
Batch Stationary Distribution Estimation

Junfeng Wen^{*1} Bo Dai^{*2} Lihong Li² Dale Schuurmans^{1,2}

Abstract

We consider the problem of approximating the stationary distribution of an ergodic Markov chain given a set of sampled transitions. Classical simulation-based approaches assume access to the underlying process so that trajectories of sufficient length can be gathered to approximate stationary sampling. Instead, we consider an alternative setting where a *fixed* set of transitions has been collected beforehand, by a separate, possibly unknown procedure. The goal is still to estimate properties of the stationary distribution, but without additional access to the underlying system. We propose a consistent estimator that is based on recovering a correction ratio function over the given data. In particular, we develop a variational power method (VPM) that provides provably consistent estimates under general conditions. In addition to unifying a number of existing approaches from different subfields, we also find that VPM yields significantly better estimates across a range of problems, including queueing, stochastic differential equations, post-processing MCMC, and off-policy evaluation.

1. Introduction

Markov chains are a pervasive modeling tool in applied mathematics of particular importance in stochastic modeling and machine learning. A key property of an *ergodic* Markov chain is the existence of a unique *stationary distribution*; *i.e.*, the long-run distribution of states that remains invariant under the transition kernel. In this paper, we consider a less well studied but still important version of the stationary distribution estimation problem, where one has access to a set of sampled transitions from a given Markov chain, but does not know the mechanism by which the probe points were chosen, nor is able to gather additional data from the

underlying process. Nevertheless, one would still like to estimate target properties of the stationary distribution, such as the expected value of a random variable of interest.

This setting is inspired by many practical scenarios where sampling from the Markov process is costly or unavailable, but data has already been collected and available for analysis. A simple example is a queueing system consisting of a service desk that serves customers in a queue. Queue length changes stochastically as customers arrive or leave after being served. The long-term distribution of queue length (*i.e.*, the stationary distribution of the underlying Markov chain) is the object of central interest for managing such a service (Haviv, 2009; Serfozo, 2009). In practice, however, queue lengths are physical quantities that can only be measured for moderate periods, perhaps on separate occasions, but rarely for sufficient time to ensure the (stochastic) queue length has reached the stationary distribution. Since the measurement process itself is expensive, it is essential to make reasonable inferences about the stationary distribution from the collected data alone.

We investigate methods for estimating properties of the stationary distribution solely from a batch of previously collected data. The key idea is to first estimate a correction ratio function over the given data, which can then be used to estimate expectations of interest with respect to the stationary distribution. To illustrate, consider an ergodic Markov chain with state space \mathcal{X} , transition kernel \mathcal{T} , and a unique stationary distribution μ that satisfies

$$\mu(x') = \int \mathcal{T}(x'|x) \mu(x) dx := (\mathcal{T}\mu)(x'). \quad (1)$$

Assume we are given a *fixed* sample of state transitions, $\mathcal{D} = \{(x, x')_{i=1}^n\} \sim \mathcal{T}(x'|x) p(x)$, such that each x has been sampled according to an *unknown* probe distribution p , but each x' has been sampled according to the true underlying transition kernel, $x'|x \sim \mathcal{T}(x'|x)$. Below we investigate procedures for estimating the point-wise ratios, $\hat{\tau}(x_i) \approx \frac{\mu(x_i)}{p(x_i)}$, such that the weighted empirical distribution

$$\hat{\mu}(x) := \left(\sum_{i=1}^n \hat{\tau}(x_i) \right)^{-1} \sum_{i=1}^n \hat{\tau}(x_i) \mathbb{I}\{x = x_i\}$$

can be used to approximate μ directly, or further used to estimate the expected value of some target function(s) of x

^{*}Equal contribution ¹Department of Computing Science, University of Alberta, Edmonton, Canada ²Google Brain. Correspondence to: Junfeng Wen <junfengwen@gmail.com>.

with respect to μ . Crucially, the approach we propose does not require knowledge of the probe distribution p , nor does it require additional access to samples drawn from the transition kernel \mathcal{T} , yet we will be able to establish consistency of the estimation strategy under general conditions.

