Double Clipping: Less-Biased Variance Reduction in Off-Policy Evaluation

JAN MALTE LICHTENBERG*, ALEXANDER BUCHHOLZ, GIUSEPPE DI BENEDETTO, MATTEO RUFFINI, and BEN LONDON, Amazon Music, USA

"Clipping" (a.k.a. importance weight truncation) is a widely used variance-reduction technique for counterfactual off-policy estimators. Like other variance-reduction techniques, clipping reduces variance at the cost of increased bias. However, unlike other techniques, the bias introduced by clipping is always a downward bias (assuming non-negative rewards), yielding a lower bound on the true expected reward. In this work we propose a simple extension, called *double clipping*, which aims to compensate this downward bias and thus reduce the overall bias, while maintaining the variance reduction properties of the original estimator.

Additional Key Words and Phrases: off-policy evaluation, OPE, inverse propensity scoring, IPS, clipping

1 INTRODUCTION

Off-policy evaluators are a crucial component in the development of many real-world recommender systems. They allow us to estimate the performance of a new *target* recommendation policy based on interaction data logged from a different *logging* policy (for instance, the current production recommender), thereby reducing the need to run slow and costly A/B tests.

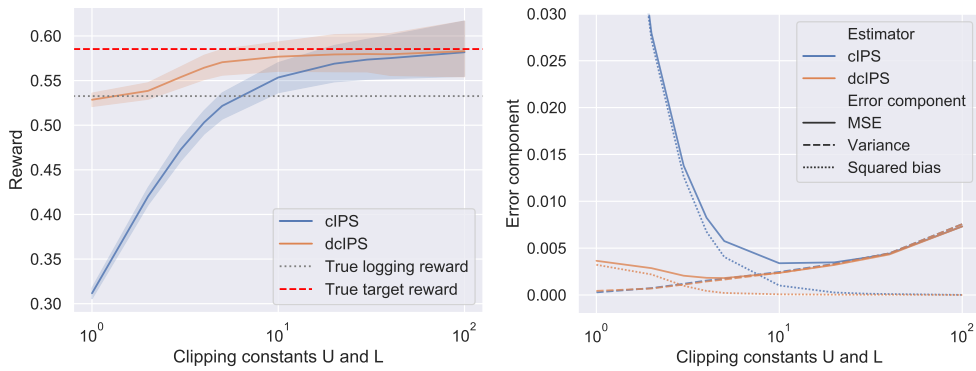
Many counterfactual off-policy estimators are based on the inverse propensity scoring (IPS) principle [2, 6, 7, 12]. Given a stochastic logging policy and some mild assumptions, IPS-based estimators are unbiased, but often suffer from high variance. This is true even on industrial-scale data sizes; in particular, if the logging policy is close to being deterministic. Intuitively speaking, most IPS estimators contain propensity ratio weights of the form $w = p_{\text{target}}/p_{\text{logging}}$, where p_{target} is a target propensity (e.g., the probability that the target policy recommends a particular action to the user) and p_{logging} is the logging propensity (e.g., the probability that the logging policy recommended that same action to the user). These ratios can become arbitrarily large for small logging propensities, which then leads to high variance in the overall estimate.

The literature has proposed various variance-reduction techniques for IPS-style estimators, including weight clipping [1, 2, 7, 12], self-normalization [17], doubly-robust estimators [4, 11, 13], as well as generalizations of those ideas [3, 15, 20]. In this article we revisit weight clipping, which is still used extensively due to its simplicity (it does not require a reward model) and its generality (it is readily applicable to IPS-style estimators used in more complex real-world applications, such as ranking [3, 9] or slate recommendation [18], where self-normalized or doubly-robust estimators are not available or difficult to implement).

The basic idea of weight clipping is to simply avoid large propensity weight ratios by (hard-)clipping the ratios by a constant upper bound U, which is usually treated as a hyper-parameter for the estimation procedure. Just like other variance-reduction techniques, the clipping procedure effectively reduces the variance of the IPS estimator at the cost of introducing a bias. Unlike other techniques, however, the bias introduced by clipping is always pessimistic. In other words, on average, the estimator underestimates the true expected reward (under the technical assumption that rewards are always non-negative), as illustrated in Figure 1a.

*Corresponding author, jlichten@amazon.com

Presented at CONSEQUENCES '23 workshop at RecSys 2023 conference, Singapore.



