ML Tools Project

Aidan Claffey, Amanda Kuznecov, Dan Turkel, Peter Simone

Spring 2021

1 Introduction

The multi-armed bandit setting entails modeling a decision-making process when the rewards are only partially observed. In particular, the value of the chosen action is revealed, but the potential values of the actions not chosen remain unknown.

Decisions in these settings are made using a *policy* which chooses an action for each trial. In the special case of *contextual* bandits, the policy has access to some vector of features called the *context* for each trial and uses this vector as input.

There are an infinite variety of potential policies, and testing a large set of them in production means potentially deploying poor-performing models. As a result, *offline policy evaluation* methods are used to estimate the value of a policy, which is the expected reward of choosing actions according to that policy.

In this paper, we focus on counterfactual evaluation methods which involve evaluating a *target policy* on feedback which was logged under the regime of a different *logging policy*. The difference between the actions in the logged data and the actions chosen by the target policy creates difficulties for policy evaluation, as we may be required to estimate rewards for action-context pairs which were not seen in the data, or else we must adjust for *covariate shift*, the shift in the data distribution. The estimator may suffer from bias or variance issues, posing a difficulty in evaluation.

Three high-level categories of policy estimation are explored in this paper. The first, *importance-weighted* (IW) value estimation, uses importance weighting between the target policy and logging policy to correct the proportions of action propensities in the logged data. The IW estimator is unbiased but can suffer from high variance, especially in instances where the propensity of an action under the logging policy is very low. The second category is *direct method* (DM) estimators, which use the logged data to build a model of the reward function, which is in turn used to predict rewards for the actions chosen by the target policy. DM estimators introduces some bias, but generally reduce variance in the estimate. The third category, *doubly robust* (DR) estimators combine the first two by leveraging both importance weighting and reward estimation. DR estimators are unbiased if the propensity model *or* the reward model performs well.

2 Prior Work

Propensity and importance-weighted methods have been formalized several times across the related disciplines of supervised learning, reinforcement learning, and bandit tasks in particular, but foundational papers include [1, 2].

One of the earliest proposals of the direct method for a bandit-like setting is [3]. A more general reference for regression with missing covariates is [4].

The conclusions of Dudík and Langford in [5], which formally integrated DR estimators into the setting of policy evaluation and learning, are of primary interest to our work.

Dudík and Langford perform experiments on several publicly available multiclass classification datasets from the popular UC Irvine machine learning repository[1] as well as proprietary data from Yahoo. The authors compare IW, DM, and DR estimators on these datasets for policy evaluation. They find that importance-weighted methods suffer from issues of variance and the direct method suffers from high bias, while the doubly robust method mitigates these effects and outperforms the two. In particular, they deem that the direct method is "significantly worse on all datasets" due to its high RMSE (compared to a ground truth policy value) and thus exclude it from their policy learning experiments.

Our report seeks to give the direct method a second chance. The performance of DM estimators relies on the fit of the reward estimation model, but only the results from a single linear ridge regression are reported in the paper. We reproduce some of the policy value estimation experiments from [5] with multiple DM models of varying levels of expressivity to provide a better baseline for comparison against DR. Furthermore, we carry out several additional experiments by analyzing the size of the public datasets and introducing a new dataset of logged bandit data which we believe provides a more realistic benchmark for the task.

3 Problem Definition and Approach

In the contextual bandit setting, we define $\mathcal{X}$ as the context space, $\mathcal{A}$ as the action space with k possible choices,

[1] https://archive.ics.uci.edu/ml/index.php

and $R(A)$ as the reward for a given action. Bandit feedback occurs over n total rounds, and in each round we are provided with history up to the current round i as $\mathcal{D}_i = ((X_1, A_1, R_1(A_1)), ...(X_{i-1}, A_{i-1}, R_{i-1}(A_{i-1}))$. With bandit feedback, we only ever observe the reward for the *selected* action.

A policy determines the probability of selecting each action, given a context: $\mathbb{P}(A_i = a \mid x_i) = \pi_0(a \mid x_i)$. When dealing with offline evaluation, the policy which created the logged data is called the logging policy, π_0. A stochastic k-armed bandit is described by the following steps:

1. A context and reward vector are sampled jointly, i.i.d., for each round $(x_i, R_i) \sim P$
2. For each round from $i = 1, ..., n$:
   (a) Our logging policy $\pi_0(a \mid x)$ selects action $A_i \in 1, ..., k$ according to $A_i \sim \pi_0(\cdot \mid x_i)$.
   (b) We observe reward $R_i(A_i)$.