In addition to developing the fundamental approach, we demonstrate its applicability and efficacy in a range of important scenarios beyond queueing, including:

- **Stochastic differential equations (SDEs)** SDEs are an essential modeling tool in many fields like statistical physics (Kadanoff, 2000), finance (Oksendal, 2013) and molecular dynamics (Liu, 2001). An autonomous SDE describes the instantaneous change of a random variable X by

$$dX = f(X) dt + \sigma(X) dW, \quad (2)$$

where $f(X)$ is a drift term, $\sigma(X)$ a diffusion term, and W the Wiener process. Given data $\mathcal{D} = \{(x, x')_{i=1}^n\}$ such that $x \sim p(x)$ is drawn from an unknown probe distribution and x' is the next state after a small time step according to (2), we consider the problem of estimating quantities of the stationary distribution μ when one exists.

- **Off-policy evaluation (OPE)** Another important application is *behavior-agnostic off-policy evaluation* (Nachum et al., 2019) in reinforcement learning (RL). Consider a Markov decision process (MDP) specified by $M = \langle \mathcal{S}, \mathcal{A}, P, R \rangle$, such that \mathcal{S} and \mathcal{A} are the state and action spaces, P is the transition function, and R is the reward function (Puterman, 2014). Given a policy π that maps $s \in \mathcal{S}$ to a distribution over \mathcal{A} , a random trajectory can be generated starting from an initial state s_0 : $(s_0, a_0, r_0, s_1, a_1, r_1, \dots)$, where $a_t \sim \pi(\cdot|s_t)$, $s_{t+1} \sim P(\cdot|s_t, a_t)$ and $r_t \sim R(s_t, a_t)$. The *value* of a policy π is defined to be its long-term average per-step reward:

$$\rho(\pi) := \lim_{T \rightarrow \infty} \mathbb{E} \left[\frac{1}{T} \sum_{t=0}^{T-1} r_t \right] = \mathbb{E}_{(s,a) \sim d_\pi \circ \pi} [R(s, a)],$$

where d_π denotes the limiting distribution over states \mathcal{S} of the Markov process induced by π . In behavior-agnostic off-policy evaluation, one is given a target policy π and a set of transitions $\mathcal{D} = \{(s, a, r, s')_{i=1}^n\} \sim P(s'|s, a) p(s, a)$, potentially generated by multiple behavior policies. From such data, an estimate for $\rho(\pi)$ can be formed in terms of a stationary ratio estimator:

$$\rho(\pi) = \mathbb{E}_{(s,a) \sim p} \left[\frac{d_\pi(s) \pi(a|s) r(s, a)}{p(s, a)} \right] \approx \frac{1}{n} \sum_{i=1}^n \hat{\tau}(s_i, a_i) r_i. \quad (3)$$

We refer the interested readers to Section 5.4 and Appendix C for further discussion.

For the remainder of the paper, we will outline four main contributions. First, we generalize the classical power iteration method to obtain an algorithm, the *Variational Power*

Method (VPM), that can work with arbitrary parametrizations in a functional space, allowing for a flexible yet practical approach. Second, we prove the consistency and convergence of VPM. Third, we illustrate how a diverse set of stationary distribution estimation problems, including those above, can be addressed by VPM in a unified manner. Finally, we demonstrate empirically that VPM significantly improves estimation quality in a range of applications, including queueing, sampling, SDEs and OPE.

2. Variational Power Method

To develop our approach, first recall the definition of \mathcal{T} and μ in (1). We make the following assumption about \mathcal{T} and μ throughout the paper.

Assumption 1 (ergodicity) *The transition operator \mathcal{T} has a unique stationary distribution, denoted μ .*

Conditions under which this assumption holds are mild, and have been extensively discussed in standard textbooks (Meyn et al., 2009; Levin and Peres, 2017).

Next, to understand the role of the probe distribution p , note that we can always rewrite the stationary distribution as $\mu = p \circ \tau$ (i.e., $\mu(x) = p(x) \tau(x)$, hence $\tau(x) = \frac{\mu(x)}{p(x)}$), provided the following assumption holds.

Assumption 2 (absolute continuity) *The stationary distribution μ is absolutely continuous w.r.t. p . That is, there exists $C < \infty$ such that $\|\tau\|_\infty \leq C$.*

Assumption 2 follows previous work (Liu and Lee, 2017; Nachum et al., 2019), and is common in density ratio estimation (Sugiyama et al., 2008; Gretton et al., 2009) and off-policy evaluation (Wang et al., 2017; Xie et al., 2019).

