On Exploration, Exploitation and Learning in Adaptive Importance Sampling

Xiaoyu Lu[1] **Tom Rainforth**[1] **Yuan Zhou**[1] **Yee Whye Teh**[1] **Frank Wood**[1]
Hongseok Yang[2] **Jan-Willem van de Meent**[3]
[1]University of Oxford; [2]KAIST, South Korea; [3]Northeastern University
{xiaoyu.lu, rainforth, y.w.teh}@stats.ox.ac.uk, yuan.zhou@cs.ox.ac.uk,
fwood@robots.ox.ac.uk,hongseok.yang@kaist.ac.kr,
j.vandemeent@northeastern.edu

1 Introduction

Monte Carlo methods form the bedrock upon which significant sections of probabilistic machine learning and computational statistics rest. An important Monte Carlo technique which forms the basis for many others is importance sampling (IS). Let $\pi(x) = f(x)/Z$ be a target density which can be evaluated pointwise up to an unknown normalising constant Z, and let $q(x)$ be a proposal distribution from which samples can be drawn and which can be evaluated pointwise. IS works by drawing a sequence of samples $x_1, x_2, \ldots$ from the proposal $q(x)$, and using these to estimate both Z and target statistics $\mathbb{E}_\pi[\phi(x)]$ for some test function $\phi(x)$. Let $w(x_t) = f(x_t)/q(x_t)$ be the importance weight of x_t. Then,

$$Z = \int f(x)dx = \mathbb{E}_q[w(X)] \approx \frac{1}{T}\sum_{t=1}^{T} w(x_t),$$

$$\mathbb{E}_\pi[\phi(x)] = \frac{\int f(x)\phi(x)dx}{\int f(x)dx} = \frac{\mathbb{E}_q[w(X)\phi(X)]}{\mathbb{E}_q[w(X)]} \approx \frac{\sum_{t=1}^{T} w(x_t)\phi(x_t)}{\sum_{t=1}^{T} w(x_t)}. \tag{1}$$

Note that the estimate for Z is unbiased, but that for the target statistics is biased but consistent.

The efficiency of IS is governed by the choice of proposal $q(x)$, with the intuition that the closer q is to π the better. Adaptive IS (AdaIS) techniques [1, 2, 3, 4] attempt to improve the efficiency of IS by adapting the proposal to be closer to the target, producing a sequence of proposals $q_1, q_2, \ldots$. The IS estimates (1) still apply with $q(x_t)$ replaced by $q_t(x_t)$. The basic idea is that previous samples x_s along with evalutions $f(x_s)$ give information about the distribution of probability masses in the target, and the proposal should place more mass where the target has more mass.

Viewed in this way, AdaIS is in effect an online learning problem, that of learning the target density π through a sequence of queries of it. As opposed to the typical setup of density estimation where each query is an iid sample from the target, here each query involves drawing a sample x_t from the current proposal q_t, and evaluating $f(x_t)$, the target density up to an unknown normalising constant. At each iteration, the proposal q_t is both our current estimate of the target, as well as our tool for querying the target.

Note that this exposes a trade off between exploration and exploitation. We would like our proposal q_t to be as close as possible to the target, so that our IS estimate is as good as possible (exploit). At the same time, q_t directs where queries of the target are made, and probability mass needs to be spread over the sample space where we have high uncertainty of the target so that we may query and reduce our uncertainty to improve our estimate of the target for the future (explore).

2 A Bandits Approach to Adaptive Important Sampling

In this paper we take the first steps towards developing an AdaIS method which optimally trades off exploration versus exploitation. Methods which address the exploration-exploitation trade off has been well-studied in online learning, most successfully under the banner of bandit algorithms [5],

Submitted to 31st Conference on Neural Information Processing Systems (NIPS 2017). Do not distribute.

with the Upper Confidence Bound (UCB) methods [6] and Thompson sampling methods [7] being popular approaches. In (basic) UCB, a finite number of arms are present, and at each iteration an arm is chosen to be pulled, which then returns a random reward. The aim to maximise rewards in the long run, by finding the arm with highest average reward. UCB operates by maintaining an estimate of the average reward for each arm, and arms are picked according to the estimates plus optimism boosts which are larger for arms where our estimates are less certain to encourage exploration. [6] showed that UCB optimally trades-off exploration and exploitation by showing that the cumulative regret (relative to an oracle which knows which arm is optimal) grows logarithmically in the number of iterations, which is the best growth rate achievable [8].