(a) (b)

Fig. 1. Comparison of clipped IPS (cIPS, blue) and doubly-clipped IPS (dcIPS, orange) in a synthetic bandit experiment (detailed setup described in Section 5). Figure 1a: Mean (solid line) and corresponding standard error bands of reward estimates across 100 repetitions as a function of clipping constants U (for both cIPS and dcIPS) and $L = U$ (only for dcIPS). The dashed red line shows the true reward of the target policy, i.e., the estimation target. The dotted grey line shows the average logging reward observed in the data set. Figure 1b: mean squared error (MSE) between reward estimate and true target reward across 100 repetitions. The dashed lines show the variance components, the dotted lines show the squared-bias components for both estimators.

In this work, we exploit this property of the clipping bias, so as to obtain more accurate estimates. Specifically, we clip the propensity ratios from both sides rather than just from above, thereby potentially correcting pessimistic underestimates with optimistic overestimates. Experiments with synthetic data show that this approach leads to a reduction in MSE.

2 BACKGROUND AND RELATED WORK

We focus on off-policy estimation in the standard contextual multi-armed bandit setting, but we note that our work is applicable to counterfactual learning-to-rank [8, 9] and slate recommendation [18].

2.1 Off-policy evaluation in the contextual bandit.

Consider a contextual bandit setting, where a stochastic *logging policy* $\pi_0(y|x)$ (e.g., the currently deployed recommender system) repeatedly selects an *action* $y \in \mathcal{Y}$ based on a given *context* $x \sim P(\mathcal{X})$ (e.g., user history, action features, etc.). The system then observes a non-negative *reward* $r \sim R(x, y) \geq 0$, which depends on the action that was selected and the context. The system does not observe the rewards for any action that was not selected by the logging policy. After n rounds, the logged data set is given by $\mathcal{D} = \{(x_i, y_i, r_i, \pi_0(\cdot|x_i))\}$, where r_i is the observed reward and $\pi_0(y_i|x_i)$ is the *propensity* (i.e., probability) of action y_i to be selected by the logging policy for context x_i. The goal of off-policy evaluation is to estimate the expected reward of a new *target* policy π, given by $R(\pi) = \mathbb{E}_{x\sim P(\mathcal{X})}\mathbb{E}_{y\sim\pi(\cdot|x)}\mathbb{E}_{r\sim R(x,y|x)}[r]$, based on the logging data set $\mathcal{D}$. The challenge is that the logged data only contains rewards for actions selected by π_0, which may be different from those selected by π. Thus, we are faced with a counterfactual estimation problem.

2.2 Counterfactual off-policy estimators.

The standard inverse propensity scoring (IPS) [2, 5, 12, 16] estimate for the contextual bandit problem is given by

$$\hat{R}_{\text{IPS}}(\pi) = \frac{1}{n}\sum_{i=1}^{n} r(x_i, y_i)\frac{\pi(y_i|x_i)}{\pi_0(y_i|x_i)} = \frac{1}{n}\sum_{i=1}^{n} r(x_i, y_i)w(x_i, y_i). \tag{1}$$

The estimator $\hat{R}_{\text{IPS}}(\pi)$ is an unbiased estimator of $R(\pi)$, given the overlap assumption: $\pi_0(y|x) > 0$ whenever $\pi(y|x) > 0$. To satisfy the overlap assumption, the logging policy is usually randomized, leading to the following dilemma. Too much randomization can degrade user experience, but too little randomization leads to high variance in IPS off-policy estimation: little randomization means that some propensity values $\pi_0(y_i|x_i)$ are tiny, which in turn leads to the occasional huge weighting factor $w(x_i, y_i)$.

A widely used technique to reduce the variance of the standard IPS estimator is to simply clip (some authors also say "truncate" or "trim") large importance weight ratios. Specifically, we use the clipped IPS estimator (cIPS) that clips the entire ratio, that is,

$$\hat{R}_{\text{cIPS}}(\pi, U) = \frac{1}{n}\sum_{i=1}^{n} r(x_i, y_i)\min\{w(x_i, y_i), U\}, \tag{2}$$

where $U \geq 1$ is the *upper clipping constant.*

3 CLIPPED IPS IS ALWAYS DOWNWARD BIASED.

Clearly, for $U = \infty$, the clipped IPS estimator (Eq. 2) is equivalent to the un-clipped IPS estimator from Eq. 1. With non-negative rewards, decreasing U effectively reduces variance, at the cost of a downward bias, as illustrated in Figure 1a. The following proposition confirms this intuition about the downward bias.