Combining these two assumptions, definition (1) yields

$$\begin{aligned} \mu(x') &= \int \mathcal{T}(x'|x) \mu(x) dx = \int \mathcal{T}(x'|x) p(x) \frac{\mu(x)}{p(x)} dx \\ &:= \int \mathcal{T}_p(x, x') \tau(x) dx, \quad \text{which implies} \\ p(x') \tau(x') &= \int \mathcal{T}_p(x, x') \tau(x) dx := \mathcal{T}_p \tau(x'). \end{aligned} \quad (4)$$

This development reveals how, under the two stated assumptions, there is sufficient information to determine the unique ratio function τ that ensures $p \circ \tau = \mu$ in principle. Given such a function τ , we can then base inferences about μ solely on data sampled from p and τ .

2.1. Variational Power Iteration

To develop a practical algorithm for recovering τ from the constraint (4), in function space, we first consider the classical power method for recovering the μ that satisfies (1).

From (1) it can be seen that the stationary distribution μ is an eigenfunction of \mathcal{T} . Moreover, it is the *principal* eigenfunction, corresponding to the largest eigenvalue $\lambda_1 = 1$. In the simpler case of finite \mathcal{X} , the vector μ is the principal (right) eigenvector of the transposed transition matrix. A standard approach to computing μ is then the power method:

$$\mu_{t+1} = \mathcal{T} \mu_t, \quad (5)$$

whose iterates converge to μ at a rate linear in $|\lambda_2|$, where λ_2 is the second largest eigenvalue of \mathcal{T} . For ergodic Markov chains, one has $|\lambda_2| < 1$ (Meyn et al., 2009, Chap 20).

Our initial aim is to extend this power iteration approach to the constraint (4) without restricting the domain \mathcal{X} to be finite. This can be naturally achieved by the update

$$\tau_{t+1} = \frac{\mathcal{T}_p \tau_t}{p}, \quad (6)$$

where the division is element-wise. Clearly the fixed point of (6) corresponds to the solution of (4) under the two assumptions stated above. Furthermore, just as for μ_t in (5), τ_t in (6) also converges to τ at a linear rate for finite \mathcal{X} . Unfortunately, the update (6) cannot be used directly in a practical algorithm for two important reasons. First, we do not have a point-wise evaluator for \mathcal{T}_p , but only samples from \mathcal{T}_p . Second, the operator \mathcal{T}_p is applied to a function τ_t , which typically involves an intractable integral over \mathcal{X} in general. To overcome these issues, we propose a variational method that considers a series of reformulated problems whose optimal solutions correspond to the updates (6).

To begin to develop a practical variational approach, first note that (6) operates directly on the density ratio, which implies the density ratio estimation techniques of Nguyen et al. (2008) and Sugiyama et al. (2012) can be applied. Let ϕ be a lower semicontinuous, convex function satisfying $\phi(1) = 0$, and consider the induced f -divergence,

$$\begin{aligned} D_\phi(\tilde{p} \parallel \tilde{q}) &= \int \tilde{p}(x) \phi\left(\frac{\tilde{q}(x)}{\tilde{p}(x)}\right) dx \\ &= -\left(\min_{\nu} \mathbb{E}_{\tilde{p}}[\phi^*(\nu)] - \mathbb{E}_{\tilde{q}}[\nu]\right), \end{aligned} \quad (7)$$

where $\phi^*(x) = \sup_{y \in \mathbb{R}} x^\top y - \phi(y)$ is the conjugate function of ϕ . The key property of this formulation is that for any distributions \tilde{p} and \tilde{q} , the inner optimum in ν satisfies $\partial\phi^*(\nu) = \tilde{q}/\tilde{p}$ (Nguyen et al., 2008); that is, the optimum in (7) can be used to directly recover the distribution ratio.

