a stationary exploration policy, and *deterministic target policy* ν, that is, $\nu(\cdot|x)$ puts all the probability on a single action. In the next section, we revisit the fully general setting and derive a finite-sample bound on the error of DR.

THEOREM 3.5. *Let Δ and ϱ_k be defined as above. If exploration policy μ and the estimator $\hat{\mu}_k$ are stationary, and the target policy ν is deterministic, then the variance of the doubly robust estimator is*

$$\begin{aligned}\mathbf{V}_\mu[\hat{V}_{\mathrm{DR}}] &= \frac{1}{n}\Bigl(\mathop{\mathbf{V}}_{(x,a)\sim\nu}\bigl[r^*(x,a) \\ &\qquad + (1-\varrho_1(x,a))\Delta(x,a)\bigr] \\ &\quad + \mathop{\mathbf{E}}_{(x,a)\sim\nu}\Bigl[\frac{1}{\hat{\mu}_1(a|x)}\cdot\varrho_1(x,a)\cdot\mathop{\mathbf{V}}_{r\sim D(\cdot|x,a)}[r]\Bigr] \\ &\quad + \mathop{\mathbf{E}}_{(x,a)\sim\nu}\Bigl[\frac{1-\mu_1(a|x)}{\hat{\mu}_1(a|x)}\cdot\varrho_1(x,a)\Delta(x,a)^2\Bigr]\Bigr).\end{aligned}$$

PROOF. The theorem follows immediately from Lemma 3.3(i). □

The variance can be decomposed into three terms. The first term accounts for the randomness in x (note that a is deterministic given x). The other two terms can be viewed as the importance weighting penalty. These two terms disappear in DM, which does not use rewards r_k. The second term accounts for randomness in rewards and disappears when rewards are deterministic functions of x and a. However, the last term stays, accounting for the disagreement between actions taken by ν and μ_1.

Similar expressions can be derived for the DM and IPS estimators. Since IPS is a special case of DR with $\hat{r} \equiv 0$, we obtain the following equation:

$$\begin{aligned}\mathbf{V}_\mu[\hat{V}_{\mathrm{IPS}}] &= \frac{1}{n}\Bigl(\mathop{\mathbf{V}}_{(x,a)\sim\nu}\bigl[\varrho_1(x,a)r^*(x,a)\bigr] \\ &\quad + \mathop{\mathbf{E}}_{(x,a)\sim\nu}\Bigl[\frac{1}{\hat{\mu}_1(a|x)}\cdot\varrho_1(x,a)\cdot\mathop{\mathbf{V}}_{r\sim D(\cdot|x,a)}[r]\Bigr] \\ &\quad + \mathop{\mathbf{E}}_{(x,a)\sim\nu}\Bigl[\frac{1-\mu_1(a|x)}{\hat{\mu}_1(a|x)}\cdot\varrho_1(x,a)r^*(x,a)^2\Bigr]\Bigr).\end{aligned}$$

The first term will be of similar magnitude as the corresponding term of the DR estimator, provided that $\varrho_1 \approx 1$. The second term is identical to the DR estimator. However, the third term can be much larger for IPS if $\mu_1(a|x) \ll 1$ and $|\Delta(x,a)|$ is smaller than $r^*(x,a)$ for the actions chosen by ν.

In contrast, for the direct method, which is a special case of DR with $\hat{\mu}_k \equiv \infty$, the following variance is obtained immediately:

$$\mathbf{V}_\mu[\hat{V}_{\mathrm{DM}}] = \frac{1}{n}\mathop{\mathbf{V}}_{(x,a)\sim\nu}\bigl[r^*(x,a) + \Delta(x,a)\bigr].$$

Thus, the variance of the direct method does not have terms depending either on the exploration policy or the randomness in the rewards. This fact usually suffices to ensure that its variance is significantly lower than that of DR or IPS. However, as mentioned in the previous section, when we can estimate μ_k reasonably well (namely, $\varrho_k \approx 1$), the bias of the direct method is typically much larger, leading to larger errors in estimating policy values.

3.4.3 *Finite-sample error bound.* By combining bias and variance bounds, we now work out a specific finite-sample bound on the error of the estimator $\hat{V}_{\mathrm{DR}}$. While such an error bound could be used as a conservative confidence bound, we expect it to be too loose in most settings (as is typical for finite-sample bounds). Instead, our main intention is to explicitly highlight how the errors of estimators $\hat{r}$ and $\hat{\mu}_k$ contribute to the final error.

To begin, we first quantify magnitudes of the additive error $\Delta = \hat{r} - r^*$ of the estimator $\hat{r}$, and the relative error $|1 - \varrho_k| = |\hat{\mu}_k - \mu_k| / \hat{\mu}_k$ of the estimator $\hat{\mu}_k$:

ASSUMPTION 3.6. Assume there exist $\delta_\Delta, \delta_\varrho \geq 0$ such that

$$\mathop{\mathbf{E}}_{(x,a)\sim\nu}\bigl[\bigl|\Delta(x,a)\bigr|\bigr] \leq \delta_\Delta,$$

and with probability one under μ:

$$\bigl|1 - \varrho_k(x,a)\bigr| \leq \delta_\varrho \quad \text{for all } k.$$

Recall that $\nu/\hat{\mu}_k \leq M$. In addition, our analysis depends on the magnitude of the ratio $\varrho_k = \mu_k/\hat{\mu}_k$ and a term that captures both the variance of the rewards and the error of $\hat{r}$.

ASSUMPTION 3.7. Assume there exist $e_{\hat{r}}, \varrho_{\max} \geq 0$ such that with probability one under μ, for all k:

$$\mathop{\mathbf{E}}_{(x,a)\sim\nu}\Bigl[\mathop{\mathbf{E}}_{r\sim D(\cdot|x,a)}\bigl[\bigl(\hat{r}(x,a) - r\bigr)^2\bigr]\Bigr] \leq e_{\hat{r}},$$

$$\varrho_k(x,a) \leq \varrho_{\max} \quad \text{for all } x, a.$$

With the assumptions above, we can now bound the bias and variance of a single term $\hat{V}_k$. As in the previous sections, the bias decreases with the quality of $\hat{r}$ and $\hat{\mu}_k$, and the variance increases with the variance of the rewards and with the magnitudes of the ratios $\nu/\hat{\mu}_k \leq M$, $\mu_k/\hat{\mu}_k \leq \varrho_{\max}$. The analysis below for instance captures the bias-variance tradeoff of using $\hat{\mu}_k \approx c\mu_k$ for some $c > 1$: such a strategy can lead to a lower variance (by lowering M and $\varrho_{\max}$) but incurs some additional bias that is controlled by the quality of $\hat{r}$.

LEMMA 3.8. *Under Assumptions 3.6–3.7, with probability one under μ, for all k:*

$$\bigl|\mathbf{E}_k^\mu[\hat{V}_k] - V\bigr| \leq \delta_\varrho\delta_\Delta,$$

$$\mathbf{V}_k^\mu[\hat{V}_k] \leq \mathbf{V}_{x\sim D}\bigl[r^*(x,\nu)\bigr] + 2\delta_\varrho\delta_\Delta + M\varrho_{\max}e_{\hat{r}}.$$

PROOF. The bias and variance bound follow from Lemma 3.2 and Lemma 3.3(ii), respectively, by Hölder's inequality. □

Using the above lemma and Freedman's inequality yields the following theorem.

THEOREM 3.9. *Under Assumptions 3.6–3.7, with probability at least $1 - \delta$,*

$$\begin{aligned}
&|\hat{V}_{\mathrm{DR}} - V| \\
&\quad\leq \delta_\varrho\delta_\Delta \\
&\qquad + 2\max\Biggl\{\frac{(1+M)\ln(2/\delta)}{n}, \\
&\qquad \sqrt{\frac{(\mathbf{V}_{x\sim D}[r^*(x,\nu)] + 2\delta_\varrho\delta_\Delta + M\varrho_{\max}e_{\hat{r}})\ln(2/\delta)}{n}}\Biggr\}.
\end{aligned}$$

PROOF. The proof follows by Freedman's inequality (Theorem B.1 in Appendix B), applied to random variables $\hat{V}_k$, whose range and variance are bounded using Lemmas 3.1 and 3.8. □

The theorem is a finite-sample error bound that holds for all sample size n, and in the limit the error converges to $\delta_\varrho\delta_\Delta$. As we mentioned, this result gives a confidence interval for the doubly-robust estimate $\hat{V}_{\mathrm{DR}}$ for any finite sample n. Other authors have used asymptotic theory to derive confidence intervals for policy evaluation by showing that the estimator is asymptotically normal (e.g., Murphy, van der Laan and Robins, 2001; Zhang et al., 2012). When using asymptotic confidence bounds, it can be difficult to know a priori whether the asymptotic distribution has been reached, whereas our bound applies to all finite sample sizes. Although our bound may be conservative for small sample sizes, it provides a "safe" nonasymptotic confidence interval. In certain applications like those on the Internet, the sample size is usually large enough for this kind of nonasymptotic confidence bound to be almost as small as its asymptotic value (the term $\delta_q\delta_\Delta$ in Theorem 3.9), as demonstrated by Bottou et al. (2013) for online advertising.

Note that Assumptions 3.6–3.7 rely on bounds of $|1 - \varrho_k|$ and ϱ_k which have to hold with probability one. In Appendix C, we replace these bounds with moment bounds, and present analogs of Lemma 3.8 and Theorem 3.9.

4. EXPERIMENTS: THE STATIONARY CASE

This section provides empirical evidence for the effectiveness of the DR estimator compared to IPS and DM. We study these estimators on several real-world datasets. First, we use public benchmark datasets for multiclass classification to construct contextual bandit data, on which we evaluate both policy evaluation and policy optimization approaches. Second, we use a proprietary dataset to model the pattern of user visits to an Internet portal. We study covariate shift, which can be formalized as a special case of policy evaluation. Our third experiment uses another proprietary dataset to model slotting of various types of search results on a webpage.

4.1 Multiclass Classification with Partial Feedback

We begin with a description of how to turn a K-class classification dataset into a K-armed contextual bandit dataset. Instead of rewards, we will work with losses, specifically the $0/1$-classification error. The actions correspond to predicted classes. In the usual multiclass classification, we can infer the loss of any action on training data (since we know its correct label), so we call this a *full feedback* setting. On the other hand, in contextual bandits, we only know the loss of the specific action that was taken by the exploration policy, but of no other action, which we call a *partial feedback* setting. After choosing an exploration policy, our transformation from full to partial feedback simply "hides" the losses of actions that were not picked by the exploration policy.

This protocol gives us two benefits: we can carry out comparison using *public* multiclass classification datasets, which are more common than contextual bandit datasets. Second, fully revealed data can be used to obtain ground truth value of an arbitrary policy. Note that the original data is real-world, but exploration and partial feedback are simulated.

4.1.1 *Data generation*. In a classification task, we assume data are drawn IID from a fixed distribution: $(x, y) \sim D$, where $x \in \mathcal{X}$ is a real-valued covariate vector and $y \in \{1, 2, \dots, K\}$ is a class label. A typical goal is to find a classifier $\nu : \mathcal{X} \mapsto \{1, 2, \dots, K\}$ minimizing the classification error:

$$e(\nu) = \mathop{\mathbf{E}}_{(x,y)\sim D}\big[I[\nu(x) \neq y]\big],$$

where $I[\cdot]$ is an indicator function, equal to 1 if its argument is true and 0 otherwise.

The classifier ν can be viewed as a deterministic stationary policy with the action set $\mathcal{A} = \{1, \dots, K\}$ and the *loss function*

$$l(y, a) = I[a \neq y].$$

Loss minimization is symmetric to the reward maximization (under transformation $r = 1 - l$), but loss minimization is more commonly used in classification setting, so we work with loss here. Note that the distribution $D(y|x)$ together with the definition of the loss above, induce the conditional probability $D(l|x, a)$ in contextual bandits, and minimizing the classification error coincides with policy optimization.

To construct partially labeled data in multiclass classification, it remains to specify the exploration policy. We simulate stationary exploration with $\mu_k(a|x) = \mu_1(a|x) = 1/K$ for all a. Hence, the original example (x, y) is transformed into an example $(x, a, l(y, a))$ for a randomly selected action $a \sim \text{uniform}(1, 2, \dots, K)$. We assume perfect logging of the exploration policy and use the estimator $\hat{\mu}_k = \mu_k$. Below, we describe how we obtained an estimator $\hat{l}(x, a)$ (the counterpart of $\hat{r}$).

Table 1 summarizes the benchmark problems adopted from the UCI repository (Asuncion and Newman, 2007).

4.1.2 *Policy evaluation*. We first investigate whether the DR technique indeed gives more accurate estimates of the policy value (or classification error in our context), compared to DM and IPS. For each dataset:

1. We randomly split data into training and evaluation sets of (roughly) the same size;
2. On the training set, we keep full classification feedback of form (x, y) and train the direct loss minimization (DLM) algorithm of McAllester, Hazan and Keshet (2011), based on gradient descent, to obtain a classifier (see Appendix D for details). This classifier constitutes the policy ν whose value we estimate on evaluation data;
3. We compute the classification error on fully observed evaluation data. This error is treated as the ground truth for comparing various estimates;
4. Finally, we apply the transformation in Section 4.1.1 to the evaluation data to obtain a partially labeled set (exploration history), from which DM, IPS and DR estimates are computed.