Proposition 3.1. *Let $w(x, y) > 0\ \forall x, y$, then the bias of $\hat{R}_{cIPS}(\pi, U)$ is given by (proof in the Appendix)*

$$Bias(\hat{R}_{cIPS}(\pi, U)) = \mathbb{E}_x\mathbb{E}_{y\sim\pi}\Bigg[\underbrace{\mathbf{1}_{\{w(x,y)>U\}}}_{\textit{Only clipped records}}\underbrace{\left(\frac{U}{w(x,y)} - 1\right)}_{\textit{Always < 0}}\underbrace{\mathbb{E}_r[r(x,y)|x,y]}_{\textit{Expected reward}}\Bigg]. \tag{3}$$

If the clipping constant U is higher than the highest attainable propensity weight ratio $w(x, y)$ across all requests, then the clipped IPS estimator essentially becomes the standard, unbiased IPS estimator. As soon as the clipping constant becomes "active" in the sense that it starts clipping propensity weight ratios, then the bias is always strictly negative assuming non-negative rewards (ignoring the trivial case in which all clipped requests have zero expected reward).

Many machine learning practitioners are happy to accept a small bias to reduce the variance of their estimators. Ideally, one would like to remove the bias from the variance reduction. However, this is difficult because often neither sign nor magnitude of the bias can be inferred from the variance reduction method. In the case of the cIPS estimator, however, Proposition 3.1 showed that the bias introduced is always negative (assuming non-negative rewards). This begs the question whether we can exploit this property to find a less bias-inducing variance-reduction method for off-policy estimation. In the following section we introduce a somewhat naïve, yet effective, method to do so.

4 TWO-SIDED DOUBLE CLIPPING

We define the *two-sided double-clipping IPS* (dcIPS) estimator as

$$\hat{R}_{\text{dcIPS}}(\pi, U, L) = \frac{1}{n}\sum_{i=1}^{n} r(x_i, y_i)\max\left\{\min\{w(x_i, y_i), U\}, \frac{1}{L}\right\}, \tag{4}$$

where $U \geq 1$ is the upper clipping constant and $L \geq 1$ is the lower clipping constant. The dcIPS subsumes the cIPS estimator; both estimators are equivalent for $L \to \infty$. On the other extreme, for both clipping constants approaching 1, the dcIPS estimator converges to the mean of rewards logged in the data set:

$$\mathbb{E}\left[\hat{R}_{\text{dcIPS}}(\pi, U, L)\right] \to R_{\text{logging}} \text{ for } U, L \to 1. \tag{5}$$

This is illustrated in Figure 1a, where the dcIPS (orange line) converges to the true logging reward (gray dotted line). This allows the intuitive interpretation of dcIPS as an estimator that regularizes towards the mean of the logging policy reward and the prior variance is determined by both clipping constants U and L. Under this regularization perspective, it makes sense to shrink the weights towards a positive constant (1 in this case) rather than to 0, because all weights are known to be positive [10].

Proposition 4.1. *Let $w(x, y) > 0\ \forall x, y$, then the bias of the dcIPS estimator with clipping constants U and L is given by*

$$Bias(\hat{R}_{dcIPS}(\pi, U, L)) = \mathbb{E}_x \mathbb{E}_{y\sim\pi}\left[\left(\underbrace{\mathbf{1}_{\{w(x,y)>U\}}\left(\frac{U}{w(x,y)} - 1\right)}_{\text{Always } \leq 0\text{, only depends on } U} + \underbrace{\mathbf{1}_{\{w(x,y)L<1\}}\left(\frac{1}{w(x,y)L} - 1\right)}_{\text{Always } \geq 0\text{, only depends on } L}\right)\underbrace{\mathbb{E}_r[r(x,y)|x,y]}_{\text{Expected reward}}\right]. \tag{6}$$

Equation 6 shows that the two clipping constants contribute separately, and in opposing directions, to the overall bias of the dcIPS estimator. In other words, we can try to tune the lower clipping constant L so as to compensate the bias introduced by the upper clipping constant U.