The value of the static policy $\pi_0(x \mid a)$ is given by:

$$V(\pi_0) = \mathbb{E}[R(A)] \tag{1}$$

where $(X, R) \sim P$ and $(A \mid X) \sim \pi_0(\cdot \mid X)$. Our experiments focus on estimating the value of a static policy π that was *not* used to generate our logged data, and therefore, it is referred to as off-policy (or counterfactual) evaluation. This can be written as:

$$V(\pi) = \mathbb{E}_{(X,R)\sim P, A|X\sim\pi(\cdot|X)}[R(A)] \tag{2}$$

Here, we would like to estimate the expected rewards for the target policy π, by using the logged data generated from the logging policy π_0. Off-policy estimation methods must contend with the covariate shift between the distributions of the two policies (and their resulting actions): the action distribution we logged might not look like the actions we would take with the new policy.

Since we are adapting a multi-class classification problem to a contextual bandit problem, some manipulation of the data is required, which we discuss in a later section. The following subsections provide background for the off-policy estimators used in our analysis.

3.1 Importance-Weighted Estimator

The importance-weighted (IW) value estimator for policy π based on bandit feedback $\mathcal{D}_i$ with actions chosen under a static logging policy π_0 is defined as:

$$\hat{V}_{iw}(\pi) = \frac{1}{n}\sum_{i=1}^{n} R_i(A_i)\frac{\pi(A_i|x_i)}{\pi_0(A_i|x_i)} \tag{3}$$

With each observed reward $R_i(A_i)$, the sample is inverse-weighted by the action's propensity under the logging policy and re-weighted by the action's propensity under the target policy, given the context. We call this ratio of propensities the importance weight:

$$W_i = \frac{\pi(A_i|x_i)}{\pi_0(A_i|x_i)} \tag{4}$$

In our experiments, this estimator is referred to by the shorthand *iw*.

3.2 Direct Method Estimator

The direct method (DM) is defined as:

$$\hat{V}_{dm}(\pi) = \frac{1}{n}\sum_{i=1}^{n}\sum_{a=1}^{k}\hat{r}(x_i, a)\pi(a \mid x_i) \tag{5}$$

where $\hat{r}(x_i, A_i)$ represents the estimated reward function. Unlike the importance-weighted method, in the direct method each round's reward is estimated as the weighted average of a regression of the reward function. Given that our action spaces for each dataset are relatively small, this formulation is suitable to our approaches.

The most basic approach is to model $\hat{r}$ as a regression. In [5], the authors used ridge regression to formulate an estimate of the rewards given the context, with one regression for each of the k actions. We have reproduced this estimator using ridge regression and refer to it as *dm* in our experiments.

We can expand on and adapt this estimator in several ways. First, we can train this estimator using importance sampling, where each sample in the data holds a weight in the regression based off the propensities of the logging and target policies for a given action. This is referred to as *dm_iw* in our experiments.

Secondly, following the data setup of [5] for the classification datasets, we model all rewards as either 0 or 1 for any given (x_i, A_i). This reward vector is well-suited for interpretation as probability estimates, so we departed from the paper by also training a logistic regression to predict the rewards. We refer to this estimator as *dm_log*. We can also apply inverse propensity weighting to this fit, which we refer to as *dm_log_iw*.

Finally, we can use more expressive models than linear regression such as a non-linear method to predict the rewards. Our last form of direct method uses random forests, which we refer to as *dm_rf* in our experiments.