TABLE 1
Characteristics of benchmark datasets used in Section 4.1

Dataset	Ecoli	Glass	Letter	Optdigits	Page-blocks	Pendigits	Satimage	Vehicle	Yeast
Classes (K)	8	6	26	10	5	10	6	4	10
Sample size	336	214	20,000	5620	5473	10,992	6435	846	1484

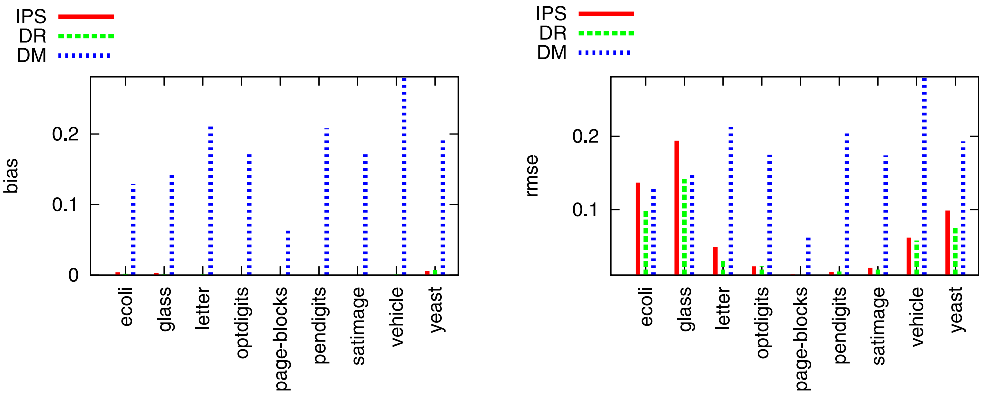

FIG. 1. *Comparison of* bias (*left*) *and* rmse (*right*) *of the three estimators of classification error on partial feedback classification data.*

Both DM and DR require estimating the expected conditional loss for a given (x, a). We use a linear loss model: $\hat{l}(x, a) = w_a \cdot x$, parameterized by K weight vectors $\{w_a\}_{a\in\{1,\dots,K\}}$, and use least-squares ridge regression to fit w_a based on the training set.

Step 4 of the above protocol is repeated 500 times, and the resulting bias and rmse (root mean squared error) are reported in Figure 1.

As predicted by analysis, both IPS and DR are unbiased, since the estimator $\hat{\mu}_k$ is perfect. In contrast, the linear loss model fails to capture the classification error accurately, and as a result, DM suffers a much larger bias.

While IPS and DR estimators are unbiased, it is apparent from the rmse plot that the DR estimator enjoys a lower variance, which translates into a smaller rmse. As we shall see next, this has a substantial effect on the quality of policy optimization.

4.1.3 *Policy optimization.* This subsection deviates from much of the paper to study *policy optimization* rather than *policy evaluation*. Given a space of possible policies, policy optimization is a procedure that searches this space for the policy with the highest value. Since policy values are unknown, the optimization procedure requires access to exploration data and uses a policy evaluator as a subroutine. Given the superiority of DR over DM and IPS for policy evaluation (in previous subsection), a natural question is whether a similar benefit can be translated into policy optimization as well. Since DM is significantly worse on all datasets, as indicated in Figure 1, we focus on the comparison between IPS and DR.

Here, we apply the data transformation in Section 4.1.1 to the *training* data, and then learn a classifier based on the loss estimated by IPS and DR, respectively. Specifically, for each dataset, we repeat the following steps 30 times:

1. We randomly split data into training (70%) and test (30%) sets;
2. We apply the transformation in Section 4.1.1 to the training data to obtain a partially labeled set (exploration history);
3. We then use the IPS and DR estimators to impute unrevealed losses in the training data; that is, we transform each partial-feedback example (x, a, l) into a *cost sensitive* example of the form $(x, l_1, \dots, l_K)$ where $l_{a'}$ is the loss for action a', imputed from the partial feedback data as follows:

$$l_{a'} = \begin{cases} \hat{l}(x, a') + \dfrac{l - \hat{l}(x, a')}{\hat{\mu}_1(a'|x)}, & \text{if } a' = a, \\ \hat{l}(x, a'), & \text{if } a' \neq a. \end{cases}$$

   In both cases, $\hat{\mu}_1(a'|x) = 1/K$ (recall that $\hat{\mu}_1 = \hat{\mu}_k$); in DR we use the loss estimate (described below), in IPS we use $\hat{l}(x, a') = 0$;
4. Two cost-sensitive multiclass classification algorithms are used to learn a classifier from the losses completed by either IPS or DR: the first is DLM used also in the previous section (see Appendix D and McAllester, Hazan and Keshet, 2011), the other is the Filter Tree reduction of Beygelzimer, Langford and Ravikumar (2008) applied to a decision-tree base learner (see Appendix E for more details);
5. Finally, we evaluate the learned classifiers on the test data to obtain classification error.

Again, we use least-squares ridge regression to build a linear loss estimator: $\hat{l}(x, a) = w_a \cdot x$. However, since the training data is partially labeled, w_a is fitted only using training data (x, a', l) for which $a = a'$. Note that this choice slightly violates our assumptions, because $\hat{l}$ is *not* independent of the training data z_n. However, we expect the dependence to be rather weak, and we find

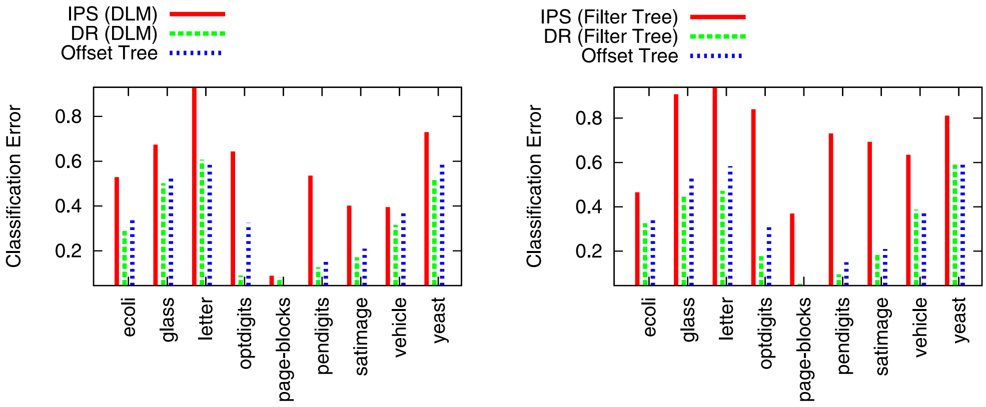

FIG. 2. *Classification error of direct loss minimization* (*left*) *and filter tree* (*right*). *Note that the representations used by DLM and the trees are very different, making any comparison between the two approaches difficult. However, the Offset Tree and Filter Tree approaches share a similar tree representation of the classifiers, so differences in performance are purely a matter of superior optimization.*

this approach to be more realistic in practical scenarios where one might want to use all available data to form the reward estimator, for instance due to data scarcity.

Average classification errors (obtained in Step 5 above) of 30 runs are plotted in Figure 2. Clearly, for policy optimization, the advantage of the DR is even greater than for policy evaluation. In all datasets, DR provides substantially more reliable loss estimates than IPS, and results in significantly improved classifiers.

Figure 2 also includes classification error of the Offset Tree reduction (Beygelzimer and Langford, 2009), which is designed specifically for policy optimization with partially labeled data.[3] While the IPS versions of DLM and Filter Tree are rather weak, the DR versions are competitive with Offset Tree in all datasets, and in some cases significantly outperform Offset Tree.

Our experiments show that DR provides similar improvements in two very different algorithms, one based on gradient descent, the other based on tree induction, suggesting the DR technique is generally useful when combined with different algorithmic choices.

4.2 Estimating the Average Number of User Visits

The next problem we consider is estimating the average number of user visits to a popular Internet portal. We formulate this as a regression problem and in our evaluation introduce an artificial covariate shift. As in the previous section, the original data is real-world, but the covariate shift is simulated.

Real user visits to the website were recorded for about 4 million *bcookies*[4] randomly selected from all bcookies during March 2010. Each bcookie is associated with a sparse binary covariate vector in 5000 dimensions. These covariates describe browsing behavior as well as other information (such as age, gender and geographical location) of the bcookie. We chose a fixed time window in March 2010 and calculated the number of visits by each selected bcookie during this window. To summarize, the dataset contains $N = 3{,}854{,}689$ data points: $D = \{(b_i, x_i, v_i)\}_{i=1,\dots,N}$, where b_i is the ith (unique) bcookie, x_i is the corresponding binary covariate vector, and v_i is the number of visits (the response variable); we treat the empirical distribution over D as the ground truth.

If it is possible to sample x uniformly at random from D and measure the corresponding value v, the sample mean of v will be an unbiased estimate of the true average number of user visits, which is 23.8 in this problem. However, in various situations, it may be difficult or impossible to ensure a uniform sampling scheme due to practical constraints. Instead, the best that one can do is to sample x from some other distribution (e.g., allowed by the business constraints) and measure the corresponding value v. In other words, the sampling distribution of x is changed, but the conditional distribution of v given x remains the same. In this case, the sample average of v may be a biased estimate of the true quantity of interest. This setting is known as *covariate shift* (Shimodaira, 2000), where

[3] We used decision trees as the base learner in Offset Trees to parallel our base learner choice in Filter Trees. The numbers reported here are not identical to those by Beygelzimer and Langford (2009), even though we used a similar protocol on the same datasets, probably because of small differences in the data structures used.

[4] A bcookie is a unique string that identifies a user. Strictly speaking, one user may correspond to multiple bcookies, but for simplicity we equate a bcookie with a user.

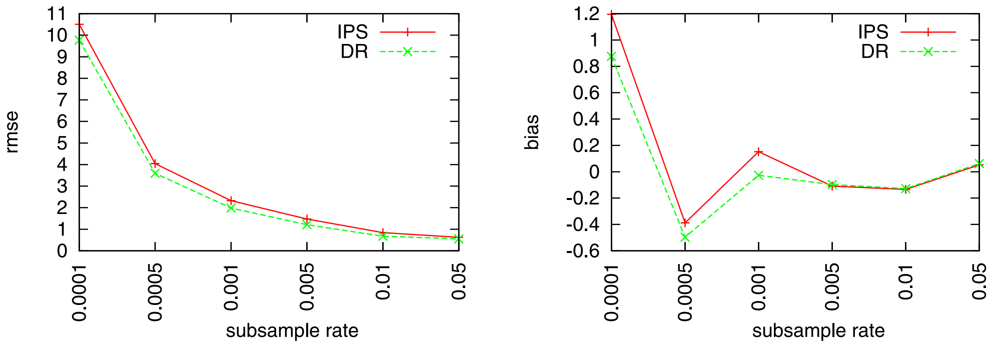

FIG. 3. *Comparison of IPS and DR*: rmse (*left*), bias (*right*). *The ground truth policy value* (*average number of user visits*) *is* 23.8.

data are missing at random (see Kang and Schafer, 2007, for related comparisons).

Covariate shift can be modeled as a contextual bandit problem with 2 actions: action $a = 0$ corresponding to "conceal the response" and action $a = 1$ corresponding to "reveal the response." Below we specify the stationary exploration policy $\mu_k(a|x) = \mu_1(a|x)$. The contextual bandit data is generated by first sampling $(x, v) \sim D$, then choosing an action $a \sim \mu_1(\cdot\,|x)$, and observing the reward $r = a \cdot v$ (i.e., reward is only revealed if $a = 1$). The exploration policy μ_1 determines the covariate shift. The quantity of interest, $\mathbf{E}_D[v]$, corresponds to the value of the constant policy ν which always chooses "reveal the response."

To define the exploration sampling probabilities $\mu_1(a = 1|x)$, we adopted an approach similar to Gretton et al. (2008), with a bias toward the smaller values along the first principal component of the distribution over x. In particular, we obtained the first principal component (denoted $\bar{x}$) of all covariate vectors $\{x_i\}_{i=1,\dots,N}$, and projected all data onto $\bar{x}$. Let ϕ be the density of a univariate normal distribution with mean $m + (\bar{m} - m)/3$ and standard deviation $(\bar{m} - m)/4$, where m is the minimum and $\bar{m}$ is the mean of the projected values. We set $\mu_1(a = 1|x) = \min\{\phi(x \cdot \bar{x}), 1\}$.

To control the size of exploration data, we randomly subsampled a fraction $f \in \{0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05\}$ from the entire dataset D and then chose actions a according to the exploration policy. We then calculated the IPS and DR estimates on this subsample, assuming perfect logging, that is, $\hat{\mu}_k = \mu_k$.[5] The whole process was repeated 100 times.

The DR estimator required building a reward model $\hat{r}(x, a)$, which, for a given covariate vector x and $a = 1$, predicted the average number of visits (and for $a = 0$ was equal to zero). Again, least-squares ridge regression was used on a separate dataset to fit a linear model $\hat{r}(x, 1) = w \cdot x$ from the exploration data.

Figure 3 summarizes the estimation error of the two methods with increasing exploration data size. For both IPS and DR, the estimation error goes down with more data. In terms of rmse, the DR estimator is consistently better than IPS, especially when dataset size is smaller. The DR estimator often reduces the rmse by a fraction between 10% and 20%, and on average by 13.6%. By comparing to the bias values (which are much smaller), it is clear that DR's gain of accuracy comes from a lower variance, which accelerates convergence of the estimator to the true value. These results confirm our analysis that DR tends to reduce variance provided that a reasonable reward estimator is available.

4.3 Content Slotting in Response to User Queries

In this section, we compare our estimators on a proprietary real-world dataset consisting of web search queries. In response to a search query, the search engine returns a set of search results. A search result can be of various types such as a web-link, a news snippet or a movie information snippet. We will be evaluating policies that decide which among the different result types to present at the first position. The reward is meant to capture the relevance for the user. It equals $+1$ if the user clicks on the result at the first position, -1 if the user clicks on some result below the first position, and 0 otherwise (for instance, if the user leaves the search page, or decides to rewrite the query). We call this a *click-skip reward*.