5 OFF-POLICY EVALUATION EXPERIMENTS

The synthetic experiments demonstrate that dcIPS is able to compensate the bias introduced by cIPS and can lead to lower estimation errors overall. We used a synthetic data setting (explained in detail in the Appendix), where we collect logging data $\mathcal{D}$ from a linear stochastic logging policy that plays a multi-armed bandit environment for $n = 300$ rounds. Based on $\mathcal{D}$, we estimate the expected reward of a new target policy using clipped IPS evaluators with different clipping constants. For dcIPS, we choose the heuristic to move U and L in unison (i.e., becoming a single hyper-parameter), but more sophisticated methods to select U and L should be investigated. We show the distribution of reward estimates (Fig. 1a) and estimation error components (Fig. 1b) as a function of the clipping constants. The figures are best interpreted in conjunction and going from right to left on the x-axis: for large ($U = L = 100$), both cIPS and dcIPS are basically equivalent to the unclipped IPS estimator: they are unbiased but show high variance. As the clipping constants decrease, the variances of both estimates (dashed lines in 1b) decreases monotonically, whereas the biases (dotted lines in 1b) increase. The lower clipping of the dcIPS compensates some of the large bias suffered by the cIPS evaluator (for a given point on the x-axis, both estimators use the same upper clipping constant U and thus the difference in biases reflects the bias compensation from using lower clipping as well). Thanks to this bias compensation, the dcIPS evaluator leads to lower MSE overall (solid lines in 1b).

6 DISCUSSION AND OUTLOOK

We analyze the bias of the clipped IPS estimator and find that negative bias provides potential for less-biased variance reduction techniques. We propose a simple method, doubly-clipped IPS, that can compensate the bias of single clipping.

One limitation is that we lack a mechanism to select clipping constants. We plan to study algorithms to select clipping constants for dcIPS in a data-driven way [1, 14, 19] and investigate theoretically when the bias of double clipping is less than standard clipping.

ACKNOWLEDGMENTS

We thank 3 anonymous reviewers for their correction of a false statement and their useful suggestions. We also thank Yannik Stein, Vito Bellini, Matej Jakimov, Thorsten Joachims, and Harrie Oosterhuis for fruitful discussion and valuable feedback given in the context of an early talk about this project.

REFERENCES

[1] Oliver Bembom and Mark J van der Laan. 2008. Data-adaptive selection of the truncation level for inverse-probability-of-treatment-weighted estimators. (2008).

[2] Léon Bottou, Jonas Peters, Joaquin Quiñonero-Candela, Denis X Charles, D Max Chickering, Elon Portugaly, Dipankar Ray, Patrice Simard, and Ed Snelson. 2013. Counterfactual Reasoning and Learning Systems: The Example of Computational Advertising. *Journal of Machine Learning Research* 14, 11 (2013).

[3] Alexander Buchholz, Ben London, Giuseppe Di Benedetto, and Thorsten Joachims. 2022. Off-policy evaluation for learning-to-rank via interpolating the item-position model and the position-based model. *arXiv preprint arXiv:2210.09512* (2022).

[4] Miroslav Dudík, John Langford, and Lihong Li. 2011. Doubly robust policy evaluation and learning. *arXiv preprint arXiv:1103.4601* (2011).

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

[6] Guido W Imbens and Donald B Rubin. 2015. *Causal inference in statistics, social, and biomedical sciences.* Cambridge University Press.

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

[8] Thorsten Joachims, Adith Swaminathan, and Tobias Schnabel. 2017. Unbiased learning-to-rank with biased feedback. In *Proceedings of the tenth ACM international conference on web search and data mining*. 781–789.

[9] Shuai Li, Yasin Abbasi-Yadkori, Branislav Kveton, Shan Muthukrishnan, Vishwa Vinay, and Zheng Wen. 2018. Offline evaluation of ranking policies with click models. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1685–1694.

[10] Jan Malte Lichtenberg and Özgür Şimşek. 2019. Regularization in directable environments with application to Tetris. In *International Conference on Machine Learning*. PMLR, 3953–3962.

[11] Harrie Oosterhuis. 2023. Doubly Robust Estimation for Correcting Position Bias in Click Feedback for Unbiased Learning to Rank. *ACM Transactions on Information Systems* 41, 3 (2023), 1–33.

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

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

[14] Yi Su, Pavithra Srinath, and Akshay Krishnamurthy. 2020. Adaptive estimator selection for off-policy evaluation. In *International Conference on Machine Learning*. PMLR, 9196–9205.

[15] Yi Su, Lequn Wang, Michele Santacatterina, and Thorsten Joachims. 2019. Cab: Continuous adaptive blending for policy evaluation and learning. In *International Conference on Machine Learning*. PMLR, 6005–6014.