3.3 Doubly Robust Estimator

The doubly robust (DR) estimator is specified in [5]. This method takes advantage of both the IW and DM methods outlined above. For each sample in the data, we use our reward estimate for the given action as a baseline in the IW estimate. Then we apply a correction according to the DM estimate. Similar to the DM approach, we can apply several different methods of reward estimation. The DR

Dataset	Samples	Features	Classes
ecoli	336	7	8
glass	214	9	6
letter-recognition	20,000	16	6
optdigits	5,620	64	10
yeast	1,484	8	10
pendigits	10,992	16	10

Table 1: Datasets from the UCI Machine Learning repository.

estimator is:

$$\hat{V}_{dr}(\pi) = \frac{1}{n}\sum_{i=1}^{n}\left[\left[\sum_{a=1}^{k}\hat{r}(x_i,a)\pi(a \mid x_i)\right] + \frac{\pi(A_i|x_i)}{\pi_0(A_i|x_i)}(R_i(A_i) - \hat{r}(x_i,A_i))\right] \tag{6}$$

In our experiments, we refer to the DR method with a ridge reward estimator as *dr*, DR method using DM with logistic reward estimator as *dr_log*, and DR method with a random forest reward estimator as *dr_rf*.

4 Experiments

We conducted experiments based on those in [5], as well as several variants an expansions in the following sections.

4.1 Multiclass Classification with Bandit Feedback

Following [5], we used public datasets from the UCI Machine Learning Repository to simulate a bandit setting: "ecoli," "glass," "letter-recognition," "optdigits," "yeast," and "pendigits." These small datasets are for multiclass classification tasks, and their contents are summarized in Table 1.

4.1.1 Data Setup

Since the datasets are originally for a supervised classification task, we followed the procedure in [5] for transforming them into bandit datasets. The features for each data point represent a context vector, and a reward of 1 is observed if the "action" corresponding to the correct class is chosen, else a reward of 0 is observed.[2]

The set of contexts X and reward vectors y (which are 1-hot with a 1 only in the position of the correct class) constitute a "fully observed" bandit dataset which we will later use to simulate our logging policy.

[2]The authors of [5] formulate the problem in terms of loss, but we have adapted the terminology to *rewards* to align with more recent value estimation literature.

Next, each dataset was randomly split into roughly equal sizes. However, since some of the datasets are fairly small, we ensured that all actions were in the training set by first selecting one instance of each action, and then randomly selecting from the remaining samples to make up the training set.

4.1.2 Experiment Procedure

Following the train/test split of the the fully-observed data, the procedure is as follows:

1. Train a multinomial logistic regression on the training set to serve as our *target policy*. The model's class probabilities for a given context vector constitute our $\pi(x)$.[3]

2. Calculate a ground truth value for the target policy using the test data. Rather than sampling from the policy, we simply take the dot product of the (one-hot) reward vector with the class probabilities for each context.

3. Perform 500 rounds of the following:

   (a) Simulate our logging policy on the fully observed test data. We sample one action-reward pair for each data point to generate a batch of logged feedback.

   (b) Use the estimation methods outlined in Section 3 to get a value estimate.

4. Calculate statistics on the value estimates for each estimator.

In our experiments, we attempted to provide a larger span of well-tuned regression models for the direct method. Our ridge regression uses scikit-learn's *RidgeCV* class, which performs cross-validation across a set of regularization weights (1, 0.1, 0.01, 0.001) to find the best performer. For logistic regression, doing cross-validation for each round unfortunately proved too computationally expensive. Lastly, our random forest regressor optimized for entropy with 100 estimators and a minimum leaf size of 5.

In an attempt to push the performance of the direct method further, we also included ridge and logistic regressions with importance-sampling to weight each data point.

4.1.3 Metrics

For each estimator, we calculate the mean value estimate across the batches, but we also calculate three other statistics: bias, variance, and RMSE.

The bias is the absolute difference between the mean estimate and the ground truth value. The variance is simply the variance of the value estimates. The RMSE is

[3]While [5] is not entirely clear on this point, we interpret the policy to be probabilistic (and have formulated Equation 6 to reflect that), meaning that it chooses actions using the class probabilities, rather than simply always choosing the maximally probable class.

the root mean squared error of the estimates compared to the ground truth.

The reason we tracked these metrics is because none tells the full story on their own. Importance weighting should have almost no bias (it is an unbiased estimator after all), but can have high variance, while direct methods often have lower variance but introduce bias in their reward estimation. The RMSE is the square root of the sum of the variance and squared bias, so we can think of it as a combination of the two. However, depending on the application, one might choose an estimator that does not have the lowest RMSE if it is more important, for example, to have minimal variance.

4.1.4 Results

Following [5], we used a uniform logging policy in our first experiment. The results are summarized in Figure 1.