Our partially labeled dataset consists of tuples of the form (x_k, a_k, r_k, p_k), where x_k is the covariate vector (a sparse, high-dimensional representation of the

[5] Assuming perfect knowledge of exploration probabilities is fair when we compare IPS and DR. However, it does not give implications of how DR compares against DM when there is an estimation error in $\hat{\mu}_k$.

terms of the query as well as other contextual information, such as user information), $a_k \in \{$web-link, news, movie$\}$ is the type of result at the first position, r_k is the click-skip reward, and p_k is the recorded probability with which the exploration policy chose the given result type. Note that due to practical constraints, the values p_k do not always exactly correspond to $\mu_k(a_k|x_k)$ and should be really viewed as the "best effort" approximation of perfect logging. We still expect them to be highly accurate, so we use the estimator $\hat{\mu}_k(a_k|x_k) = p_k$.

The page views corresponding to these tuples represent a small percentage of user traffic to a major website; any visit to the website had a small chance of being part of this experiment. Data was collected over a span of several days during July 2011. It consists of 1.2 million tuples, out of which the first 1 million were used for estimating $\hat{r}$ (training data) with the remainder used for policy evaluation (evaluation data). The evaluation data was further split into 10 independent subsets of equal size, which were used to estimate variance of the compared estimators.

We estimated the value of two policies: the exploration policy itself, and the *argmax* policy (described below). Evaluating exploration policy on its own exploration data (we call this setup *self-evaluation*) serves as a sanity check. The *argmax* policy is based on a linear estimator $r'(x, a) = w_a \cdot x$ (in general different from $\hat{r}$), and chooses the action with the largest predicted reward $r'(x, a)$ (hence the name). We fitted $r'(x, a)$ on training data by importance-weighted linear regression with importance weights $1/p_k$. Note that both $\hat{r}$ and r' are linear estimators obtained from the same training set, but $\hat{r}$ was computed without importance weights and we therefore expect it to be more biased.

Table 2 contains the comparison of IPS, DM and DR, for both policies under consideration. For business reasons, we do not report the estimated reward directly, but normalize to either the empirical average reward (for *self-evaluation*) or the IPS estimate (for the *argmax* policy evaluation).

The experimental results are generally in line with theory. The variance is smallest for DR, although IPS does surprisingly well on this dataset, presumably because $\hat{r}$ is not sufficiently accurate. The Direct Method (DM) has an unsurprisingly large bias. If we divide the listed standard deviations by $\sqrt{10}$, we obtain standard errors, suggesting that DR has a slight bias (on self-evaluation where we know the ground truth). We believe that this is due to imperfect logging.

TABLE 2

The results of different policy evaluators on two standard policies for a real-world exploration problem. In the first column, results are normalized by the (known) actual reward of the deployed policy. In the second column, results are normalized by the reward reported by IPS. All ± are computed as standard deviations over results on 10 disjoint test sets. In previous publication of the same experiments (Dudík et al., 2012), we used a deterministic-policy version of DR (the same as in Dudík, Langford and Li, 2011), hence the results for self-evaluation presented there slightly differ

	Self-evaluation	**Argmax**
IPS	0.995 ± 0.041	1.000 ± 0.027
DM	1.213 ± 0.010	1.211 ± 0.002
DR	0.974 ± 0.039	0.991 ± 0.026

5. EVALUATION OF NONSTATIONARY POLICIES

5.1 Problem Definition

The contextual bandit setting can also be used to model a broad class of sequential decision-making problems, where the decision maker adapts her action-selection policy over time, based on her observed history of context-action-reward triples. In contrast to policies studied in the previous two sections, such a policy depends on both the current context and the current *history* and is therefore *nonstationary*.

In the personalized news recommendation example (Li et al., 2010), a learning algorithm chooses an article (an action) for the current user (the context), with the need for balancing *exploration* and *exploitation*. Exploration corresponds to presenting articles about which the algorithm does not yet have enough data to conclude if they are of interest to a particular type of user. Exploitation corresponds to presenting articles for which the algorithm collected enough data to know that they elicit a positive response. At the beginning, the algorithm may pursue more aggressive exploration since it has a more limited knowledge of what the users like. As more and more data is collected, the algorithm eventually converges to a good recommendation policy and performs more exploitation. Obviously, for the same user, the algorithm may choose different articles in different stages, so the policy is *not* stationary. In machine learning terminology, such adaptive procedures are called *online learning* algorithms. Evaluating performance of an online learning algorithm (in terms of average per-step reward when run for T steps) is an important problem in practice. Online learning algorithms are specific instances of nonstationary policies.

Formally, a nonstationary randomized policy is described by a conditional distribution $\pi(a_t|x_t, h_{t-1})$ of choosing an action a_t on a context x_t, given the history of past observations

$$h_{t-1} = (x_1, a_1, r_1), \ldots, (x_{t-1}, a_{t-1}, r_{t-1}).$$

We use the index t (instead of k), and write h_t (instead of z_k) to make clear the distinction between the histories experienced by the target policy π versus the exploration policy μ.

A target history of length T is denoted h_T. In our analysis, we extend the target policy $\pi(a_t|x_t, h_{t-1})$ into a probability distribution over h_T defined by the factoring

$$\pi(x_t, a_t, r_t|h_{t-1}) = D(x_t)\pi(a_t|x_t, h_{t-1})D(r_t|x_t, a_t).$$

Similarly to μ, we define shorthands $\pi_t(x, a, r)$, $\mathbf{P}_t^\pi$, $\mathbf{E}_t^\pi$. The goal of nonstationary policy evaluation is to estimate the expected cumulative reward of policy π after T rounds:

$$V_{1:T} = \mathop{\mathbf{E}}_{h_T \sim \pi}\left[\sum_{t=1}^{T} r_t\right].$$

In the news recommendation example, r_t indicates whether a user clicked on the recommended article, and $V_{1:T}$ is the expected number of clicks garnered by an online learning algorithm after serving T user visits. A more effective learning algorithm, by definition, will have a higher $V_{1:T}$ value (Li et al., 2010).

Again, to have unbiased policy evaluation, we assume that if $\pi_t(a|x) > 0$ for any t (and some history h_{t-1}) then $\mu_k(a|x) > 0$ for all k (and all possible histories z_{k-1}). This clearly holds for instance if $\mu_k(a|x) > 0$ for all a.

In our analysis of nonstationary policy evaluation, we assume perfect logging, that is, we assume access to probabilities

$$p_k := \mu_k(a_k|x_k).$$

Whereas in general this assumption does not hold, it is realistic in some applications such as those on the Internet. For example, when a website chooses one news article from a pool to recommend to a user, engineers often have full control/knowledge of how to randomize the article selection process (Li et al., 2010; Li et al., 2011).

5.2 Relation to Dynamic Treatment Regimes

The nonstationary policy evaluation problem defined above is closely related to DTR analysis in a longitudinal observational study. Using the same notation, the inference goal in DTR is to estimate the expected sum of rewards by following a possibly randomized rule π for T steps.[6] Unlike contextual bandits, there is no assumption on the distribution from which the data z_n is generated. More precisely, given an exploration policy μ, the data generation is described by

$$\begin{aligned}&\mu(x_k, a_k, r_k|z_{k-1})\\ &\quad = D(x_k|z_{k-1})\mu(a_k|x_k, z_{k-1})D(r_k|x_k, a_k, z_{k-1}).\end{aligned}$$

Compared to the data-generation process in contextual bandits (see Section 3.1), one allows the laws of x_k and r_k to depend on history z_{k-1}. The target policy π is subject to the same conditional laws. The setting in longitudinal observational studies is therefore more general than contextual bandits.

IPS-style estimators (such as DR of the previous section) can be extended to handle nonstationary policy evaluation, where the likelihood ratios are now the ratios of likelihoods of the whole length-T trajectories. In DTR analysis, it is often assumed that the number of trajectories is much larger than T. Under this assumption and with T small, the variance of IPS-style estimates is on the order of $O(1/n)$, diminishing to 0 as $n \to \infty$.

In contextual bandits, one similarly assumes $n \gg T$. However, the number of steps T is often large, ranging from hundreds to millions. The likelihood ratio for a length-T trajectory can be exponential in T, resulting in exponentially large variance. As a concrete example, consider the case where the exploration policy (i.e., the treatment mechanism) chooses actions uniformly at random from K possibilities, and where the target policy π is a deterministic function of the current history and context. The likelihood ratio of any trajectory is exactly K^T, and there are n/T trajectories (by breaking z_n into n/T pieces of length T). Assuming bounded variance of rewards, the variance of IPS-style estimators given data z_n is $O(TK^T/n)$, which can be extremely large (or even vacuous) for even moderate values of T, such as those in the studies of online learning in the Internet applications.

[6] In DTR often the goal is to estimate the expectation of a *composite* outcome that depends on the entire length-T trajectory. However, the objective of composite outcomes can easily be reformulated as a sum of properly redefined rewards.

In contrast, the "replay" approach of Li et al. (2011) takes advantage of the independence between (x_k, r_k) and history z_{k-1}. It has a variance of $O(KT/n)$, ignoring logarithmic terms, when the exploration policy is uniformly random. When the exploration data is generated by a nonuniformly random policy, one may apply rejection sampling to simulate uniformly random exploration, obtaining a subset of the exploration data, which can then be used to run the replay approach. However, this method may discard a large fraction of data, especially when the historical actions in the log are chosen from a highly nonuniform distribution, which can yield an unacceptably large variance. The next subsection describes an improved replay-based estimator that uses doubly-robust estimation as well as a variant of rejection sampling.

5.3 A Nonstationary Policy Evaluator

Our replay-based nonstationary policy evaluator (Algorithm 1) takes advantage of high accuracy of DR estimator while tackling nonstationarity via rejection sampling. We substatially improve sample use (i.e., acceptance rate) in rejection sampling while only modestly increasing the bias. This algorithm is referred to as DR-ns, for "doubly robust nonstationary." Over the run of the algorithm, we process the exploration history and run rejection sampling [Steps (5)–(6)] to create a simulated history h_t of the interaction between the target policy and the environment. If the algorithm manages to simulate T steps of history, it exits and returns an estimate $\hat{V}_{\text{DR-ns}}$ of the cumulative reward $V_{1:T}$, and an estimate $\hat{V}^{\text{avg}}_{\text{DR-ns}}$ of the average reward $V_{1:T}/T$; otherwise, it reports failure indicating not enough data is available.

Since we assume $n \gg T$, the algorithm fails with a small probability as long as the exploration policy does not assign too small probabilities to actions. Specifically, let $\alpha > 0$ be a lower bound on the acceptance probability in the rejection sampling step; that is, the condition in Step (6) succeeds with probability at least α. Then, using the Hoeffding's inequality, one can show that the probability of failure of the algorithm is at most δ if

$$n \geq \frac{T + \ln(e/\delta)}{\alpha}.$$

Algorithm 1
DR-ns(π, $\{(x_k, a_k, r_k, p_k)\}_{k=1,2,\dots,n}$, $\hat{r}$, q, $c_{\max}$, T)

```
Input:
    target nonstationary policy $\pi$
    exploration data $\{(x_k, a_k, r_k, p_k)\}_{k=1,2,\dots,n}$
    reward estimator $\hat{r}(x, a)$
    rejection sampling parameters:
        $q \in [0, 1]$ and $c_{\max} \in (0, 1]$
    number of steps $T$ for estimation

Initialize:
    simulated history of target policy $h_0 \leftarrow \varnothing$
    simulated step of target policy $t \leftarrow 0$
    acceptance rate multiplier $c_1 \leftarrow c_{\max}$
    cumulative reward estimate $\hat{V}_{\text{DR-ns}} \leftarrow 0$
    cumulative normalizing weight $C \leftarrow 0$
    importance weights seen so far $Q \leftarrow \varnothing$

For $k = 1, 2, \dots$ consider event $(x_k, a_k, r_k, p_k)$:
(1) $\hat{V}_k \leftarrow \hat{r}(x_k, \pi_t) + \frac{\pi_t(a_k|x_k)}{p_k} \cdot (r_k - \hat{r}(x_k, a_k))$
(2) $\hat{V}_{\text{DR-ns}} \leftarrow \hat{V}_{\text{DR-ns}} + c_t \hat{V}_k$
(3) $C \leftarrow C + c_t$
(4) $Q \leftarrow Q \cup \{\frac{p_k}{\pi_t(a_k|x_k)}\}$
(5) Let $u_k \sim$ UNIFORM[0, 1]
(6) If $u_k \leq \frac{c_t \pi_t(a_k|x_k)}{p_k}$
    (a) $h_t \leftarrow h_{t-1} + (x_k, a_k, r_k)$
    (b) $t \leftarrow t + 1$
    (c) if $t = T + 1$, go to "Exit"
    (d) $c_t \leftarrow \min\{c_{\max}, q\text{th quantile of } Q\}$

Exit: If $t < T + 1$, report failure and terminate;
otherwise, return:
    cumulative reward estimate $\hat{V}_{\text{DR-ns}}$
    average reward estimate $\hat{V}^{\text{avg}}_{\text{DR-ns}} := \hat{V}_{\text{DR-ns}}/C$
```

Note that the algorithm returns one "sample" of the policy value. In reality, the algorithm continuously consumes a stream of n data, outputs a sample of policy value whenever a length-T history is simulated, and finally returns the average of these samples. Suppose we aim to simulate m histories of length T. Again, by Hoeffding's inequality, the probability of failing to obtain m trajectories is at most δ if

$$n \geq \frac{mT + \ln(e/\delta)}{\alpha}.$$

Compared with naive rejection sampling, our approach differs in two respects. First, we use not only the accepted samples, but also the rejected ones to estimate the expected reward $\mathbf{E}^{\pi}_{t}[r]$ with a DR estimator [see Step (1)]. As we will see below, the value of $1/c_t$ is in expectation equal to the total number of exploration samples used while simulating the tth action of the target policy. Therefore, in Step (2), we effectively take an average of $1/c_t$ estimates of $\mathbf{E}_t^{\pi}[r]$, decreasing the variance of the final estimator. This is in addition to lower variance due to the use of the doubly robust estimate in Step (1).