To apply this construction to our setting, first consider solving a problem of the following form in the dual space:

$$\nu_{t+1} = \arg \min_{\nu} \mathbb{E}_{p(x')} [\phi^*(\nu(x'))] \quad (8)$$

$$\begin{aligned} & - \mathbb{E}_{\mathcal{T}_p(x, x')} [\partial\phi^*(\nu_t(x)) \cdot \nu(x')] \\ &= \arg \min_{\nu} \mathbb{E}_{p(x')} [\phi^*(\nu(x'))] \quad (9) \\ & - \mathbb{E}_{\mathcal{T}_p(x, x') \tau_t(x)} [\nu(x')], \end{aligned}$$

where to achieve (9) we have applied the inductive assumption that $\tau_t = \partial\phi^*(\nu_t)$. Then, by the optimality property of ν_{t+1} , we know that the solution ν_{t+1} must satisfy

$$\partial\phi^*(\nu_{t+1}) = \frac{\mathcal{T}_p \tau_t}{p} = \tau_{t+1}, \quad (10)$$

hence the updated ratio τ_{t+1} in (6) can be directly recovered from the dual solution ν_{t+1} , while also retaining the inductive property that $\tau_{t+1} = \partial\phi^*(\nu_{t+1})$ for the next iteration.

These developments can be further simplified by considering the specific choice $\phi(\tau) = (\tau - 1)^2$, which leads to $\phi^*(\nu) = \nu + \frac{\nu^2}{4}$, $\tau = \partial\phi^*(\nu) = 1 + \frac{\nu}{2}$ and

$$\tau_{t+1} = \arg \min_{\tau \geq 0} \frac{1}{2} \mathbb{E}_{p(x')} [\tau^2(x')] - \mathbb{E}_{\mathcal{T}_p(x, x')} [\tau_t(x) \tau(x')]. \quad (11)$$

Crucially, this variational update (11) determines the same update as (6), but overcomes the two aforementioned difficulties. First, it bypasses the direct evaluation of \mathcal{T}_p and p , and allows these to be replaced by unbiased estimates of expectations extracted from the data. Second, it similarly bypasses the intractability of the operator application $\mathcal{T}_p \tau_t$ in the functional space, replacing this with an expectation of $\tau_t \circ \tau$ that can also be directly estimated from the data.

We now discuss some practical refinements of the approach.

2.2. Normalization

For τ_t to be a proper ratio $\frac{\mu_t}{p}$, it should be normalized w.r.t. p , i.e. $\mathbb{E}_p[\tau_t] = 1$. To address this issue, we explicitly ensure normalization by considering a constrained optimization in place of (11):

$$\begin{aligned} \min_{\tau \geq 0} \frac{1}{2} \mathbb{E}_{p(x')} [\tau^2(x')] - \mathbb{E}_{\mathcal{T}_p(x, x')} [\tau_t(x) \tau(x')], \\ \text{s.t. } \mathbb{E}_{p(x)} [\tau(x)] = 1. \end{aligned} \quad (12)$$

We can tackle this by solving its Lagrangian. To avoid instability, we add a regularization term:

$$\begin{aligned} \min_{\tau \geq 0} \max_{v \in \mathbb{R}} J(\tau, v) = \frac{1}{2} \mathbb{E}_{p(x')} [\tau^2(x')] \\ - \mathbb{E}_{\mathcal{T}_p(x, x')} [\tau_t(x) \tau(x')] + v (\mathbb{E}_p[\tau] - 1) - \frac{\lambda}{2} v^2. \end{aligned} \quad (13)$$

where $\lambda > 0$ is a regularization parameter. Crucially, the dual variable v is a scalar, making this problem much simpler than dual embedding (Dai et al., 2017), where the dual variables form a parameterized function that introduces approximation error. The problem (13) is a straightforward convex-concave objective with respect to (τ, v) that can be optimized by stochastic gradient descent.

The following theorem shows that under certain conditions, the normalization will be maintained for any $\lambda > 0$.