In this section we will develop an AdaIS method which has a similar flavour to UCB. We assume that we have a partition of our sample space into K disjoint subsets (corresponding to bandit arms), and that we have a tractable base distribution $g_a(x)$ for each subset indexed by $a \in \{1, \dots, K\}$. We consider proposal distributions of the form

$$q_t(x) = \sum_{a=1}^{K} q_{at} g_a(x) \tag{2}$$

where $\sum_a q_{at} = 1$ and the probability masses q_{at} of the subsets are to be adapted in the scheme. In the next section we will extend the approach to a hierarchical partition of the sample space, where subsets are recursively split where necessary.

A measure of the (in)efficiency of the proposals q_t is also required so that we can define what makes an optimal proposal among the class above, and the regret of using a proposal relative to the optimal. We will use the KL divergence $\mathsf{KL}(\pi \| q_t)$ as this measure, which has been shown by Chatterjee and Diaconis [9] to be the correct measure of the inefficiency of an IS with proposal q_t. For proposals as given by (2), this is,

$$\mathsf{KL}(\pi \| q_t) = \sum_a \int_{\mathcal{X}_a} \pi(x) \log \frac{\pi(x)}{g_a(x)} dx - \sum_a \pi_a \log q_{at} \tag{3}$$

where $\pi_a = \int_{\mathcal{X}_a} \pi(x) dx = Z_a / Z$ is the target probability mass of subset $\mathcal{X}_a$ and Z_a is the corresponding subset partition function. The optimal parameters q_{at} are seen to be $q_a^* = \pi_a$, and the regret $R(q_t)$ of proposal q_t is,

$$R(q_t) = \mathsf{KL}(\pi \| q_t) - \mathsf{KL}(\pi \| q^*) = \sum_a \pi_a \log \frac{\pi_a}{q_{at}}, \tag{4}$$

which is just the KL divergence between the two finite vectors of probability masses.

At iteration t, a query of the target starts with a sample x_t from q_t, which involves first picking a subset A_t according to the probability masses q_{at}, then drawing a sample $x_t \in \mathcal{X}_{A_t}$ from $g_{A_t}(x)$. This then informs our knowledge of the target probability mass of subset $\mathcal{X}_{A_t}$. For each subset a we have $\pi_a = Z_a / \sum_b Z_b$ where

$$Z_a = \int_{\mathcal{X}_a} f(x) dx = \mathbb{E}_{g_a} \left[\frac{f(x)}{g_a(x)} \right]. \tag{5}$$

And so, if $A_t = a$, the importance weight $Y_{at} := f(x_t)/g_a(x_t)$ of the proposal $g_a(x)$ for subset a gives an unbiased estimate of Z_a.

Naively, for the next iteration we can now estimate each Z_a using

$$\hat{Z}_{a,t+1} := \frac{\sum_{l \le t : A_l = a} Y_{al}}{N_{a,t+1}} \tag{6}$$

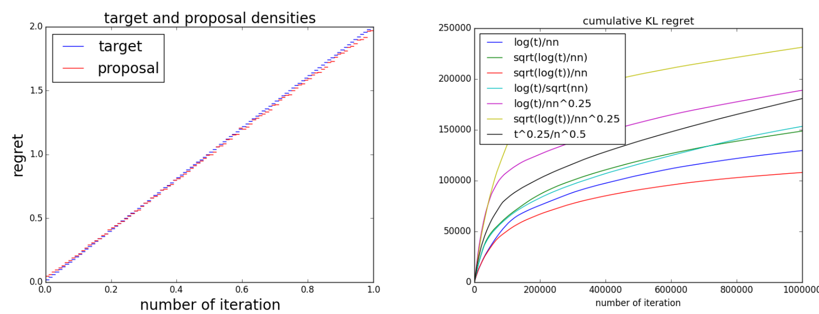
where $N_{a,t+1} = \#\{l : l \le t, A_l = a\}$ is the number of times subset a has been chosen up to time t, and use the estimate $q_{a,t+1} = \hat{Z}_{a,t+1} / \sum_b \hat{Z}_{b,t+1}$. A problem with this naive scheme is that if by chance our estimate $\hat{Z}_{at}$ for some subset a is too small, the resulting low estimated proposal probablity will result in low probability for the subset to be picked in future, and hence the bad estimate may not be corrected. This is a symptom of under-exploration. As in UCB, we will consider encouraging exploration by using an optimism boost:

$$q_{a,t+1} = \frac{\hat{Z}_{a,t+1} + \sigma_{a,t+1}}{\sum_{a=1}^{K} (\hat{Z}_{a,t+1} + \sigma_{a,t+1})} \tag{7}$$

where σ_{at} should be decreasing with N_{at} but grows with t. The intuition is that if we have not explored the subset a sufficiently, σ_{at} is relatively large, which compensates and boosts q_{at}, allowing us to have higher chance to explore subset a and correct the under estimate. The growth with t is to ensure sufficient exploration of all subsets over time.