The second modification is in the control of the acceptance rate (i.e., the bound α above). When simulating the tth action of the target policy, we accept exploration samples with a probability $\min\{1, c_t\pi_t/p_k\}$ where c_t is a multiplier [see Steps (5)–(6)]. We will see below that the bias of the estimator is controlled by the probability that $c_t\pi_t/p_k$ exceeds 1, or equivalently, that p_k/π_t falls below c_t. As a heuristic toward controlling this probability, we maintain a set Q consisting of observed density ratios p_k/π_t, and at the beginning of simulating the tth action, we set c_t to the qth quantile of Q, for some small value of q [Step (6)(d)], while never allowing it to exceed some predetermined $c_{\max}$. Thus, the value q approximately corresponds to the probability value that we wish to control. Setting $q = 0$, we obtain the unbiased case (in the limit). By using larger values of q, we increase the bias, but reach the length T with fewer exploration samples thanks to increased acceptance rate. A similar effect is obtained by varying $c_{\max}$, but the control is cruder, since it ignores the evaluated policy. In our experiments, we therefore set $c_{\max} = 1$ and rely on q to control the acceptance rate. It is an interesting open question how to select q and c in practice.

To study our algorithm DR-ns, we modify the definition of the exploration history so as to include the samples u_k from the uniform distribution used by the algorithm when processing the kth exploration sample. Thus, we have an augmented definition

$$z_k = (x_1, a_1, r_1, u_1, \dots, x_k, a_k, r_k, u_k).$$

With this in mind, expressions $\mathbf{P}_k^{\mu}$ and $\mathbf{E}_k^{\mu}$ include conditioning on variables $u_1, \dots, u_{k-1}$, and μ is viewed as a distribution over augmented histories z_n.

For convenience of analysis, we assume in this section that we have access to an infinite exploration history z (i.e., z_n for $n = \infty$) and that the counter t in the pseudocode eventually becomes $T + 1$ with probability one (at which point h_T is generated). Such an assumption is mild in practice when n is much larger than T.

Formally, for $t \geq 1$, let $\kappa(t)$ be the index of the tth sample accepted in Step (6); thus, κ converts an index in the target history into an index in the exploration history. We set $\kappa(0) = 0$ and define $\kappa(t) = \infty$ if fewer than t samples are accepted. Note that κ is a deterministic function of the history z (thanks to including samples u_k in z). We assume that $\mathbf{P}_{\mu}[\kappa(T) = \infty] = 0$. This means that the algorithm (together with the exploration policy μ) generates a distribution over histories h_T; we denote this distribution $\hat{\pi}$.

Let $B(t) = \{\kappa(t-1) + 1, \kappa(t-1) + 2, \dots, \kappa(t)\}$ for $t \geq 1$ denote the set of sample indices between the $(t-1)$st acceptance and the tth acceptance. This set of samples is called the tth block. The contribution of the tth block to the value estimator is denoted $\hat{V}_{B(t)} = \sum_{k \in B(t)} \hat{V}_k$. After completion of T blocks, the two estimators returned by our algorithm are

$$\hat{V}_{\text{DR-ns}} = \sum_{t=1}^{T} c_t \hat{V}_{B(t)}, \qquad \hat{V}_{\text{DR-ns}}^{\text{avg}} = \frac{\sum_{t=1}^{T} c_t \hat{V}_{B(t)}}{\sum_{t=1}^{T} c_t |B(t)|}.$$

5.4 Bias Analysis

A simple approach to evaluating a nonstationary policy is to divide the exploration data into several parts, run the algorithm separately on each part to generate simulated histories, obtaining estimates $\hat{V}_{\text{DR-ns}}^{(1)}, \dots, \hat{V}_{\text{DR-ns}}^{(m)}$, and return the average $\sum_{i=1}^{m} \hat{V}_{\text{DR-ns}}^{(i)}/m$.[7] Here, we assume n is large enough so that m simulated histories of length T can be generated with high probability. Using standard concentration inequalities, we can then show that the average is within $O(1/\sqrt{m})$ of the expectation $\mathbf{E}_{\mu}[\hat{V}_{\text{DR-ns}}]$. The remaining piece is then bounding the bias term $\mathbf{E}_{\mu}[\hat{V}_{\text{DR-ns}}] - \mathbf{E}_{\pi}[\sum_{t=1}^{T} r_t]$.[8]

Recall that $\hat{V}_{\text{DR-ns}} = \sum_{t=1}^{T} c_t \hat{V}_{B(t)}$. The source of bias are events when c_t is not small enough to guarantee that $c_t\pi_t(a_k|x_k)/p_k$ is a probability. In this case, the probability that the kth exploration sample includes the action a_k and is accepted is

$$p_k \min\left\{1, \frac{c_t\pi_t(a_k|x_k)}{p_k}\right\} = \min\{p_k, c_t\pi_t(a_k|x_k)\}, \tag{5.1}$$

which may violate the unbiasedness requirement of rejection sampling, requiring that the probability of acceptance be proportional to $\pi_t(a_k|x_k)$.

Conditioned on z_{k-1} and the induced target history h_{t-1}, define the event

$$\mathcal{E}_k := \{(x, a) : c_t\pi_t(a|x) > \mu_k(a|x)\},$$

which contributes to the bias of the estimate, because it corresponds to cases when the minimum in equation

[7] We only consider estimators for cumulative rewards (not average rewards) in this section. We assume that the division into parts is done sequentially, so that individual estimates are built from nonoverlapping sequences of T consecutive blocks of examples.

[8] As shown in Li et al. (2011), when m is constant, making T large does not necessarily reduce variance of any estimator of nonstationary policies.

(5.1) is attained by p_k. Associated with this event is the "bias mass" ε_k, which measures (up to scaling by c_t) the difference between the probability of the bad event under π_t and under the run of our algorithm:

$$\varepsilon_k := \mathbf{P}_{(x,a)\sim\pi_t}[\mathcal{E}_k] - \mathbf{P}_{(x,a)\sim\mu_k}[\mathcal{E}_k]/c_t.$$

Notice that from the definition of $\mathcal{E}_k$, this mass is nonnegative. Since the first term is a probability, this mass is at most 1. We will assume that this mass is bounded away from 1, that is, that there exists ε such that for all k and z_{k-1}

$$0 \le \varepsilon_k \le \varepsilon < 1.$$

The following theorem analyzes how much bias is introduced in the worst case, as a function of ε. It shows how the bias mass controls the bias of our estimator.

THEOREM 5.1. *For $T \ge 1$,*

$$\left|\mathbf{E}_\mu\left[\sum_{t=1}^{T} c_t \hat{V}_{B(t)}\right] - \mathbf{E}_\pi\left[\sum_{t=1}^{T} r_t\right]\right| \le \frac{T(T+1)}{2}\cdot\frac{\varepsilon}{1-\varepsilon}.$$

Intuitively, this theorem says that if a bias of ε is introduced in round t, its effect on the sum of rewards can be felt for $T-t$ rounds. Summing over rounds, we expect to get an $O(\varepsilon T^2)$ effect on the estimator of the cumulative reward. In general a very slight bias can result in a significantly better acceptance rate, and hence more replicates $\hat{V}^{(i)}_{\text{DR-ns}}$.

This theorem is the first of this sort for policy evaluators, although the mechanics of its proof have appeared in model-based reinforcement-learning (e.g., Kearns and Singh, 1998).

To prove the main theorem, we state two technical lemmas bounding the differences of probabilities and expectations under the target policy and our algorithm (for proofs of lemmas, see Appendix F). The theorem follows as their immediate consequence. Recall that $\hat{\pi}$ denotes the distribution over target histories generated by our algorithm (together with the exploration policy μ).

LEMMA 5.2. *Let $t \le T$, $k \ge 1$ and let z_{k-1} be such that the kth exploration sample marks the beginning of the tth block, that is, $\kappa(t-1) = k-1$. Let h_{t-1} and c_t be the target history and acceptance rate multiplier induced by z_{k-1}. Then*:

$$\sum_{x,a}\left|\mathbf{P}^{\mu}_{k}[x_{\kappa(t)} = x, a_{\kappa(t)} = a] - \pi_t(x,a)\right| \le \frac{2\varepsilon}{1-\varepsilon},$$

$$\left|c_t\mathbf{E}^{\mu}_{k}[\hat{V}_{B(t)}] - \mathbf{E}^{\pi}_{t}[r]\right| \le \frac{\varepsilon}{1-\varepsilon}.$$

LEMMA 5.3.

$$\sum_{h_T}\left|\hat{\pi}(h_T) - \pi(h_T)\right| \le (2\varepsilon T)/(1-\varepsilon).$$

PROOF OF THEOREM 5.1. First, bound $|\mathbf{E}_\mu[c_t \cdot \hat{V}_{B(t)}] - \mathbf{E}_\pi[r_t]|$ using the previous two lemmas, the triangle inequality and Hölder's inequality:

$$\begin{aligned}
&\left|\mathbf{E}_\mu[c_t\hat{V}_{B(t)}] - \mathbf{E}_\pi[r_t]\right| \\
&\quad = \left|\mathbf{E}_\mu[c_t\mathbf{E}^{\mu}_{\kappa(t)}[\hat{V}_{B(t)}]] - \mathbf{E}_\pi[r_t]\right| \\
&\quad \le \left|\mathbf{E}_\mu[\mathbf{E}^{\pi}_{t}[r_t]] - \mathbf{E}_\pi[\mathbf{E}^{\pi}_{t}[r_t]]\right| + \frac{\varepsilon}{1-\varepsilon} \\
&\quad = \left|\mathop{\mathbf{E}}_{h_{t-1}\sim\hat{\pi}}\left[\mathbf{E}^{\pi}_{t}\left[r-\frac{1}{2}\right]\right] - \mathop{\mathbf{E}}_{h_{t-1}\sim\pi}\left[\mathbf{E}^{\pi}_{t}\left[r-\frac{1}{2}\right]\right]\right| \\
&\qquad + \frac{\varepsilon}{1-\varepsilon} \\
&\quad \le \frac{1}{2}\sum_{h_{t-1}}\left|\hat{\pi}(h_{t-1}) - \pi(h_{t-1})\right| + \frac{\varepsilon}{1-\varepsilon} \\
&\quad \le \frac{1}{2}\cdot\frac{2\varepsilon(t-1)}{1-\varepsilon} + \frac{\varepsilon}{1-\varepsilon} = \frac{\varepsilon t}{1-\varepsilon}.
\end{aligned}$$

The theorem now follows by summing over t and using the triangle inequality. □

6. EXPERIMENTS: THE NONSTATIONARY CASE

We now study how DR-ns may achieve greater sample efficiency than rejection sampling through the use of a controlled bias. We evaluate our estimator on the problem of a multiclass multi-label classification with partial feedback using the publicly available dataset `rcv1` (Lewis et al., 2004). In this data, the goal is to predict whether a news article is in one of many Reuters categories given the contents of the article. This dataset is chosen instead of the UCI benchmarks in Section 4 because of its bigger size, which is helpful for simulating online learning (i.e., adaptive policies).

6.1 Data Generation

For multi-label dataset like `rcv1`, an example has the form $(\tilde{x}, Y)$, where $\tilde{x}$ is the covariate vector and $Y \subseteq \{1, \dots, K\}$ is the *set* of correct class labels.[9] In our modeling, we assume that any $y \in Y$ is the correct prediction for $\tilde{x}$. Similar to Section 4.1, an example $(\tilde{x}, Y)$ may be interpreted as a bandit event with context $\tilde{x}$ and loss $l(Y, a) := I(a \notin Y)$, for every action

[9] The reason why we call the covariate vector $\tilde{x}$ rather than x becomes in the sequel.

$a \in \{1, \ldots, K\}$. A classifier can be interpreted as a *stationary* policy whose expected loss is its classification error. In this section, we again aim at evaluating expected policy loss, which can be understood as negative reward. For our experiments, we only use the $K = 4$ top-level classes in `rcv1`, namely $\{C, E, G, M\}$. We take a random selection of 40,000 data points from the whole dataset and call the resulting dataset D.

To construct a partially labeled exploration dataset, we simulate a stationary but nonuniform exploration policy with a bias toward correct answers. This is meant to emulate the typical setting where a baseline system already has a good understanding of which actions are likely best. For each example $(\tilde{x}, Y)$, a uniformly random value $s(a) \in [0.1, 1]$ is assigned independently to each action a, and the final probability of action a is determined by

$$\mu_1(a|\tilde{x}, Y, s) = \frac{0.3 \times s(a)}{\sum_{a'} s(a')} + \frac{0.7 \times I(a \in Y)}{|Y|}.$$

Note that this policy will assign a nonzero probability to every action. Formally, our exploration policy is a function of an extended context $x = (\tilde{x}, Y, s)$, and our data generating distribution $D(x)$ includes the generation of the correct answers Y and values s. Of course, we will be evaluating policies π that only get to see $\tilde{x}$, but have no access to Y and s. Also, the estimator $\hat{l}$ (recall that we are evaluating loss here, not reward) is purely a function of $\tilde{x}$ and a. We stress that in a real-world setting, the exploration policy would not have access to all correct answers Y.

6.2 Evaluation of a Nonstationary Policy

As described before, a fixed (nonadaptive) classifier can be interpreted as a stationary policy. Similarly, a classifier that adapts as more data arrive is equivalent to a nonstationary policy.

In our experiments, we evaluate performance of an adaptive ϵ-*greedy* classifier defined as follows: with probability $\epsilon = 0.1$, it predicts a label drawn uniformly at random from $\{1, 2, \ldots, K\}$; with probability $1 - \epsilon$, it predicts the best label according to a linear score (the "greedy" label):

$$\operatorname*{argmax}_a \{w_a^t \cdot \tilde{x}\},$$

where $\{w_a^t\}_{a \in \{1,\ldots,K\}}$ is a set of K weight vectors at time t. This design mimics a commonly used ϵ-greedy exploration strategy for contextual bandits (e.g., Li et al., 2010). Weight vectors w_a^t are obtained by fitting a logistic regression model for the binary classification problem $a \in Y$ (positive) versus $a \notin Y$ (negative). The data used to fit w_a^t is described below. Thus, the greedy label is the most likely label according to the current set of logistic regression models. The loss estimator $\hat{l}(\tilde{x}, a)$ is also obtained by fitting a logistic regression model for $a \in Y$ versus $a \notin Y$, potentially on a different dataset.