Theorem 1 (Normalization of solution) *If $\mathbb{E}_p[\tau_t] = 1$, then for any $\lambda > 0$, the estimator (13) has the same solution as (12), hence $\mathbb{E}_p[\tau_{t+1}] = 1$.*

Algorithm 1 Variational Power Method

```

1: Input: Transition data  $\mathcal{D} = \{(x, x')_{i=1}^n\}$ , learning rate
    $\alpha_\theta, \alpha_v$ , number of power steps  $T$ , number of inner opti-
   mization steps  $M$ , batch size  $B$ 
2: Initialize  $\tau_\theta$ 
3: for  $t = 1 \dots T$  do
4:   Update and fix the reference network  $\tau_t = \tau_\theta$ 
5:   for  $m = 1 \dots M$  do
6:     Sample transition data  $\{(x, x')_{i=1}^B\}$ 
7:     Compute gradients  $\nabla_\theta J$  and  $\nabla_v J$  from (16)
8:      $\theta = \theta - \alpha_\theta \nabla_\theta J$  ▷ gradient descent
9:      $v = v + \alpha_v \nabla_v J$  ▷ gradient ascent
10:  end for
11: end for
12: Return  $\tau_\theta$ 

```

Hence, we can begin with any τ_0 satisfying $\mathbb{E}_p[\tau_0] = 1$ (e.g., $\tau_0 = \arg \min_\tau (\mathbb{E}_p[\tau] - 1)^2$), and the theorem ensures that the normalization of τ_{t+1} will be inductively maintained using any fixed $\lambda > 0$. The proof is given in Appendix A.

2.3. Damped Iteration

The next difficulty to be addressed arises from the fact that, in practice, we need to optimize the variational objective based on sampled data, which induces approximation error since we are replacing the true operator \mathcal{T}_p by a stochastic estimate $\hat{\mathcal{T}}_p$ such that $\mathbb{E}[\hat{\mathcal{T}}_p] = \mathcal{T}_p$. Without proper adjustment, such estimation errors can accumulate over the power iterations, and lead to inaccurate results.

To control the error due to sampling, we introduce a damped version of the update (Ryu and Boyd, 2016), where instead of performing a stochastic update $\tau_{t+1} = \frac{\hat{\mathcal{T}}_p}{p} \tau_t$, we instead perform a damped update given by

$$\tau_{t+1} = (1 - \alpha_{t+1}) \cdot \tau_t + \alpha_{t+1} \cdot \frac{\hat{\mathcal{T}}_p}{p} \tau_t \quad (14)$$

where $\alpha_t \in (0, 1)$ is a stepsize parameter. Intuitively, the update error introduced by the stochasticity of $\hat{\mathcal{T}}_p$ is now controlled by the stepsize α_t . The choice of stepsize and convergence of the algorithm is discussed in Section 3.

The damped iteration can be conveniently implemented with minor modifications to the previous objective. We only need to change the sample from \mathcal{T}_p in (13) by a weighted sample:

$$\begin{aligned} \min_{\tau \geq 0} \max_{v \in \mathbb{R}} J(\tau, v) &= \frac{1}{2} \mathbb{E}_{p(x')} [\tau^2(x')] \\ &\quad - (1 - \alpha_{t+1}) \mathbb{E}_{p(x')} [\tau_t(x') \tau(x')] \\ &\quad - \alpha_{t+1} \mathbb{E}_{\mathcal{T}_p(x, x')} [\tau_t(x) \tau(x')] + v (\mathbb{E}_p[\tau] - 1) - \frac{\lambda}{2} v^2. \end{aligned} \quad (15)$$

2.4. A Practical Algorithm

A practical version of VPM is described in Algorithm 1. It solves (15) using a parameterized $\tau : \mathcal{X} \mapsto \mathbb{R}$ expressed as

a neural network τ_θ with parameters θ . Given the constraint $\tau \geq 0$, we added a softplus activation $\log(1 + \exp(\cdot))$ to the final layer to ensure positivity. The expectations with respect to p and \mathcal{T}_p are directly estimated from sampled data. When optimizing τ_θ by stochastic gradient methods, we maintain a copy of the previous network τ_t as the reference network to compute the second and third terms of (15). The gradients of $J(\tau, v)$ with respect to θ and v are given by

$$\begin{aligned} \nabla_\theta J(\tau, v) &= \mathbb{E}_p[\tau \nabla_\theta \tau] - (1 - \alpha_{t+1}) \mathbb{E}_p[\tau_t \nabla_\theta \tau] \\ &\quad - \alpha_{t+1} \mathbb{E}_{\mathcal{T}_p}[\tau_t \nabla_\theta \tau] + v \mathbb{E}_p[\nabla_\theta \tau], \quad (16) \\ \nabla_v J(\tau, v) &= \mathbb{E}_p[\tau] - 1 - \lambda v. \end{aligned}$$

After convergence of τ_θ in each iteration, the reference network is updated by setting $\tau_{t+1} = \tau_\theta$. Note that one may apply other gradient-based optimizers instead of SGD.