As expected, the IW estimator is unbiased as we know the true propensities under the logging policy: $1/k$ for all actions. Similarly, the importance-sampled regression estimators (*dm_iw* and *dm_log_iw*) are largely unbiased compared to other estimators. The DR estimators are also largely unbiased, which we expect thanks to the inclusion of importance weighting, with the exception of *dr_rf*. This is slightly surprising as random forests are quite expressive, but it is likely that some datasets simply did not have enough data for the random forest to fit them well. The direct methods *dm*, *dm_log* and *dm_rf* incur a large amount of bias which is attributed to the inability of each of these model types (linear regression, logistic regression and random forest) to capture the correct action.

The *iw* estimator had the largest variance across most datasets. As expected, the DM methods show the least amount of variance, with the exception of *dm_rf*, which is likely due to the small dataset size. It is also worth noting that the "ecoli" and "glass" datasets, which are at minimum four times smaller than the other datasets, have much larger variance. This suggests that with larger datasets, estimator variance is minimal and roughly equal across all estimators, however the *iw* estimator remains the largest source of variance.

Evaluating estimator performance with RMSE shows that the *dm_log* and *dm_rf* estimators were the worst across all datasets. It appears that the IW method coupled with the DM method produced the best results; either *dm_iw* or *dm_log_iw* had the lowest RMSE per dataset. DR methods *dr* and *dr_log* also exhibited good performance as these estimators also combine IW and DM methods. Again it is notable that with larger datasets, the RMSE values drop off significantly.

One consideration we had about the experimental design from [5] was that importance weighting had an "advantage" because not only was the logging policy known, it was very simple. As a follow-up experiment, we designed a logging policy which generated a random $j \times k$ weight matrix W (where j is the length of the context vector and k is the number of actions) and selected actions with probabilities from the normalized vector $x_i W$. Thus not only is the policy nonuniform, but it is context-dependent.

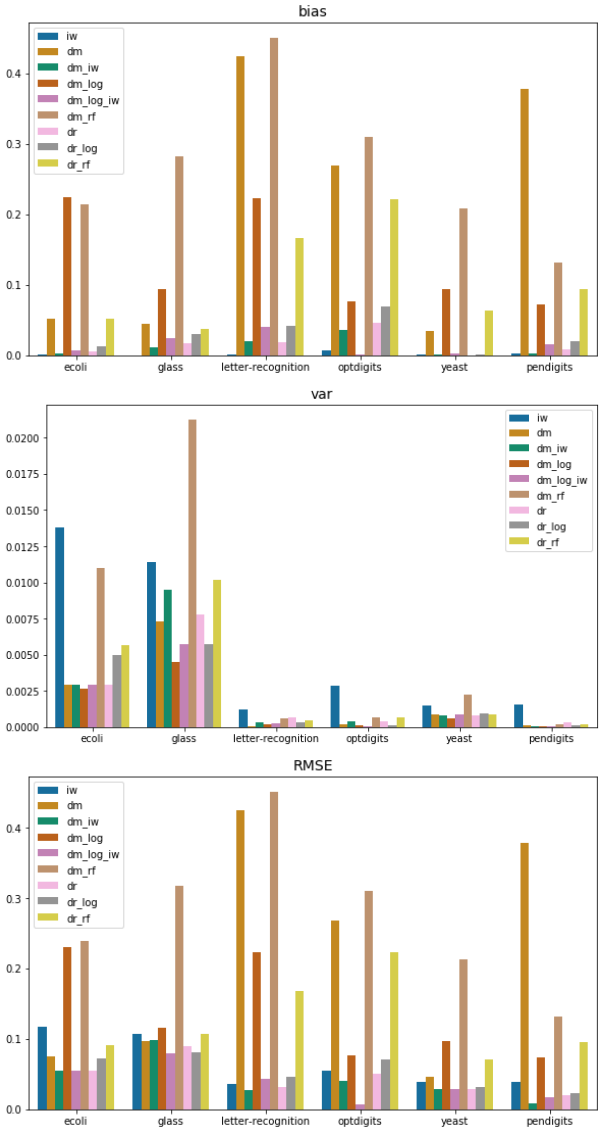

Figure 1: Bias (top), variance (middle) and RMSE (bottom) of all estimators for off-policy value for a uniform logging policy.