We partition the whole data D randomly into three disjoint subsets: D_{init} (initialization set), D_{valid} (validation set), and D_{eval} (evaluation set), consisting of 1%, 19%, and 80% of D, respectively. Our goal in this experiment is to estimate the expected loss, $V_{1:T}$, of an adaptive policy π after $T = 300$ rounds.

The full-feedback set D_{init} is used to fit the loss estimator $\hat{l}$.

Since D_{valid} is a random subset of D, it may be used to simulate the behavior of policy π to obtain an *unbiased* estimate of $V_{1:T}$. We do this by taking an average of 2000 simulations of π on random shuffles of the set D_{valid}. This estimate, denoted $\bar{V}_{1:T}$, is a highly accurate approximation to (the unknown) $V_{1:T}$, and serves as our ground truth.

To assess different policy-value estimators, we randomly permute D_{eval} and transform it into a partially labeled set as described in Section 6.1. On the resulting partially labeled data, we then evaluate the policy π up to round T, obtaining an estimate of $V_{1:T}$. If the exploration history is not exhausted, we start the evaluation of π again, continuing with the next exploration sample, but restarting from empty target history (for T rounds), and repeat until we use up all the exploration data. The final estimate is the average across thus obtained replicates. We repeat this process (permutation of D_{eval}, generation of exploration history, and policy evaluation until using up all exploration data) 50 times, so that we can compare the 50 estimates against the ground truth $\bar{V}_{1:T}$ to compute bias and standard deviation of a policy-value estimator.

Finally, we describe in more detail the ε-greedy adaptive classifier π being evaluated:

- First, the policy is initialized by fitting weights w_a^0 on the full-feedback set D_{init} (similarly to $\hat{l}$). This step mimics the practical situation where one usually has prior information (in the form of either domain knowledge or historical data) to initialize a policy, instead of starting from scratch.
- After this "warm-start" step, the "online" phase begins: in each round, the policy observes a randomly selected $\tilde{x}$, predicts a label in an ϵ-greedy fashion (as described above), and then observes the corresponding 0/1 prediction loss. The policy is updated every 15 rounds. On those rounds, we retrain weights w_a^t

for each action a, using the full feedback set D_{init} as well as all the data from the online phase where the policy chose action a. The online phase terminates after $T = 300$ rounds.

6.3 Compared Evaluators

We compared the following evaluators described earlier: DM for direct method, RS for the unbiased evaluator based on rejection sampling and "replay" (Li et al., 2011), and DR-ns as in Algorithm 1 (with $c_{\max} = 1$). We also tested a variant of DR-ns, which does not monitor the quantile, but instead uses c_t equal to $\min_D \mu_1(a|x)$; we call it DR-ns-wc since it uses the worst-case (most conservative) value of c_t that ensures unbiasedness of rejection sampling.

6.4 Results

Table 3 summarizes the accuracy of different evaluators in terms of rmse (root mean squared error), bias (the absolute difference between the average estimate and the ground truth) and stdev (standard deviation of the estimates across different runs). It should be noted that, given the relatively small number of trials, the measurement of bias is not statistically significant. However, the table provides 95% confidence interval for the rmse metric that allows a meaningful comparison.

It is clear that although rejection sampling is guaranteed to be unbiased, its variance is usually the dominating part of its rmse. At the other extreme is the direct method, which has the smallest variance but often suffers large bias. In contrast, our method DR-ns is able to find a good balance between the two extremes and, with proper selection of the parameter q, is able to make the evaluation results much more accurate than others.

It is also clear that the main benefit of DR-ns is its low variance, which stems from the adaptive choice of c_t values. By slightly violating the unbiasedness guarantee, it increases the effective data size significantly, hence reducing the variance of its evaluation. For $q > 0$, DR-ns was able to extract many more trajectories of length 300 for evaluating π, while RS and DR-ns-wc were able to find only one such trajectory out of the evaluation set. In fact, if we increase the trajectory length of π from 300 to 500, both RS and DR-ns-wc are not able to construct a complete trajectory of length 500 and fail the task completely.

TABLE 3
Nonstationary policy evaluation results

Evaluator	**rmse** (±95% C.I.)	**bias**	**stdev**
DM	0.0329 ± 0.0007	0.0328	0.0027
RS	0.0179 ± 0.0050	0.0007	0.0181
DR-ns-wc	0.0156 ± 0.0037	0.0086	0.0132
DR-ns ($q = 0$)	0.0129 ± 0.0034	0.0046	0.0122
DR-ns ($q = 0.01$)	**0.0089** ± 0.0017	0.0065	0.0062
DR-ns ($q = 0.05$)	0.0123 ± 0.0017	0.0107	0.0061
DR-ns ($q = 0.1$)	0.0946 ± 0.0015	0.0946	0.0053

7. CONCLUSIONS

Doubly robust policy estimation is an effective technique which virtually always improves on the widely used inverse propensity score method. Our analysis shows that doubly robust methods tend to give more reliable and accurate estimates, for evaluating both stationary and nonstationary policies. The theory is corroborated by experiments on benchmark data as well as two large-scale real-world problems. In the future, we expect the DR technique to become common practice in improving contextual bandit algorithms.

APPENDIX A: PROOFS OF LEMMAS 3.1–3.3

Throughout proofs in this appendix, we write $\hat{r}$ and r^* instead of $\hat{r}(x,a)$ and $r^*(x,a)$ when x and a are clear from the context, and similarly for Δ and ϱ_k.

LEMMA 3.1. *The range of $\hat{V}_k$ is bounded as*

$$|\hat{V}_k| \le 1 + M.$$

PROOF.

$$\begin{aligned}|\hat{V}_k| &= \left|\hat{r}(x_k, \nu) + \frac{\nu(a_k|x_k)}{\hat{\mu}_k(a_k|x_k)} \cdot \bigl(r_k - \hat{r}(x_k, a_k)\bigr)\right| \\ &\le |\hat{r}(x_k, \nu)| + \frac{\nu(a_k|x_k)}{\hat{\mu}_k(a_k|x_k)} \cdot |r_k - \hat{r}(x_k, a_k)| \\ &\le 1 + M,\end{aligned}$$

where the last inequality follows because $\hat{r}$ and r_k are bounded in [0, 1]. □

LEMMA 3.2. *The expectation of the term $\hat{V}_k$ is*

$$\mathbf{E}_k^{\mu}[\hat{V}_k] = \mathop{\mathbf{E}}_{(x,a)\sim\nu}\bigl[r^*(x,a) + \bigl(1 - \varrho_k(x,a)\bigr)\Delta(x,a)\bigr].$$

PROOF.

$$\mathbf{E}_k^{\mu}[\hat{V}_k] = \mathop{\mathbf{E}}_{(x,a,r)\sim\mu_k}\left[\hat{r}(x,\nu) + \frac{\nu(a|x)}{\mu_k(a|x)} \cdot \varrho_k \cdot (r - \hat{r})\right]$$

$$
\begin{aligned}
&= \mathop{\mathbf{E}}_{x\sim D}\big[\hat{r}(x,\nu)\big] \\
&\quad + \mathop{\mathbf{E}}_{x\sim D}\bigg[\sum_{a\in\mathcal{A}}\mu_k(a|x)\mathop{\mathbf{E}}_{r\sim D(\cdot|x,a)}\bigg[\frac{\nu(a|x)}{\mu_k(a|x)} \\
&\qquad\qquad \cdot\varrho_k\cdot(r-\hat{r})\bigg]\bigg] \\
&= \mathop{\mathbf{E}}_{x\sim D}\big[\hat{r}(x,\nu)\big] \\
&\quad + \mathop{\mathbf{E}}_{x\sim D}\bigg[\sum_{a\in\mathcal{A}}\nu(a|x)\mathop{\mathbf{E}}_{r\sim D(\cdot|x,a)}\big[\varrho_k\cdot(r-\hat{r})\big]\bigg] \\
&= \mathop{\mathbf{E}}_{(x,a)\sim\nu}[\hat{r}] + \mathop{\mathbf{E}}_{(x,a,r)\sim\nu}\big[\varrho_k\cdot(r-\hat{r})\big] \\
&= \mathop{\mathbf{E}}_{(x,a)\sim\nu}\big[r^* + (\hat{r}-r^*) + \varrho_k\cdot(r^*-\hat{r})\big] \qquad \text{(A.1)} \\
&= \mathop{\mathbf{E}}_{(x,a)\sim\nu}\big[r^* + (1-\varrho_k)\Delta\big]. \qquad \square
\end{aligned}
$$

LEMMA 3.3. *The variance of the term $\hat{V}_k$ can be decomposed and bounded as follows*:

(i) $\mathbf{V}_k^{\mu}[\hat{V}_k]$

$$
\begin{aligned}
&= \mathop{\mathbf{V}}_{x\sim D}\Big[\mathop{\mathbf{E}}_{a\sim\nu(\cdot|x)}\big[r^*(x,a) \\
&\qquad\qquad + \big(1-\varrho_k(x,a)\big)\Delta(x,a)\big]\Big] \\
&\quad - \mathop{\mathbf{E}}_{x\sim D}\Big[\mathop{\mathbf{E}}_{a\sim\nu(\cdot|x)}\big[\varrho_k(x,a)\Delta(x,a)\big]^2\Big] \\
&\quad + \mathop{\mathbf{E}}_{(x,a)\sim\nu}\bigg[\frac{\nu(a|x)}{\hat{\mu}_k(a|x)}\cdot\varrho_k(x,a)\cdot\mathop{\mathbf{V}}_{r\sim D(\cdot|x,a)}[r]\bigg] \\
&\quad + \mathop{\mathbf{E}}_{(x,a)\sim\nu}\bigg[\frac{\nu(a|x)}{\hat{\mu}_k(a|x)}\cdot\varrho_k(x,a)\Delta(x,a)^2\bigg].
\end{aligned}
$$

(ii) $\mathbf{V}_k^{\mu}[\hat{V}_k]$

$$
\begin{aligned}
&\le \mathop{\mathbf{V}}_{x\sim D}\big[r^*(x,\nu)\big] \\
&\quad + 2\mathop{\mathbf{E}}_{(x,a)\sim\nu}\big[\big|\big(1-\varrho_k(x,a)\big)\Delta(x,a)\big|\big] \\
&\quad + M\mathop{\mathbf{E}}_{(x,a)\sim\nu}\Big[\varrho_k(x,a) \\
&\qquad\qquad \cdot\mathop{\mathbf{E}}_{r\sim D(\cdot|x,a)}\big[\big(r-\hat{r}(x,a)\big)^2\big]\Big].
\end{aligned}
$$

PROOF.

$$
\begin{aligned}
\mathbf{E}_k^{\mu}[\hat{V}_k^2] &= \mathop{\mathbf{E}}_{(x,a,r)\sim\mu_k}\bigg[\bigg(\hat{r}(x,\nu) + \frac{\nu(a|x)}{\mu_k(a|x)}\cdot\varrho_k \\
&\qquad\qquad \cdot(r-\hat{r})\bigg)^2\bigg] \\
&= \mathop{\mathbf{E}}_{x\sim D}\big[\hat{r}(x,\nu)^2\big] \\
&\quad + 2\mathop{\mathbf{E}}_{(x,a,r)\sim\mu_k}\bigg[\hat{r}(x,\nu) \\
&\qquad\qquad \cdot\frac{\nu(a|x)}{\mu_k(a|x)}\cdot\varrho_k\cdot(r-\hat{r})\bigg] \\
&\quad + \mathop{\mathbf{E}}_{(x,a,r)\sim\mu_k}\bigg[\frac{\nu(a|x)}{\mu_k(a|x)} \\
&\qquad\qquad \cdot\frac{\nu(a|x)}{\hat{\mu}_k(a|x)}\cdot\varrho_k\cdot(r-\hat{r})^2\bigg] \\
&= \mathop{\mathbf{E}}_{x\sim D}\big[\hat{r}(x,\nu)^2\big] \qquad \text{(A.2)} \\
&\quad + 2\mathop{\mathbf{E}}_{(x,a,r)\sim\nu}\big[\hat{r}(x,\nu)\cdot\varrho_k\cdot(r-\hat{r})\big] \\
&\quad + \mathop{\mathbf{E}}_{(x,a,r)\sim\nu}\bigg[\frac{\nu(a|x)}{\hat{\mu}_k(a|x)}\cdot\varrho_k\cdot(r-\hat{r})^2\bigg] \\
&= \mathop{\mathbf{E}}_{(x,a)\sim\nu}\big[\big(\hat{r}(x,\nu)-\varrho_k\Delta\big)^2\big] \qquad \text{(A.3)} \\
&\quad - \mathop{\mathbf{E}}_{(x,a)\sim\nu}\big[\varrho_k^2\Delta^2\big] + E,
\end{aligned}
$$

where E denotes the term

$$
E := \mathop{\mathbf{E}}_{(x,a,r)\sim\nu}\bigg[\frac{\nu(a|x)}{\hat{\mu}_k(a|x)}\cdot\varrho_k\cdot(r-\hat{r})^2\bigg].
$$

To obtain an expression for the variance of $\hat{V}_k$, first note that by equation (A.1),

$$
\mathbf{E}_k^{\mu}[\hat{V}_k] = \mathop{\mathbf{E}}_{(x,a)\sim\nu}\big[\hat{r}(x,\nu)-\varrho_k\Delta\big]. \tag{A.4}
$$