3. Convergence Analysis

We now demonstrate that the final algorithm obtains sufficient control over error accumulation to achieve consistency. For notation brevity, we discuss the result for the simpler form (5) instead of the ratio form (6). The argument easily extends to the ratio form.

Starting from the plain stochastic update $\mu_t = \hat{\mathcal{T}} \mu_{t-1}$, the damped update can be expressed by

$$\begin{aligned} \mu_t &= (1 - \alpha_t) \mu_{t-1} + \alpha_t \hat{\mathcal{T}} \mu_{t-1} \\ &= (1 - \alpha_t) \mu_{t-1} + \alpha_t \mathcal{T} \mu_{t-1} + \alpha_t \epsilon, \end{aligned} \quad (17)$$

where ϵ is the error due to stochasticity in $\hat{\mathcal{T}}$. The following theorem establishes the convergence properties of the damped iteration.

Theorem 2 (Informal) *Under mild conditions, after t iteration with step-size $\alpha_t = 1/\sqrt{t}$, we have*

$$\mathbb{E} \left[\|\mu_R - \mathcal{T} \mu_R\|_2^2 \right] \leq \frac{C_1}{\sqrt{t}} \|\mu_0 - \mu\|_2^2 + \frac{C_2 \ln t}{\sqrt{t}} \|\epsilon\|_2^2,$$

for some constants $C_1, C_2 > 0$, where the expectation is taken over the distribution of iterates $(\mu_R)_{R=1}^t$. In other words, $\mathbb{E} \left[\|\mu_R - \mathcal{T} \mu_R\|_2^2 \right] = \tilde{O}(t^{-1/2})$, and consequently μ_R converges to μ for ergodic \mathcal{T} .

The precise version of the theorem statement, together with a complete proof, is given in Appendix B.

Note that the optimization quality depends on the number of samples, the approximation error of the parametric family, and the optimization algorithm. There is a complex trade-off between these factors (Bottou and Bousquet, 2008). On one hand, with more data, the statistical error is reduced, but the computational cost of the optimization increases. On

the other hand, with a more flexible parametrization, such as neural networks, reduces the approximation error, but adds to the difficulty of optimization as the problem might no longer be convex. Alternatively, if the complexity of the parameterized family is increased, the consequences of statistical error also increases.

Representing τ in a reproducing kernel Hilbert space (RKHS) is a particularly interesting case, because the problem (13) becomes convex, hence the optimization error of the empirical surrogate is reduced to zero. Nguyen et al. (2008, Theorem 2) show that, under mild conditions, the statistical error can be bounded in rate $\mathcal{O}\left(n^{-\frac{1}{2+\beta}}\right)$ in terms of Hellinger distance (β denotes the exponent in the bracket entropy of the RKHS), while the approximation error will depend on the RKHS (Bach, 2014).

4. Related Work

The algorithm we have developed reduces distribution estimation to density ratio estimation, which has been extensively studied in numerous contexts. One example is learning under covariate shift (Shimodaira, 2000), where the ratio τ can be estimated by different techniques (Gretton et al., 2009; Nguyen et al., 2008; Sugiyama et al., 2008; Sugiyama and Kawanabe, 2012). These previous works differ from the current setting in that they require data to be sampled from both the target and proposal distributions. By contrast, we consider a substantially more challenging problem, where only data sampled from the proposal is available, and the target distribution is given only *implicitly* by (1) through the transition kernel \mathcal{T} . A more relevant approach is Stein importance sampling (Liu and Lee, 2017), where the ratio is estimated by minimizing the kernelized Stein discrepancy (Liu et al., 2016). However, it requires additional gradient information about the target potential, whereas our method only requires sampled transitions. Moreover, the method of Liu and Lee (2017) is computationally expensive and does not extrapolate to new examples.