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

[17] Adith Swaminathan and Thorsten Joachims. 2015. The self-normalized estimator for counterfactual learning. *advances in neural information processing systems* 28 (2015).

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

[19] Takuma Udagawa, Haruka Kiyohara, Yusuke Narita, Yuta Saito, and Kei Tateno. 2023. Policy-adaptive estimator selection for off-policy evaluation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 10025–10033.

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

A EXPERIMENTAL SETUP

We used the following classic synthetic data setting for the off-policy evaluation experiments in Section 5.

We started by collected a logging data set $\mathcal{D}$ from a linear stochastic logging policy that played a multi-armed bandit environment for $n = 300$ rounds. More specifically, the environment had $|\mathcal{Y}| = 8$ actions. Each action y_j was represented by a contextual feature vector $\phi_j \in \mathbb{R}^8$, which was drawn from a normal distribution with mean equal

to the 1-hot encoding of the action y_j (i.e., $\phi_j[k] = 0$ in all positions $k = [1, ..., 8], k \neq j$ and $\phi_j[k] = 1$ for $k = j$), and standard deviation $\sigma = 1.0$. (Thus, context is defined implicitly via the action features.)

The logging policy was a linear policy with weights $\beta_{\text{logging}} = [\frac{1}{9}, \frac{2}{9}, \frac{3}{9}, \ldots, \frac{8}{9}]^T$ and selected actions by sampling from a softmax distribution over the scores $\Phi\beta_{\text{logging}}$, where Φ is "feature matrix" obtained by concatenating the feature vectors ϕ_j for all actions.

The target policy was a similar linear policy but with "flipped" weights $\beta_{\text{target}} = [\frac{8}{9}, \frac{7}{9}, \frac{6}{9}, \ldots, \frac{1}{9}]^T$.

This lead to a situation where both logging and target policy had full support (all actions had a positive probability of being selected) but the logging policy "favored" actions in order a_8, a_7, a_6, ..., a_1, whereas the target policy "favored" actions in order a_1, a_2, a_3, ..., a_8.

The reward function was again based on a linear function of the action features with weight vector $\beta_{\text{reward}} = [0, 0.5, 0.0.5, 0, 0, 0, 0]$ and providing a reward of 1 for action j if $\phi_j^T \beta_{\text{reward}} > 0$ and 0 reward otherwise. (This reward is stochastic because the action features are stochastic.) Note that this reward function lead to a higher expected reward for the target policy compared to the logging policy (as can be seen in Figure 1a).

The data set $\mathcal{D} = \{(x_i, y_i, r_i, \pi_0(\cdot|x_i)\}, i = 1, \ldots 300$ collected by the logging policy was then used to estimate the target policy reward using different IPS estimators.

B PROOF OF BIAS OF CIPS (PROPOSITION 3.1)

Proof. For the proof we widely follow the bias expressions in [4, 15]. We recall the full support assumption $w(x, y) > 0 \; \forall x, y$ (also termed *absolute continuity*). In all steps of the proof we use the full support assumption to avoid division by 0. Recall the expression for cIPS: $\hat{R}_{\text{cIPS}}(\pi, U) = \frac{1}{n} \sum_{i=1}^{n} r(x_i, y_i) \min\{w(x_i, y_i), U\}$. We restrict our proof to one single sample (i.e., $n = 1$). The application to the average follows from the linearity of the expectation. Our proof is based on the idea that we can decompose the event

$$\min\{w(x, y), U\} = \mathbf{1}_{\{w(x,y)>U\}} U + \mathbf{1}_{\{w(x,y)<U\}} w(x, y).$$

In the same spirit we can rewrite

$$w(x, y) = \mathbf{1}_{\{w(x,y)>U\}} w(x, y) + \mathbf{1}_{\{w(x,y)<U\}} w(x, y).$$

Based on the two formulations above we get

$$\begin{aligned}
\text{Bias}(\hat{R}_{\text{cIPS}}(\pi, U)) &= \mathbb{E}[r(x, y) \min\{w(x, y), U\} - r(x, y) w(x, y)], \\
&= \mathbb{E}[r(x, y) \mathbf{1}_{\{w(x,y)>U\}} (U - w(x, y))], \\
&= \mathbb{E}\left[r(x, y) \mathbf{1}_{\{w(x,y)>U\}} w(x, y) \left(\frac{U}{w(x, y)} - 1\right)\right].
\end{aligned}$$