Combining this with equation (A.3), we obtain

$$
\begin{aligned}
\mathbf{V}_k^{\mu}[\hat{V}_k] &= \mathop{\mathbf{V}}_{(x,a)\sim\nu}\big[\hat{r}(x,\nu)-\varrho_k\Delta\big] \\
&\quad - \mathop{\mathbf{E}}_{(x,a)\sim\nu}\big[\varrho_k^2\Delta^2\big] + E \\
&= \mathop{\mathbf{V}}_{x\sim D}\Big[\mathop{\mathbf{E}}_{a\sim\nu(\cdot|x)}\big[\hat{r}(x,\nu)-\varrho_k\Delta\big]\Big] \\
&\quad + \mathop{\mathbf{E}}_{x\sim D}\Big[\mathop{\mathbf{V}}_{a\sim\nu(\cdot|x)}\big[\hat{r}(x,\nu)-\varrho_k\Delta\big]\Big] \\
&\quad - \mathop{\mathbf{E}}_{x\sim D}\Big[\mathop{\mathbf{V}}_{a\sim\nu(\cdot|x)}[\varrho_k\Delta]\Big] \\
&\quad - \mathop{\mathbf{E}}_{x\sim D}\Big[\mathop{\mathbf{E}}_{a\sim\nu(\cdot|x)}[\varrho_k\Delta]^2\Big] + E \\
&= \mathop{\mathbf{V}}_{x\sim D}\Big[\mathop{\mathbf{E}}_{a\sim\nu(\cdot|x)}\big[r^* + (1-\varrho_k)\Delta\big]\Big] \\
&\quad + \mathop{\mathbf{E}}_{x\sim D}\Big[\mathop{\mathbf{V}}_{a\sim\nu(\cdot|x)}[\varrho_k\Delta]\Big] \\
&\quad - \mathop{\mathbf{E}}_{x\sim D}\Big[\mathop{\mathbf{V}}_{a\sim\nu(\cdot|x)}[\varrho_k\Delta]\Big] \\
&\quad - \mathop{\mathbf{E}}_{x\sim D}\Big[\mathop{\mathbf{E}}_{a\sim\nu(\cdot|x)}[\varrho_k\Delta]^2\Big] + E
\end{aligned}
$$

$$
\begin{aligned}
&= \mathop{\mathbf{V}}_{x\sim D}\Big[\mathop{\mathbf{E}}_{a\sim\nu(\cdot|x)}[r^* + (1-\varrho_k)\Delta]\Big] \\
&\quad - \mathop{\mathbf{E}}_{x\sim D}\Big[\mathop{\mathbf{E}}_{a\sim\nu(\cdot|x)}[\varrho_k\Delta]^2\Big] + E.
\end{aligned}
$$

We now obtain part (i) of the lemma by decomposing the term E:

$$
\begin{aligned}
E &= \mathop{\mathbf{E}}_{(x,a,r)\sim\nu}\left[\frac{\nu(a|x)}{\hat{\mu}_k(a|x)}\cdot\varrho_k\cdot\big(r-r^*\big)^2\right] \\
&\quad + \mathop{\mathbf{E}}_{(x,a)\sim\nu}\left[\frac{\nu(a|x)}{\hat{\mu}_k(a|x)}\cdot\varrho_k\cdot\big(r^*-\hat{r}\big)^2\right] \\
&= \mathop{\mathbf{E}}_{(x,a)\sim\nu}\left[\frac{\nu(a|x)}{\hat{\mu}_k(a|x)}\cdot\varrho_k\cdot\mathop{\mathbf{V}}_{r\sim D(\cdot|x,a)}[r]\right] \\
&\quad + \mathop{\mathbf{E}}_{(x,a)\sim\nu}\left[\frac{\nu(a|x)}{\hat{\mu}_k(a|x)}\cdot\varrho_k\Delta^2\right].
\end{aligned}
$$

To prove part (ii) of the lemma, first note that

$$
\begin{aligned}
\hat{r}(x,\nu)^2 &= \Big(r^*(x,\nu) + \mathop{\mathbf{E}}_{a\sim\nu(\cdot|x)}\big[\Delta(x,a)\big]\Big)^2 \\
&= r^*(x,\nu)^2 + 2r^*(x,\nu)\mathop{\mathbf{E}}_{a\sim\nu(\cdot|x)}\big[\Delta(x,a)\big] \\
&\quad + \mathop{\mathbf{E}}_{a\sim\nu(\cdot|x)}\big[\Delta(x,a)\big]^2 \\
&= r^*(x,\nu)^2 + 2\hat{r}(x,\nu)\mathop{\mathbf{E}}_{a\sim\nu(\cdot|x)}\big[\Delta(x,a)\big] \\
&\quad - \mathop{\mathbf{E}}_{a\sim\nu(\cdot|x)}\big[\Delta(x,a)\big]^2 \\
&\le r^*(x,\nu)^2 + 2\hat{r}(x,\nu)\mathop{\mathbf{E}}_{a\sim\nu(\cdot|x)}\big[\Delta(x,a)\big].
\end{aligned}
$$

Plugging this in equation (A.2), we obtain

$$
\begin{aligned}
\mathbf{E}_k^{\mu}\big[\hat{V}_k^2\big] &= \mathop{\mathbf{E}}_{x\sim D}\big[\hat{r}(x,\nu)^2\big] \\
&\quad + 2\mathop{\mathbf{E}}_{(x,a,r)\sim\nu}\big[\hat{r}(x,\nu)\cdot\varrho_k\cdot(r-\hat{r})\big] + E \\
&\le \mathop{\mathbf{E}}_{x\sim D}\big[r^*(x,\nu)^2\big] \\
&\quad + 2\mathop{\mathbf{E}}_{x\sim D}\Big[\hat{r}(x,\nu)\mathop{\mathbf{E}}_{a\sim\nu(\cdot|x)}[\Delta]\Big] \\
&\quad + 2\mathop{\mathbf{E}}_{(x,a)\sim\nu}\big[\hat{r}(x,\nu)\cdot(-\varrho_k)\cdot\Delta\big] + E
\end{aligned}
$$

$$
\begin{aligned}
&= \mathop{\mathbf{E}}_{x\sim D}\big[r^*(x,\nu)^2\big] \\
&\quad + 2\mathop{\mathbf{E}}_{(x,a)\sim\nu}\big[\hat{r}(x,\nu)\cdot(1-\varrho_k)\cdot\Delta\big] + E.
\end{aligned} \tag{A.5}
$$

On the other hand, equation (A.4) can be rewritten as

$$
\mathbf{E}_k^{\mu}[\hat{V}_k] = \mathop{\mathbf{E}}_{(x,a)\sim\nu}\big[r^*(x,\nu) + (1-\varrho_k)\Delta\big].
$$

Combining with equation (A.5), we obtain

$$
\begin{aligned}
\mathbf{V}_k^{\mu}[\hat{V}_k] &\le \mathop{\mathbf{V}}_{x\sim D}\big[r^*(x,\nu)\big] \\
&\quad + 2\mathop{\mathbf{E}}_{(x,a)\sim\nu}\big[\hat{r}(x,\nu)\cdot(1-\varrho_k)\Delta\big] \\
&\quad - 2\mathop{\mathbf{E}}_{x\sim D}\big[r^*(x,\nu)\big]\mathop{\mathbf{E}}_{(x,a)\sim\nu}\big[(1-\varrho_k)\Delta\big] \\
&\quad - \mathop{\mathbf{E}}_{(x,a)\sim\nu}\big[(1-\varrho_k)\Delta\big]^2 + E \\
&\le \mathop{\mathbf{V}}_{x\sim D}\big[r^*(x,\nu)\big] \\
&\quad + 2\mathop{\mathbf{E}}_{(x,a)\sim\nu}\big[\big(\hat{r}(x,\nu)-\tfrac{1}{2}\big)(1-\varrho_k)\Delta\big] \\
&\quad - 2\mathop{\mathbf{E}}_{x\sim D}\big[r^*(x,\nu)-\tfrac{1}{2}\big]\mathop{\mathbf{E}}_{(x,a)\sim\nu}\big[(1-\varrho_k)\Delta\big] \\
&\quad + E \\
&\le \mathop{\mathbf{V}}_{x\sim D}\big[r^*(x,\nu)\big] + \mathop{\mathbf{E}}_{(x,a)\sim\nu}\big[\big|(1-\varrho_k)\Delta\big|\big] \\
&\quad + \Big|\mathop{\mathbf{E}}_{(x,a)\sim\nu}\big[(1-\varrho_k)\Delta\big]\Big| + E,
\end{aligned}
$$

where the last inequality follows by Hölder's inequality and the observations that $|\hat{r}-1/2|\le 1/2$ and $|r^*-1/2|\le 1/2$. Part (ii) now follows by the bound

$$
\begin{aligned}
E &= \mathop{\mathbf{E}}_{(x,a,r)\sim\nu}\left[\frac{\nu(a|x)}{\hat{\mu}_k(a|x)}\cdot\varrho_k\cdot(r-\hat{r})^2\right] \\
&\le M\mathop{\mathbf{E}}_{(x,a)\sim\nu}\Big[\varrho_k\mathop{\mathbf{E}}_{r\sim D(\cdot|x,a)}\big[(r-\hat{r})^2\big]\Big]. \qquad \square
\end{aligned}
$$

APPENDIX B: FREEDMAN'S INEQUALITY

The following is a corollary of Theorem 1 of Beygelzimer et al. (2011). It can be viewed as a version of Freedman's inequality Freedman's (1975). Let $y_1,\dots,y_n$ be a sequence of real-valued random variables. Let $\mathbf{E}_k$ denote $\mathbf{E}[\,\cdot\,|y_1,\dots,y_{k-1}]$ and $\mathbf{V}_k$ conditional variance.

THEOREM B.1. *Let $V, D\in\mathbb{R}$ such that*

$$
\sum_{k=1}^{n}\mathbf{V}_k[y_k]\le V,
$$

and for all k, $|y_k-\mathbf{E}_k[y_k]|\le D$. Then for any $\delta>0$, with probability at least $1-\delta$,

$$
\left|\sum_{k=1}^{n}y_k-\sum_{k=1}^{n}\mathbf{E}_k[y_k]\right|\le 2\max\big\{D\ln(2/\delta),\sqrt{V\ln(2/\delta)}\big\}.
$$

APPENDIX C: IMPROVED FINITE-SAMPLE ERROR BOUND

In this appendix, we analyze the error of $\hat{V}_{\mathrm{DR}}$ in estimating the value of a stationary policy ν. We generalize the analysis of Section 3.4.3 by replacing conditions on the ranges of variables by conditions on the moments.

For a function $f : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$ and $1 \le p < \infty$, we define the $L_p(\nu)$ norm as usual:

$$\|f\|_{p,\nu} = \mathbf{E}_{(x,a)\sim\nu}\big[|f(x,a)|^p\big]^{1/p}.$$

For $p = \infty$, $\|f\|_{\infty,\nu}$ is the essential supremum of $|f|$ under ν.

As in Section 3.4.3, we first simplify Lemmas 3.1–3.3, and then apply Freedman's inequality to obtain a specific error bound.

LEMMA C.1. *Let $1 \le p, q \le \infty$ be such that $1/p + 1/q = 1$. Assume there are finite constants $M, e_{\hat{r}}, \delta_\Delta, \delta_\varrho, \varrho_{\max} \ge 0$ such that with probability one under μ, for all k:*

$$\begin{aligned}\nu(a_k|x_k)/\hat{\mu}_k(a_k|x_k) &\le M,\\ \|\Delta\|_{q,\nu} &\le \delta_\Delta,\\ \|1-\varrho_k\|_{p,\nu} &\le \delta_\varrho,\\ \|\varrho_k\|_{p,\nu} &\le \varrho_{\max},\\ \mathop{\mathbf{E}}_{(x,a)\sim\nu}\Big[\mathop{\mathbf{E}}_{r\sim D(\cdot|x,a)}\big[(\hat{r}(x,a)-r)^2\big]^q\Big]^{1/q} &\le e_{\hat{r}}.\end{aligned}$$

Then with probability one under μ, for all k:

$$\begin{aligned}&\big|\mathbf{E}_k^\mu[\hat{V}_k] - V\big| \le \delta_\varrho\delta_\Delta,\\ &\qquad \mathbf{V}_k^\mu[\hat{V}_k] \le \mathbf{V}_{x\sim D}\big[r^*(x,\nu)\big] + 2\delta_\varrho\delta_\Delta + M\varrho_{\max}e_{\hat{r}}.\end{aligned}$$

PROOF. The bias and variance bound follow from Lemma 3.2 and Lemma 3.3(ii), respectively, by Hölder's inequality. □

THEOREM C.2. *If assumptions of Lemma C.1 hold, then with probability at least $1-\delta$,*

$$\begin{aligned}&|\hat{V}_{\mathrm{DR}} - V|\\ &\quad \le \delta_\varrho\delta_\Delta\\ &\qquad + 2\max\Bigg\{\frac{(1+M)\ln(2/\delta)}{n},\\ &\qquad \sqrt{\frac{(\mathbf{V}_{x\sim D}[r^*(x,\nu)] + 2\delta_\varrho\delta_\Delta + M\varrho_{\max}e_{\hat{r}})\ln(2/\delta)}{n}}\Bigg\}.\end{aligned}$$

PROOF. The proof follows by Freedman's inequality (Theorem B.1 in Appendix B), applied to random variables $\hat{V}_k$, whose range and variance are bounded using Lemma 3.1 and C.1. □

APPENDIX D: DIRECT LOSS MINIMIZATION

Given cost-sensitive multiclass classification data $\{(x, l_1, \dots, l_K)\}$, we perform approximate gradient descent on the policy loss (or classification error). In the experiments of Section 4.1, policy ν is specified by K weight vectors $\theta_1, \dots, \theta_K$. Given $x \in \mathcal{X}$, the policy predicts as follows: $\nu(x) = \mathrm{argmax}_{a\in\{1,\dots,K\}}\{x \cdot \theta_a\}$.