To add another degree of complexity, rather than provide the estimators with the true propensities for each action given the context, we instead modeled the logging policy propensities by fitting a logistic regression to the logged feedback. This simulates the challenging scenario of learning off-policy from an unknown logging policy.

We then tested the same value estimators as in our first experiment.[4] To our surprise, the results (which are

[4]We note that this change does not impact the results of the *dm*,

summarized in Figure 6 in the Appendix) were largely unchanged.

Forced to model a more difficult logging policy, the *iw* policy gained some bias across most datasets, though the importance-sampled direct method estimators did not suffer any additional bias. Additionally, the doubly robust estimators maintained their relatively low bias, lending credibility to their robustness.

The most surprising result came in variance, where *iw* had its variance reduced on several datasets, especially the small "ecoli" and "glass" datasets. We would only expect reduced variance if the new logging policy was more similar to the target policy than the uniform policy had been, but this seems unlikely since it is randomized in each batch, so this finding warrants further investigation.

In terms of RMSE, the added difficulty of modeling the propensities did not hamper the performance of *iw* or the DR methods. Indeed, the DR estimators were robust to the complex propensities and the importance weighted estimator seemed to model the propensities extremely well.

Nonetheless, these datasets are ultimately not from true bandit settings, which means they may not be a perfect setting for evaluating bandit value estimators, which motivates our next experiment.

4.2 Estimating Average Number of Clicks

The other experiment in [5] was to test different value estimators on real-world website traffic data that captures user visits to a "popular Internet portal" whose data is not publicly available. Their dataset contains 4 million user records that each have a sparse binary context vector x_i of length 5000 and a value v_i that represents the number of visits by the user to the website. Their experiment looked to measure the efficacy of DR and IW to estimate the average number of visits to a website from biased samples of the logged data with known probabilities. (DM was omitted due to its poor performance in the first experiment.) Their main results imply that the DR estimator consistently outperforms IW across different subsampling rates (from 0.0001 to 0.05).

While we cannot directly reproduce these results since the dataset is not available, we performed a similar experiment with a publicly available, large logged bandit dataset: the Open Bandit Dataset (OBD) from ZOZOTOWN [6], one of Japan's largest fashion e-commerce websites.

4.2.1 Data Setup

The OBD contains 6 separate logged datasets that cover three campaigns: *men*, *women*, and *all* using two different logging policies, *uniform random* and *Bernoulli Thompson sampling* (BTS). The BTS policy was pre-trained for a month before deployment. Table 2 shows some statistics on each set.

Policy	Campaign	# Samples	# Items
Uniform	Women	864,585	46
	Men	452,949	34
	All	1,374,327	80
BTS	Women	7,765,497	46
	Men	4,077,727	34
	All	12,168,084	80

Table 2: Campaigns in the Open Bandit Dataset. The context vector for each sample contains 26 features.

Each sample is a tuple of the form (x_i, a_i, r_i, p_i), where x_i is the context vector, a_i is the index of the item shown to the user, r_i is the binary reward denoting whether the user clicked on the given item, and p_i is the propensity of the item that was shown.

The clicks are very sparse. The average click-rate for each campaign ranged from 0.35% for *all – uniform random* to 0.67% for *men – bts*. For each campaign, the *bts* policy outperformed the *random uniform* policy at a statistically significant level.

4.2.2 Experiment Procedure

To evaluate the performance of each estimator on the logged data, we took an approach similar to [6]: using data from each campaign (*women*, *men*, and *all*) and each policy (*uniform random* and *bts*), we estimate the average number of clicks for the *uniform random* policy. One set of experiments estimates the performance of a policy using the same logged policy (which we will call *uniform* $\rightarrow$ *uniform*) and the other set will estimate the performance of a policy using a different logged policy (which we will call *bts* $\rightarrow$ *uniform*). We are also interested in the effect of data size on performance across estimators, as [5] examines in their web-traffic experiment.

For a given campaign and logging policy, the experiment proceeds as follows:

1. Compute an estimate of the value of the target (*uniform*) policy by taking the mean across all rewards for the *uniform* dataset of the given campaign.
2. For each subsample rate $\rho \in \{0.01, 0.05, 0.1\}$, do 50 trials of the following:
   (a) Take a bootstrapped (with replacement) sample of the campaign's dataset using subsample rate ρ.
   (b) Calculate the off-policy value using each estimator described in Section 3.
3. Calculate sample mean, bias, variance, and RMSE for each estimator for each campaign and logging policy.

dm_log, or *dm_rf* methods, since they do not utilize the logging propensities.