Note that the estimate $\hat{Z}_{a,t+1}$ above is in fact biased, since $N_{a,t+1}$ is random and correlated with previous values of Y_{al} for $l \leq t$[1]. If it were unbiased, it is possible to bound the regret (4) by using concentration inequalities to bound the probability of large deviations. It is our ongoing work to fix this.

We demonstrate our method on a simple problem, and empirically evaluate a number forms of the optimism boost. Our sample space is the unit interval $[0, 1]$, and we partitioned it evenly into 100 subintervals. We picked the target density $\pi(x)$ and the subproposals $g_a(x)$ such that Y_{at} has distribution $2a\text{Bernoulli}(1/100)/101$ for interval $a = \{1, \ldots, 100\}$. We made this choice so that Y_{at} has a large but controllable variance, so that the resulting problem of adaptation is hard enough for exploration to be important. Figure 1a shows that our algorithm is able to adapt the proposals so that they converge to the optimum by balancing between exploitation and exploration well. Note that low probability intervals have their mass over-estimated by the proposal due to exploration. We compare the cumulative regret for different forms of optimism boosts σ_{at} in Figure 1b. For each form we have optimised over a constant multiplier to minimise cumulative regret. We observe that optimism boosts with an inverse relationship with N_{at} and slow growth with t work well, and seem to achieve sublinear cumulative regret.



(a) Target and proposal probabilities at final iteration, for optimism boost $\log(t)/\sqrt{N_{at}}$.

(b) Cumulative regrets as functions of iteration.

Figure 1: Results for our AdaIS method, with 100 subsets, averaged over 10 runs.

3 Hierarchical Partition

Instead of fixing the number of partitions of the sample space to be K, we extend the approach to a binary hierarchical partition of the sample space using the *Polyá tree* idea from Bayesian nonparametrics [10][11] and the UCT algorithm[12]. The idea is that the subsets are recursively split when the relative masses are high in the subset, enabling a finer partioning over the sample space which has higher density. We now explain in detail of how the algorithm works.

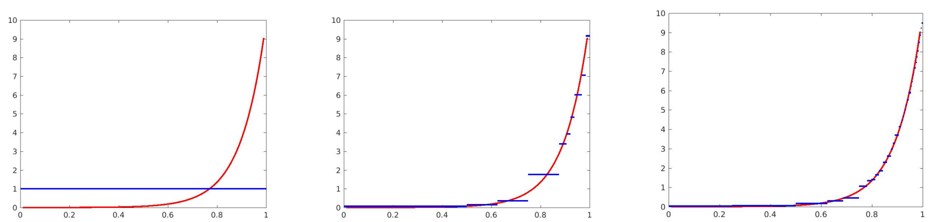
For simplicity, consider the sample space $\mathcal{X}$ to be the interval $[0, 1]$, the idea is to conduct a hierarchical binary partition where the interval is recursively evenly split into left or right subspace, inducing a tree structure. At each iteration, sampling from the root $\mathcal{X}$ down the tree according to the probability of going left or right at each node, which are computed using samples already drawn, until a leaf (a subspace) is visited. Draw a new sample in that subspace with a default proposal (e.g. uniform), and keep track of whether the new sample falls into the left or the right of the leaf node, as well as its importance weights. Relevant statistics are then updated along the path to the root to update the sequence of probabilities of going left/right. Now one can decide whether to further partition the leaf node according to some splitting criterion. For example, when the tree reaches some truncation level, $i.e.$, the depth of the tree is fixed; or more intuitively, we partition the leaf node further when it reduces the KL-divergence without compromising too much computational costs[9].

[1]We thank Tom Rainforth for pointing this out to us.

We demonstrate the algorithm on a simple example, consider the target density to be $\pi(x) \propto \exp(10(x-1))\mathbf{1}_{x\in(0,1)}$, and we set the truncation level to be 10. The result can be seen in figure 2. It can be seen that more intervals have been partitioned over higher mass regions, and the algorithm is able to adapt the proposal distributions according to the relative masses it has already obtained. We also applied the algorithm to the classic $2D$ banana shaped problem, with density $f(x_1, x_2) \propto \exp\{-0.5(0.03x_1^2 + (x_2 + 0.03(x_1^2 - 100))^2)\}$, where we used the KL-divergence splitting criterion. It can be seen from figure 3 that we successfully recover the density within $100,000$ samples, and the the region with low density does not get splitted whereas the region with high density has finer grid for more accurate estimates. Moreover, the region with high density stops splitting infinitely to balance with the computational expense.