To optimize θ_a, we adapt the "toward-better" version of the direct loss minimization method of McAllester, Hazan and Keshet (2011) as follows: given any data point $(x, l_1, \dots, l_K)$ and the current weights θ_a, the weights are adjusted by

$$\begin{aligned}\theta_{a_1} &\leftarrow \theta_{a_1} + \eta x,\\ \theta_{a_2} &\leftarrow \theta_{a_2} - \eta x,\end{aligned}$$

where $a_1 = \mathrm{argmax}_a\{x \cdot \theta_a - \epsilon l_a\}$, $a_2 = \mathrm{argmax}_a\{x \cdot \theta_a\}$, $\eta \in (0, 1)$ is a decaying learning rate, and $\epsilon > 0$ is an input parameter.

For computational reasons, we actually perform batch updates rather than incremental updates. Updates continue until the weights converge. We found that the learning rate $\eta = t^{-0.3}/2$, where t is the batch iteration, worked well across all datasets. The parameter ϵ was fixed to 0.1 for all datasets.

Furthermore, since the policy loss is not convex in the weight vectors, we repeat the algorithm 20 times with randomly perturbed starting weights and then return the best run's weight according to the learned policy's loss in the training data. We also tried using a holdout validation set for choosing the best weights out of the 20 candidates, but did not observe benefits from doing so.

APPENDIX E: FILTER TREE

The Filter Tree (Beygelzimer, Langford and Ravikumar, 2008) is a reduction from multiclass cost-sensitive classification to binary classification. Its input is of the same form as for Direct Loss Minimization, but its output is a Filter Tree: a decision tree, where each inner node is itself implemented by some binary classifier (called base classifier), and leaves correspond to classes of the original multiclass problem. As base classifiers we used J48 decision trees implemented in Weka 3.6.4 (Hall et al., 2009). Thus, there are 2-class decision trees in the nodes, with the nodes arranged as per a Filter Tree. Training in a Filter Tree proceeds bottom-up, but the classification in a trained Filter Tree proceeds root-to-leaf, with the running time logarithmic in the number of classes. We did not test the all-pairs Filter Tree, which classifies examples in the time linear in the number of classes, similar to DLM.

APPENDIX F: PROOFS OF LEMMAS 5.2 AND 5.3

LEMMA 5.2. *Let $t \le T$, $k \ge 1$ and let z_{k-1} be such that the kth exploration sample marks the beginning of the tth block, that is, $\kappa(t-1) = k-1$. Let h_{t-1} and c_t be the target history and acceptance rate multiplier induced by z_{k-1}. Then*:

$$\sum_{x,a} \bigl| \mathbf{P}_k^\mu[x_{\kappa(t)} = x, a_{\kappa(t)} = a] - \pi_t(x,a) \bigr| \le \frac{2\varepsilon}{1-\varepsilon}, \tag{F.1}$$

$$\bigl| c_t \mathbf{E}_k^\mu[\hat{V}_{B(t)}] - \mathbf{E}_t^\pi[r] \bigr| \le \frac{\varepsilon}{1-\varepsilon}. \tag{F.2}$$

PROOF. We begin by showing equation (F.1). Consider the mth exploration sample $(x,a) \sim \mu_m$ and assume that this sample is in the tth block. The probability of accepting this sample is

$$\begin{aligned}
&\mathbf{P}_{u \sim \mu_m(\cdot|x,a)}\biggl[u \le \frac{c_t \pi_t(a|x)}{\mu_m(a|x)}\biggr] \\
&\quad = I\bigl[(x,a) \in \mathcal{E}_m\bigr] + \frac{c_t \pi_t(a|x)}{\mu_m(a|x)} I\bigl[(x,a) \notin \mathcal{E}_m\bigr],
\end{aligned}$$

where $I[\cdot]$ is the indicator function equal to 1 when its argument is true and 0 otherwise. The probability of seeing and accepting a sample (x,a) from μ_m is

$$\begin{aligned}
&\text{accept}_m(x,a) \\
&\quad := \mu_m(x,a)\biggl(I\bigl[(x,a) \in \mathcal{E}_m\bigr] \\
&\qquad\qquad + \frac{c_t \pi_t(a|x)}{\mu_m(a|x)} I\bigl[(x,a) \notin \mathcal{E}_m\bigr]\biggr) \\
&\quad = \mu_m(x,a) I\bigl[(x,a) \in \mathcal{E}_m\bigr] \\
&\qquad + c_t \pi_t(x,a) I\bigl[(x,a) \notin \mathcal{E}_m\bigr] \\
&\quad = c_t \pi_t(x,a) \\
&\qquad - \bigl(c_t \pi_t(x,a) - \mu_m(x,a)\bigr) I\bigl[(x,a) \in \mathcal{E}_m\bigr]
\end{aligned}$$

and the marginal probability of accepting a sample from μ_m is

$$\begin{aligned}
\text{accept}_m(*) &:= \sum_{x,a} \text{accept}_m(x,a) \\
&= c_t - c_t \varepsilon_m = c_t(1 - \varepsilon_m).
\end{aligned}$$

In order to accept the mth exploration sample, samples k through $m-1$ must be rejected. The probability of eventually accepting (x,a), conditioned on z_{k-1} is therefore

$$\begin{aligned}
&\mathbf{P}_k^\mu(x_{\kappa(t)} = x, a_{\kappa(t)} = a) \\
&\quad = \mathbf{E}_k^\mu\Biggl[\sum_{m \ge k}^{\infty} \text{accept}_m(x,a) \prod_{k'=k}^{m-1} \bigl(1 - \text{accept}_{k'}(*)\bigr)\Biggr] \\
&\quad = c_t \pi_t(x,a) \\
&\qquad \cdot \mathbf{E}_k^\mu\Biggl[\sum_{m \ge k}^{\infty} \prod_{k'=k}^{m-1} \bigl(1 - \text{accept}_{k'}(*)\bigr)\Biggr] \qquad \text{(F.3)} \\
&\qquad - \mathbf{E}_k^\mu\Biggl[\sum_{m \ge k}^{\infty} \bigl(c_t \pi_t(x,a) - \mu_m(x,a)\bigr) \qquad \text{(F.4)} \\
&\qquad\qquad \cdot I\bigl[(x,a) \in \mathcal{E}_m\bigr] \\
&\qquad\qquad \cdot \prod_{k'=k}^{m-1} \bigl(1 - \text{accept}_{k'}(*)\bigr)\Biggr].
\end{aligned}$$

To bound $|\mathbf{P}_k^\mu[x_{\kappa(t)} = x, a_{\kappa(t)} = a] - \pi_t(x,a)|$ and prove equation (F.1), we first need to bound equations (F.3) and (F.4). Note that from the definition of $\mathcal{E}_m$, the expression inside the expectation of equation (F.4) is always nonnegative. Let $E_1(x,a)$ denote the expression in equation (F.3) and $E_2(x,a)$ the expression in equation (F.4). We bound $E_1(x,a)$ and $E_2(x,a)$ separately, using bounds $0 \le \varepsilon_m \le \varepsilon$:

$$\begin{aligned}
E_1(x,a) &= c_t \pi_t(x,a) \mathbf{E}_k^\mu\Biggl[\sum_{m \ge k}^{\infty} \prod_{k'=k}^{m-1} \bigl(1 - \text{accept}_{k'}(*)\bigr)\Biggr] \\
&\le c_t \pi_t(x,a) \mathbf{E}_k^\mu\Biggl[\sum_{m \ge k}^{\infty} \prod_{k'=k}^{m-1} \bigl(1 - c_t(1-\varepsilon)\bigr)\Biggr] \\
&= \frac{\pi_t(x,a)}{1-\varepsilon}, \\
E_1(x,a) &\ge c_t \pi_t(x,a) \mathbf{E}_k^\mu\Biggl[\sum_{m \ge k}^{\infty} \prod_{k'=k}^{m-1} (1 - c_t)\Biggr] \\
&= \pi_t(x,a), \\
E_2(x,a) &= \mathbf{E}_k^\mu\Biggl[\sum_{m \ge k}^{\infty} \bigl(c_t \pi_t(x,a) - \mu_m(x,a)\bigr) \\
&\qquad \cdot I\bigl[(x,a) \in \mathcal{E}_m\bigr] \\
&\qquad \cdot \prod_{k'=k}^{m-1} \bigl(1 - \text{accept}_{k'}(*)\bigr)\Biggr]
\end{aligned}$$

$$\le \mathbf{E}_k^{\mu}\Bigg[\sum_{m\ge k}^{\infty}\big(c_t\pi_t(x,a)-\mu_m(x,a)\big) \cdot I\big[(x,a)\in\mathcal{E}_m\big] \cdot \big(1-c_t(1-\varepsilon)\big)^{m-k}\Bigg].$$

Now we are ready to prove equation (F.1):

$$\begin{aligned}
&\sum_{x,a}|\mathbf{P}_k^{\mu}[x_{\kappa(t)}=x, a_{\kappa(t)}=a]-\pi_t(x,a)|\\
&\quad=\sum_{x,a}|E_1(x,a)-\pi_t(x,a)-E_2(x,a)|\\
&\quad\le\sum_{x,a}|E_1(x,a)-\pi_t(x,a)|+\sum_{x,a}E_2(x,a)\\
&\quad\le\sum_{x,a}\frac{\pi_t(x,a)\varepsilon}{1-\varepsilon}\\
&\qquad+\mathbf{E}_k^{\mu}\Bigg[\sum_{m\ge k}^{\infty}\sum_{x,a}\big(c_t\pi_t(x,a)-\mu_m(x,a)\big) \cdot I\big[(x,a)\in\mathcal{E}_m\big]\big(1-c_t(1-\varepsilon)\big)^{m-k}\Bigg]\\
&\quad=\frac{\varepsilon}{1-\varepsilon}+\mathbf{E}_k^{\mu}\Bigg[\sum_{m\ge k}^{\infty}c_t\varepsilon_m\big(1-c_t(1-\varepsilon)\big)^{m-k}\Bigg]\\
&\quad\le\frac{2\varepsilon}{1-\varepsilon}
\end{aligned}$$

proving equation (F.1).

Let reach$_m$ denote the indicator of the event that the mth sample is in block t (i.e., samples $k, k+1, \dots, m-1$ are rejected). Then

$$\begin{aligned}
\mathbf{E}_k^{\mu}[\hat{V}_{B(t)}]&=\sum_{m=k}^{\infty}\mathbf{E}_k^{\mu}[\hat{V}_m\,\mathrm{reach}_m]\\
&=\sum_{m=k}^{\infty}\mathbf{E}_k^{\mu}[\mathbf{E}_m^{\mu}[\hat{V}_m\,\mathrm{reach}_m]]\\
&=\sum_{m=k}^{\infty}\mathbf{E}_k^{\mu}[\mathrm{reach}_m\,\mathbf{E}_m^{\mu}[\hat{V}_m]], \qquad \text{(F.5)}
\end{aligned}$$

where equation (F.5) follows because the event of reaching the mth sample depends only on the preceding samples, and hence it is a deterministic function of z_{m-1}. Plugging Lemma 3.2 in equation (F.5), we obtain

$$\begin{aligned}
&c_t\mathbf{E}_k^{\mu}[\hat{V}_{B(t)}]\\
&\quad=c_t\mathop{\mathbf{E}}_{r\sim\pi_t}[r]\sum_{m=k}^{\infty}\mathbf{E}_k^{\mu}[\mathrm{reach}_m]\\
&\quad=c_t\mathop{\mathbf{E}}_{r\sim\pi_t}[r]\mathbf{E}_k^{\mu}\Bigg[\sum_{m=k}^{\infty}\prod_{k'=k}^{m-1}\big(1-\mathrm{accept}_{k'}(*)\big)\Bigg]
\end{aligned}$$

(because $\mathbf{E}_{r\sim\pi_t}[r]$ is a deterministic function of z_{k-1}). This can be bounded, similarly as before, as

$$\mathop{\mathbf{E}}_{r\sim\pi_t}[r]\le c_t\mathbf{E}_k^{\mu}[\hat{V}_{B(t)}]\le\frac{\mathbf{E}_{r\sim\pi_t}[r]}{1-\varepsilon}$$

yielding equation (F.2). □

LEMMA 5.3.

$$\sum_{h_T}\big|\hat{\pi}(h_T)-\pi(h_T)\big|\le(2\varepsilon T)/(1-\varepsilon).$$

PROOF. We prove the lemma by induction and the triangle inequality (essentially following Kakade, Kearns and Langford, 2003). The lemma holds for $T=0$ since there is only one empty history (and hence both $\hat{\pi}$ and π are point distributions over h_0). Now assume the lemma holds for $T-1$. We prove it for T:

$$\begin{aligned}
&\sum_{h_T}\big|\hat{\pi}(h_T)-\pi(h_T)\big|\\
&\quad=\sum_{h_{T-1}}\sum_{(x_T,a_T,r_T)}\big|\hat{\pi}(h_{T-1})\hat{\pi}_T(x_T,a_T,r_T)-\pi(h_{T-1})\pi_T(x_T,a_T,r_T)\big|\\
&\quad\le\sum_{h_{T-1}}\sum_{(x_T,a_T,r_T)}\big(\big|\hat{\pi}(h_{T-1})\hat{\pi}_T(x_T,a_T,r_T)-\hat{\pi}(h_{T-1})\pi_T(x_T,a_T,r_T)\big|\\
&\qquad\qquad+\big|\hat{\pi}(h_{T-1})\pi_T(x_T,a_T,r_T)-\pi(h_{T-1})\pi_T(x_T,a_T,r_T)\big|\big)\\
&\quad=\mathop{\mathbf{E}}_{h_{T-1}\sim\hat{\pi}}\Bigg[\sum_{(x_T,a_T,r_T)}\big|\hat{\pi}_T(x_T,a_T,r_T)-\pi_T(x_T,a_T,r_T)\big|\Bigg]\\
&\qquad+\sum_{h_{T-1}}\big|\hat{\pi}(h_{T-1})-\pi(h_{T-1})\big|\\
&\quad\le\frac{2\varepsilon}{1-\varepsilon}+\frac{2\varepsilon(T-1)}{1-\varepsilon}=\frac{2\varepsilon T}{1-\varepsilon}.
\end{aligned}$$

□

APPENDIX G: PROGRESSIVE VALIDATION POLICY

In Section 4.1.3, we showed how the stationary DR estimator can be used not only for policy evaluation, but also for policy optimization by transforming the contextual bandit problem into a cost sensitive classification problem.