4.2.3 Results

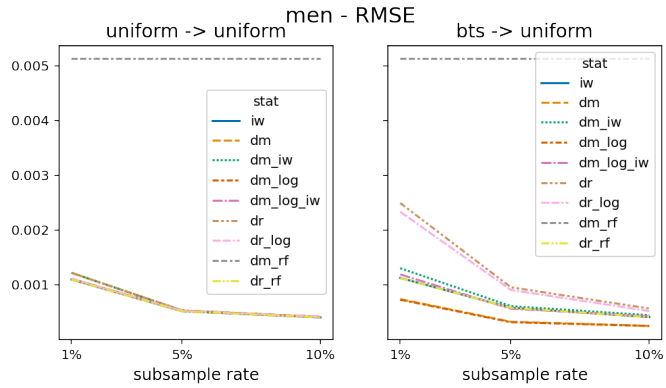

Figure 2: RMSE of the policy value estimate for different subsample rates and estimates for the *men* campaign

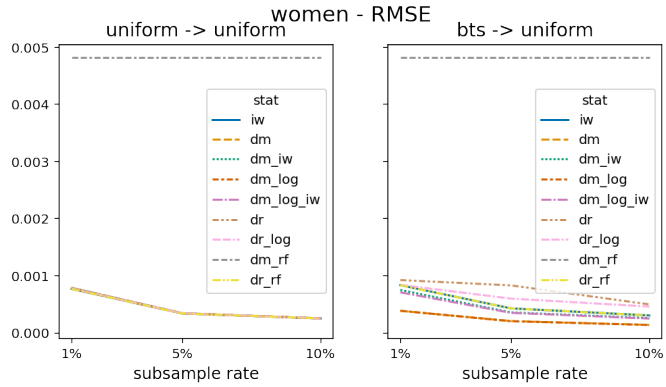

Figure 3: RMSE of the policy value estimate for different subsample rates and estimates for the *women* campaign

Figure 2 and Figure 3 display the RMSE for different estimators for the *men* campaign and s *women* campaign, respectively. Both demonstrate that *dm_rf* did not work at all (this method also performed poorly in our first experiment). With the *men* campaign, more data seemed to help lower the RMSE more so than in the *women* campaign, although both showed moderate improvement with increased sampling size.

When comparing the two logging policy performances, it makes sense that the *uniform* policy had a much lower spread among the estimators when compared to *bts*, since the logging policy is the exact same as the target policy.

Surprisingly, and against the results of [5], DR methods using the linear and logistic models (*dr* and *dm_log*) had the worst performance in terms of RMSE in both *men* and *women* campaigns for the *bts* → *uniform* experiment.

In general, it was hard to determine any concrete results from the Open Bandit Dataset experiments. This is in large part due to the nature of the logged data. Even with hundreds of thousands or millions of rows, when only 0.5% of the logs result in a click, and with only 26 binary features, it is very hard to learn a pattern. However, the estimators *do* work—the *bts* logging policy for each of the *men* and *women* campaigns give an average click-rate at around 0.65%, but estimating the click rate for the target *uniform* policy, all estimators (besides random forest) correctly settled on a value close to the logged *uniform* policy value of around 0.50%. Figures 4 and 5 demonstrate this result.

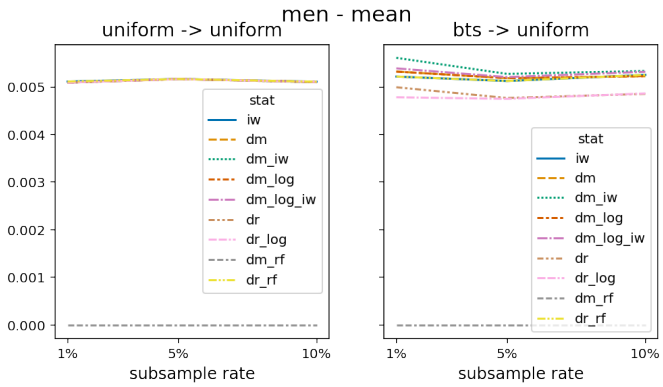

Figure 4: Mean policy value estimate for different subsample rates and estimators for the *men* campaign