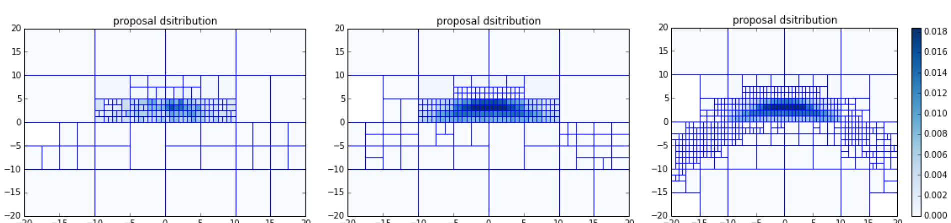
4 Conclusion

In this work, we have addressed the issue of exploration-exploitation in adaptive importance sampling, and proposed a novel approach through the lens of multi-armed bandit problems, borrowing the ideas of upper confidence bounds. We extend our method to the hierarchical case, where the sample space is recursively split in high density regions, and demonstrated experimentally that our method gives promising performance with little computational costs. In ongoing work we are investigating unbiased estimates of Z_a and a finite-time theoretical analysis to understand the growth of the cumulative regret. So far as we know, the concept of trading off exploration versus exploitation has not been widely considered in adaptive Monte Carlo, except for [13][14]. [13] considers a number of Monte Carlo estimates, where the idea is to pick the one with lowest variance with the goal of picking the optimal estimate as much as possible. [14] connsiders the problem of stratified sampling, where each strata is viewed as an arm and the mean in each strata is estimated. Both cases are closer to the bandit problem setting whereas our importance sampelr does not choose the arm but learns a distribution over arms instead.



(a) (b) (c)

Figure 2: Results using the hierarchical algorithm: target and proposal density, the target density are plotted in red, and the adaptive proposal probabilities are plotted in blue, at iteration 1,(a); 50,(b); and 200,(c) respectively.



(a) (b) (c)

Figure 3: Banana shaped example, learned proposal distributions and partitions at iteration 1000,(a); 10000,(b); 100000,(c) respectively. Darker color indicated high density.

References

[1] Jun S Liu. *Monte Carlo strategies in scientific computing*. Springer Science & Business Media, 2008.

[2] Olivier Cappé, Arnaud Guillin, Jean-Michel Marin, and Christian P Robert. Population monte carlo. *Journal of Computational and Graphical Statistics*, 13(4):907–929, 2004.

[3] Jean Cornuet, JEAN-MICHEL MARIN, Antonietta Mira, and Christian P Robert. Adaptive multiple importance sampling. *Scandinavian Journal of Statistics*, 39(4):798–812, 2012.

[4] Olivier Cappé, Randal Douc, Arnaud Guillin, Jean-Michel Marin, and Christian P Robert. Adaptive importance sampling in general mixture classes. *Statistics and Computing*, 18(4):447–459, 2008.

[5] Donald A Berry and Bert Fristedt. *Bandit problems: sequential allocation of experiments (Monographs on statistics and applied probability)*. Springer, 1985.

[6] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002.

[7] Shipra Agrawal and Navin Goyal. Analysis of thompson sampling for the multi-armed bandit problem. In *COLT*, pages 39–1, 2012.

[8] Tze Leung Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22, 1985.

[9] Sourav Chatterjee and Persi Diaconis. The sample size required in importance sampling. *arXiv preprint arXiv:1511.01437*, 2015.

[10] R Daniel Mauldin, William D Sudderth, and SC Williams. Polya trees and random distributions. *The Annals of Statistics*, pages 1203–1221, 1992.

[11] R Daniel Mauldin and SC Williams. Reinforced random walks and random distributions. *Proceedings of the American Mathematical Society*, 110(1):251–258, 1990.

[12] Levente Kocsis and Csaba Szepesvári. Bandit based monte-carlo planning. In *ECML*, volume 6, pages 282–293. Springer, 2006.

[13] James Neufeld, András György, Dale Schuurmans, and Csaba Szepesvári. Adaptive monte carlo via bandit allocation. *arXiv preprint arXiv:1405.3318*, 2014.

[14] Alexandra Carpentier and Rémi Munos. Finite time analysis of stratified sampling for monte carlo. In *Advances in Neural Information Processing Systems*, pages 1278–1286, 2011.