In this appendix, we show how the *nonstationary* DR estimator, when applied to an online learning algorithm, can also be used to obtain a high-performing *stationary* policy. The value of this policy concentrates around the average per-step reward estimated for the online learning algorithm. Thus, to the extent that the online algorithm achieves a high reward, so does this stationary policy. The policy is constructed using the ideas behind the "progressive validation" error bound (Blum, Kalai and Langford, 1999), and hence we call it a "progressive validation policy."

Assume that the algorithm DR-ns successfully terminates after generating T blocks. The *progressive validation policy* is the randomized stationary policy π_{PV} defined as

$$\pi_{\mathrm{PV}}(a|x) := \sum_{t=1}^{T} \frac{c_t|B(t)|}{C} \pi(a|x, h_{t-1}).$$

Conceptually, this policy first picks among the histories $h_0, \dots, h_{T-1}$ with probabilities $c_1|B(1)|/C, \dots, c_t|B(T)|/C$, and then executes the policy π given the chosen history. We extend π_{PV} to a distribution over triples

$$\pi_{\mathrm{PV}}(x, a, r) = D(x)\pi_{\mathrm{PV}}(a|x)D(r|x, a).$$

We will show that the average reward estimator $\hat{V}^{\mathrm{avg}}_{\mathrm{DR\text{-}ns}}$ returned by our algorithm estimates the expected reward of π_{PV} with an error $O(1/\sqrt{N})$ where N is the number of exploration samples used to generate T blocks. Thus, assuming that the nonstationary policy π improves with more data, we expect to obtain the *best-performing* progressive validation policy with the *most accurate* value estimate by running the algorithm DR-ns on all of the exploration data.

The error bound in the theorem below is proved by analyzing range and variance of $\hat{V}_k$ using Lemma 3.8. The theorem relies on the following conditions (mirroring the assumptions of Lemma 3.8):

- There is a constant $M > 0$ such that $\pi_t(a_k|x_k)/p_k \le M$.
- There is a constant $e_{\hat{r}} > 0$ such that $\mathbf{E}_{(x,a)\sim\pi_t} [\mathbf{E}_D[(\hat{r} - r)^2 \,|\, x, a]] \le e_{\hat{r}}$.
- There is a constant $v_r > 0$ such that $\mathbf{V}_{x\sim D} [\mathbf{E}_{r,a\sim\pi_t(\cdot,\cdot|x)}[r]] \le v_r$.

These conditions ensure boundedness of density ratios, squared prediction error of rewards, and variance of a conditional expected reward, respectively. It should be noted that, since rewards are assumed to be in [0, 1], one can always choose $e_{\hat{r}}$ and v_r that are no greater than 1.

THEOREM G.1. *Let N be the number of exploration samples used to generate T blocks, that is, $N = \sum_{t=1}^{T} |B(t)|$. Assume the above conditions hold for all k and t (and all histories z_{k-1} and h_{t-1}). Then, with probability at least $1 - \delta$,*

$$\begin{aligned} &|\hat{V}^{\mathrm{avg}}_{\mathrm{DR\text{-}ns}} - \mathop{\mathbf{E}}_{r\sim\pi_{\mathrm{PV}}}[r]| \\ &\quad \le \frac{N c_{\max}}{C} \\ &\qquad \cdot 2\max\left\{\frac{(1+M)\ln(2/\delta)}{N}, \right. \\ &\qquad\qquad \left.\sqrt{\frac{(v_r + M e_{\hat{r}})\ln(2/\delta)}{N}}\right\}. \end{aligned}$$

PROOF. The proof follows by Freedman's inequality (Theorem B.1 in Appendix B), applied to random variables $c_t\hat{V}_k$, whose range and variance can be bounded using Lemma 3.8 and the bound $c_t \le c_{\max}$. In applying Lemma 3.8, note that $\delta_\varrho = 0$ and $\varrho_{\max} = 1$, because $\hat{\mu}_k = \mu_k$. □

ACKNOWLEDGMENTS

We thank the editors and reviewers for their vast amount of patience and efforts that have improved the paper substantially.

REFERENCES

AGARWAL, D., CHEN, B.-C., ELANGO, P. and RAMAKRISHNAN, R. (2013). Content recommendation on web portals. *Comm. ACM* **56** 92–101.

ASUNCION, A. and NEWMAN, D. J. (2007). UCI machine learning repository. Available at http://www.ics.uci.edu/~mlearn/MLRepository.html.

AUER, P., CESA-BIANCHI, N., FREUND, Y. and SCHAPIRE, R. E. (2002/03). The nonstochastic multiarmed bandit problem. *SIAM J. Comput.* **32** 48–77 (electronic). MR1954855

BARTO, A. G. and ANANDAN, P. (1985). Pattern-recognizing stochastic learning automata. *IEEE Trans. Systems Man Cybernet.* **15** 360–375. MR0793841

BEYGELZIMER, A. and LANGFORD, J. (2009). The offset tree for learning with partial labels. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (*KDD*) 129–138. Association for Computing Machinery, New York.

BEYGELZIMER, A., LANGFORD, J. and RAVIKUMAR, P. (2008). Multiclass classification with filter-trees. Unpublished technical report. Available at http://arxiv.org/abs/0902.3176.

BEYGELZIMER, A., LANGFORD, J., LI, L., REYZIN, L. and SCHAPIRE, R. E. (2011). Contextual bandit algorithms with supervised learning guarantees. In *International Conference on Artificial Intelligence and Statistics* (*AI&Stats*) 19–26. jmlr.org.

Blum, A., Kalai, A. and Langford, J. (1999). Beating the hold-out: Bounds for K-fold and progressive cross-validation. In *Proceedings of the Twelfth Annual Conference on Computational Learning Theory* (*Santa Cruz*, *CA*, 1999) 203–208. ACM, New York. MR1811616

Bottou, L., Peters, J., Quiñonero-Candela, J., Charles, D. X., Chickering, D. M., Portugaly, E., Ray, D., Simard, P. and Snelson, E. (2013). Counterfactual reasoning and learning systems: The example of computational advertising. *J*. *Mach*. *Learn*. *Res*. **14** 3207–3260. MR3144461

Cassel, C. M., Särndal, C. E. and Wretman, J. H. (1976). Some results on generalized difference estimation and generalized regression estimation for finite populations. *Biometrika* **63** 615–620. MR0445666

Chan, D., Ge, R., Gershony, O., Hesterberg, T. and Lambert, D. (2010). Evaluating online ad campaigns in a pipeline: Causal models at scale. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (*KDD*) 7–16. ACM, New York.

Chapelle, O. and Li, L. (2012). An empirical evaluation of Thompson sampling. In *Advances in Neural Information Processing Systems* 24 (*NIPS*) 2249–2257. Curran Associates, Red Hook, NY.

Dudík, M., Langford, J. and Li, L. (2011). Doubly robust policy evaluation and learning. In *International Conference on Machine Learning* (*ICML*).

Dudík, M., Erhan, D., Langford, J. and Li, L. (2012). Sample-efficient nonstationary-policy evaluation for contextual bandits. In *Conference on Uncertainty in Artificial Intelligence* (*UAI*) 1097–1104. Association for Computing Machinery, New York.

Freedman, D. A. (1975). On tail probabilities for martingales. *Ann*. *Probab*. **3** 100–118. MR0380971

Gretton, A., Smola, A. J., Huang, J., Schmittfull, M., Borgwardt, K. and Schölkopf, B. (2008). Dataset shift in machine learning. In *Covariate Shift and Local Learning by Distribution Matching* (J. Quiñonero-Candela, M. Sugiyama, A. Schwaighofer and N. Lawrence, eds.) 131–160. MIT Press, Cambridge, MA.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P. and Witten, I. H. (2009). The WEKA data mining software: An update. *SIGKDD Explorations* **11** 10–18.

Hazan, E. and Kale, S. (2009). Better algorithms for benign bandits. In *Proceedings of the Twentieth Annual ACM-SIAM Symposium on Discrete Algorithms* 38–47. SIAM, Philadelphia, PA. MR2809303

Horvitz, D. G. and Thompson, D. J. (1952). A generalization of sampling without replacement from a finite universe. *J*. *Amer*. *Statist*. *Assoc*. **47** 663–685. MR0053460

Kakade, S., Kearns, M. and Langford, J. (2003). Exploration in metric state spaces. In *International Conference on Machine Learning* (*ICML*) 306–312. AAAI Press, Palo Alto, CA.

Kang, J. D. Y. and Schafer, J. L. (2007). Demystifying double robustness: A comparison of alternative strategies for estimating a population mean from incomplete data. *Statist*. *Sci*. **22** 523–539. MR2420458

Kearns, M. and Singh, S. (1998). Near-optimal reinforcement learning in polynomial time. In *International Conference on Machine Learning* (*ICML*) 260–268. Morgan Kaufmann, Burlington, MA.

Lambert, D. and Pregibon, D. (2007). More bang for their bucks: Assessing new features for online advertisers. In *International Workshop on Data Mining for Online Advertising and Internet Economy* (*ADKDD*) 100–107. Association for Computing Machinery, New York.

Langford, J., Strehl, A. L. and Wortman, J. (2008). Exploration scavenging. In *International Conference on Machine Learning* (*ICML*) 528–535. Association for Computing Machinery, New York, NY.

Langford, J. and Zhang, T. (2008). The Epoch–Greedy algorithm for contextual multi-armed bandits. In *Advances in Neural Information Processing Systems* (*NIPS*) 817–824. Curran Associates, Red Hook, NY.

Lewis, D. D., Yang, Y., Rose, T. G. and Li, F. (2004). RCV1: A new benchmark collection for text categorization research. *J*. *Mach*. *Learn*. *Res*. **5** 361–397.

Li, L., Chu, W., Langford, J. and Schapire, R. E. (2010). A contextual-bandit approach to personalized news article recommendation. In *International Conference on World Wide Web* (*WWW*) 661–670. Association for Computing Machinery, New York.

Li, L., Chu, W., Langford, J. and Wang, X. (2011). Unbiased offline evaluation of contextual-bandit-based news article recommendation algorithms. In *ACM International Conference on Web Search and Data Mining* (*WSDM*) 297–306. Association for Computing Machinery, New York.

Lunceford, J. K. and Davidian, M. (2004). Stratification and weighting via the propensity score in estimation of causal treatment effects: A comparative study. *Stat*. *Med*. **23** 2937–2960.

McAllester, D., Hazan, T. and Keshet, J. (2011). Direct loss minimization for structured prediction. In *Advances in Neural Information Processing Systems* (*NIPS*) 1594–1602. Curran Associates, Red Hook, NY.

Murphy, S. A. (2003). Optimal dynamic treatment regimes. *J*. *R*. *Stat*. *Soc*. *Ser*. *B Stat*. *Methodol*. **65** 331–366. MR1983752

Murphy, S. A., van der Laan, M. J. and Robins, J. M. (2001). Marginal mean models for dynamic regimes. *J*. *Amer*. *Statist*. *Assoc*. **96** 1410–1423. MR1946586

Orellana, L., Rotnitzky, A. and Robins, J. M. (2010). Dynamic regime marginal structural mean models for estimation of optimal dynamic treatment regimes. Part I: Main content. *Int*. *J*. *Biostat*. **6** Art. 8, 49. MR2602551

Precup, D., Sutton, R. S. and Singh, S. P. (2000). Eligibility traces for off-policy evaluation. In *International Conference on Machine Learning* (*ICML*) 759–766. Morgan Kaufmann, Burlington, MA.

Robbins, H. (1952). Some aspects of the sequential design of experiments. *Bull*. *Amer*. *Math*. *Soc*. (*N*.*S*.) **58** 527–535. MR0050246

Robins, J. (1986). A new approach to causal inference in mortality studies with a sustained exposure period—Application to control of the healthy worker survivor effect. Mathematical models in medicine: Diseases and epidemics. Part 2. *Math*. *Modelling* **7** 1393–1512. MR0877758

Robins, J. M. (1998). Marginal structural models. In 1997 *Proceedings of the American Statistical Association*, *Section on Bayesian Statistical Science* 1–10. Amer. Statist. Assoc., Alexandria, VA.

Robins, J. M. and Rotnitzky, A. (1995). Semiparametric efficiency in multivariate regression models with missing data. *J*. *Amer*. *Statist*. *Assoc*. **90** 122–129. MR1325119

ROBINS, J. M., ROTNITZKY, A. and ZHAO, L. P. (1994). Estimation of regression coefficients when some regressors are not always observed. *J. Amer. Statist. Assoc.* **89** 846–866. MR1294730

ROTNITZKY, A. and ROBINS, J. M. (1995). Semiparametric regression estimation in the presence of dependent censoring. *Biometrika* **82** 805–820. MR1380816

SHIMODAIRA, H. (2000). Improving predictive inference under covariate shift by weighting the log-likelihood function. *J. Statist. Plann. Inference* **90** 227–244. MR1795598

STREHL, A., LANGFORD, J., LI, L. and KAKADE, S. (2011). Learning from logged implicit exploration data. In *Advances in Neural Information Processing Systems* (*NIPS*) 2217–2225. Curran Associates, Red Hook, NY.

VANSTEELANDT, S., BEKAERT, M. and CLAESKENS, G. (2012). On model selection and model misspecification in causal inference. *Stat. Methods Med. Res.* **21** 7–30. MR2867536

ZHANG, B., TSIATIS, A. A., LABER, E. B. and DAVIDIAN, M. (2012). A robust method for estimating optimal treatment regimes. *Biometrics* **68** 1010–1018. MR3040007