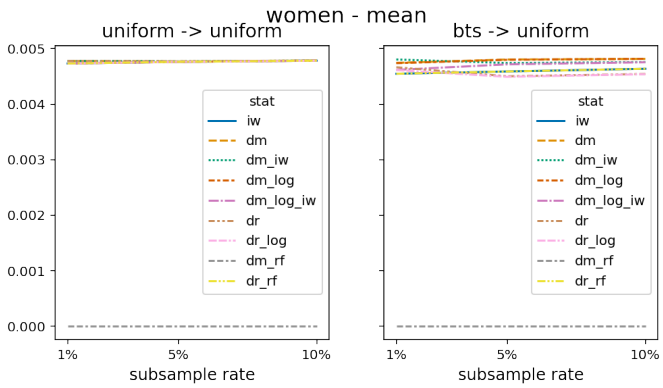

Figure 5: Mean policy value estimate for different subsample rates and estimators for the *women* campaign

5 Conclusion

We revisited the experiments in [5] because we felt that the direct method was short-changed. The conditions of the experiment seemed to lean against the direct method: the logging policy was uniform and known, so that importance weighting was at lower risk of suffering from high variance, and the direct method featured only a single linear model for reward imputation.

In our experiments on the same datasets, we made several modifications in an attempt to level the playing field. We added several variants of the direct method

using different regression models which we felt were better tuned, and we tested importance-sampled regressors, which also leverage propensities, but in a simpler way than the doubly robust method. Furthermore, we expanded the experiment to include off-policy estimation with a more complex logging policy and required estimation of the propensities. Additionally, we included a new bandit dataset to get a more realistic appraisal of estimator performance.

We believe our results constitute a modest vindication for the direct method. Performance could at times vary dramatically from one dataset to another (which should emphasize the importance of testing multiple estimators whenever approaching a new dataset), but on the smaller datasets ("ecoli" and "glass"), the direct method was fairly competitive with the doubly robust variants, even without the addition of importance sampling. In those same datasets, it was importance weighting which often lagged in performance.

Model specification clearly proved to be quite important to direct method performance. We were surprised that [5] chose to model rewards with a linear rather than logistic regression, given that the rewards themselves were binary. Indeed, on "optdigits" and "pendigits," the *dm_log* had substantially less bias than vanilla *dm*, yet on "ecoli", the effect is reversed. This leads to the unsurprising conclusion that an estimator which relies on a regression of the rewards requires a well-specified regression model.

While *iw* did have higher variance than the direct method estimators, ultimately the bias was more varied across the estimators and contributed much more to the overall performance. Since the direct method inherits bias from its reward model, it loses out here.

We thought that changing the underlying logging distribution would create a bigger challenge for the IW and DR estimators, but they ultimately overcame it. It may be that we need to conduct experiments with more complex logging policies, though in real-world scenarios logging policies are often deliberately simple in order to make off-policy estimation easier.

On the much larger OpenBanditDataset, we could not make strong conclusions about the performance of any specific estimator, besides a simple random forest model that could not generalize due to the high class imbalance. Naturally, we found that as the subsample size increased, the variance decreased on all estimators, but especially for the doubly robust methods. The results do show that even with millions of data points, finding patterns for what is essentially anomaly detection, when only 0.5% of visits to the site end in a click, when there are anywhere from 34 to 80 possible actions to choose from, is quite difficult. In [5], their dataset also contained millions of web-traffic entries, but each context vector was a sparse binary vector of size 5000, and each data point contained the *number* of visits across an entire month to a website, rather than the reward given by each visit. Our assumption is that with a larger, more expressive dataset, the Direct Method can work well given a model of appropriate size.

Ultimately our results provide a more nuanced picture of the comparative performance of the estimators. The direct method is undoubtedly more biased than importance weighting or doubly robust methods, but with the right regression model, the story is not nearly as drastic as the findings in [5] might have you believe. Furthermore, the doubly robust is only so robust: poorly specified reward models still harm performance.

We should note that the unsung hero of our experiments was the importance-sampled direct method, which is a direct method at heart but also relies upon propensities (either known or estimated) during training. When the regression was well-specified, these models frequently outperformed the rest across the datasets. However, importance weighting cannot save ill-specified reward models; even with importance weights, the direct method and doubly robust estimators that use a random forest for the "optdigits" dataset had very low performance compared to other reward imputation models.