The bias is determined by the weights for which the upper clipping constant U is exceeded. Now, by rearranging the terms and applying the expectation to the reward we get

$$\begin{aligned}
\text{Bias}(\hat{R}_{\text{cIPS}}(\pi, U)) &= \mathbb{E}_x \mathbb{E}_{y \sim \pi_0}\left[\mathbf{1}_{\{w(x,y)>U\}} w(x, y) \left(\frac{U}{w(x, y)} - 1\right) \mathbb{E}_r[r(x, y)|x, y]\right], \\
&= \mathbb{E}_x \mathbb{E}_{y \sim \pi_0}\left[\mathbf{1}_{\{w(x,y)>U\}} \frac{\pi(y|x)}{\pi_0(y|x)} \left(\frac{U}{w(x, y)} - 1\right) \mathbb{E}_r[r(x, y)|x, y]\right], \\
&= \mathbb{E}_x \mathbb{E}_{y \sim \pi}\left[\mathbf{1}_{\{w(x,y)>U\}} \left(\frac{U}{w(x, y)} - 1\right) \mathbb{E}_r[r(x, y)|x, y]\right].
\end{aligned}$$

In the last line we used the importance sampling identity, i.e., $\mathbb{E}_{y\sim\pi_0}[f(y)\frac{\pi(y|x)}{\pi_0(y|x)}] = \mathbb{E}_{y\sim\pi}[f(y)]$, for some function f of y, which concludes our proof. □

C PROOF OF BIAS OF DCIPS (4.1)

Proof. First note that we assume $U > \frac{1}{L}$; i.e., the upper clipping constant always needs to be larger than the lower clipping constant. Note that in all steps of the proof we use the full support assumption to avoid division by 0. We will again focus on the expectation of a single sample, the application to the average is immediate. In the same spirit of the previous proof we rewrite

$$\max\left\{\min\{w(x_i,y_i),U\},\frac{1}{L}\right\} = \mathbf{1}_{\{w(x,y)>U\}}U + \mathbf{1}_{\{1/L<w(x,y)<U\}}w(x,y) + \mathbf{1}_{\{1/L>w(x,y)\}}1/L,$$

i.e., we clip the weights $w(x,y)$ if they fall outside the interval $[1/L, U]$. Again, we apply the same reasoning to the unclipped weights and get

$$w(x,y) = \mathbf{1}_{\{w(x,y)>U\}}w(x,y) + \mathbf{1}_{\{1/L<w(x,y)<U\}}w(x,y) + \mathbf{1}_{\{1/L>w(x,y)\}}w(x,y).$$

Now, we apply this to the computation of the bias of dcIPS and get

$$\begin{aligned} Bias(\hat{R}_{\text{dcIPS}}(\pi,U,L)) &= \mathbb{E}\left[r(x,y)\left(\max\{\min\{w(x_i,y_i),U\},\frac{1}{L}\} - w(x,y)\right)\right] \\ &= \mathbb{E}\left[r(x,y)\left(\mathbf{1}_{\{w(x,y)>U\}}(U-w(x,y)) + \mathbf{1}_{\{1/L>w(x,y)\}}(1/L-w(x,y))\right)\right]. \end{aligned}$$

Rearranging terms, applying the expectation to the reward and factoring out $w(x,y)$ then yields

$$\begin{aligned} Bias(\hat{R}_{\text{dcIPS}}(\pi,U,L)) &= \mathbb{E}_x\mathbb{E}_{y\sim\pi_0}\left[w(x,y)\left(\mathbf{1}_{\{w(x,y)>U\}}\left(\frac{U}{w(x,y)}-1\right) + \mathbf{1}_{\{w(x,y)L<1\}}\left(\frac{1}{w(x,y)L}-1\right)\right)\mathbb{E}_r[r(x,y)|x,y]\right], \\ &= \mathbb{E}_x\mathbb{E}_{y\sim\pi}\left[\left(\mathbf{1}_{\{w(x,y)>U\}}\left(\frac{U}{w(x,y)}-1\right) + \mathbf{1}_{\{w(x,y)L<1\}}\left(\frac{1}{w(x,y)L}-1\right)\right)\mathbb{E}_r[r(x,y)|x,y]\right], \end{aligned}$$

where, as before, we used the importance sampling identity to rewrite the expectation with respect to the target distribution, which completes the proof. □