Our principal finding is thus that the doubly robust method is not guaranteed to be the best estimator method for all datasets, and the direct method is certainly not universally useless. Even those who are inclined toward the doubly robust estimator for its theoretical guarantees should be sure to experiment with their reward estimation model to ensure they are not dragging down their performance with lousy imputations.

References

[1] Doina Precup, Richard S. Sutton, and Satinder Singh. "Eligibility Traces for Off-Policy Policy Evaluation". In: *Proceedings of the Seventeenth International Conference on Machine Learning*. Morgan Kaufmann, 2000, pp. 759–766.

[2] Alexander L. Strehl et al. "Learning from Logged Implicit Exploration Data". In: *Proceedings of the 23rd International Conference on Neural Information Processing Systems - Volume 2*. Vancouver, British Columbia, Canada, 2010, pp. 2217–2225.

[3] Sang-June Park and Minhi Hahn. "Direct Estimation of Batsell and Polking's Model". In: *Marketing Science* 17.2 (1998), pp. 170–178.

[4] James M. Robins, Andrea Rotnitzky, and Lue Ping Zhao. "Estimation of Regression Coefficients When Some Regressors Are Not Always Observed". In: *Journal of the American Statistical Association* 89.427 (1994), pp. 846–866.

[5] Miroslav Dudík, John Langford, and Lihong Li. "Doubly Robust Policy Evaluation and Learning". In: *Proceedings of the 28th International Conference on International Conference on Machine Learning*. Bellevue, Washington, USA, June 2011, pp. 1097–1104.

[6] Yuta Saito et al. "Open Bandit Dataset and Pipeline: Towards Realistic and Reproducible Off-Policy Evaluation". In: *arXiv preprint arXiv:2008.07146* (2020).

Appendix

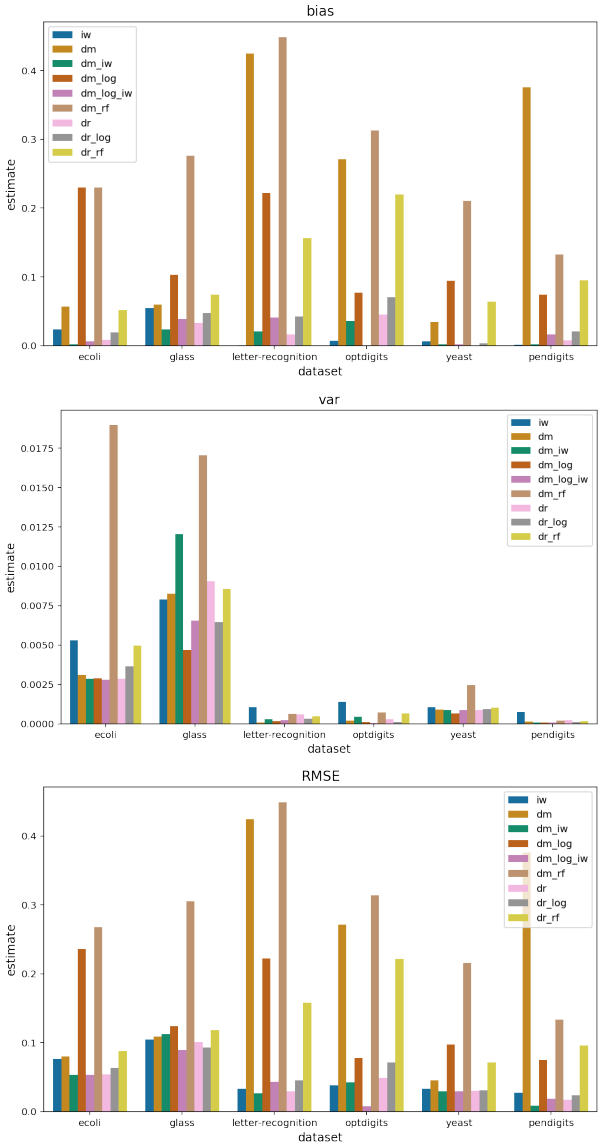

Figure 6: Bias (top), variance (middle) and RMSE (bottom) of all estimators for off-policy value on a randomized contextual logging policy.