The algorithm we develop in this paper is inspired by the classic power method for finding principal eigenvectors. Many existing works have focused on the finite-dimension setting (Balsubramani et al., 2013; Hardt and Price, 2014; Yang et al., 2017), while Kim et al. (2005) and Xie et al. (2015) have extended the power method to the infinite-dimension case using RKHS. Not only do these algorithms require access to the transition kernel \mathcal{T} , but they also require tractable operator multiplications. In contrast, our method avoids direct interaction with the operator \mathcal{T} , and can use flexible parametrizations (such as neural networks) to learn the density ratio without per-step renormalization.

Another important class of methods for estimating or sampling from stationary distributions are based on simula-

tions. A prominent example is Markov chain Monte Carlo (MCMC), which is widely used in many statistical inference scenarios (Andrieu et al., 2003; Koller and Friedman, 2009; Welling and Teh, 2011). Existing MCMC methods (e.g., Neal et al., 2011; Hoffman and Gelman, 2014) require repeated, and often many, interactions with the transition operator \mathcal{T} to acquire a single sample from the stationary distribution. Instead, VPM can be applied when only a fixed sample is available. Interestingly, this suggests that VPM can be used to “post-process” samples generated from typical MCMC methods to possibly make more effective use of the data. We demonstrated this possibility empirically in Section 5. Unlike VPM, other post-processing methods (Oates et al., 2017) require additional information about the target distribution (Robert and Casella, 2004). Recent advances have also shown that learning parametric samplers can be beneficial (Song et al., 2017; Li et al., 2019), but require the potential function. In contrast, VPM directly learns the stationary density ratio solely from transition data.

One important application of VPM is off-policy RL (Precup et al., 2001). In particular, in off-policy evaluation (OPE), one aims to evaluate a target policy’s performance, given data collected from a different behavior policy. This problem matches our proposed framework as the collected data naturally consists of transitions from a Markov chain, and one is interested in estimating quantities computed from the stationary distribution of a different policy. (See Appendix C for a detailed description of how the VPM algorithm can be applied to OPE, even when $\gamma = 1$.) Standard importance weighting is known to have high variance, and various techniques have been proposed to reduce variance (Precup et al., 2001; Jiang and Li, 2016; Rubinstein and Kroese, 2016; Thomas and Brunskill, 2016; Guo et al., 2017). However, these methods still exhibit exponential variance in the trajectory length (Li et al., 2015; Jiang and Li, 2016).

More related to the present paper is the recent work on off-policy RL that avoids the exponential blowup of variance. It is sufficient to adjust observed rewards according to the ratio between the target and behavior stationary distributions (Hallak and Mannor, 2017; Liu et al., 2018; Gelada and Bellemare, 2019). Unfortunately, these methods require knowledge of the behavior policy, $p(a|s)$, in addition to the transition data, which is not always available in practice. In this paper, we focus on the behavior-agnostic scenario where $p(a|s)$ is unknown. Although the recent work of Nachum et al. (2019) considers the same scenario, their approach is only applicable when the discount factor $\gamma < 1$, whereas the method in this paper can handle any $\gamma \in [0, 1]$.

5. Experimental Evaluation

In this section, we demonstrate the advantages of VPM in four representative applications. Due to space limit, experi-

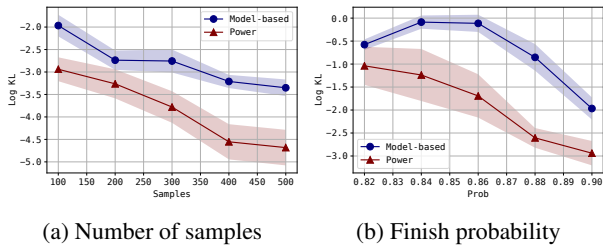


Figure 1. Log KL divergence between estimation and the truth.

ment details are provided in Appendix D.

5.1. Queueing

In this subsection, we use VPM to estimate the stationary distribution of queue length. Following the standard Kendall’s notation in queueing theory (Haviv, 2009; Serfozo, 2009), we analyze the discrete-time Geo/Geo/1 queue, which is commonly used in the literature (Atencia and Moreno, 2004; Li and Tian, 2008; Wang et al., 2014). Here the customer inter-arrival time and service time are geometrically distributed with one service desk. The probe distribution $p(x)$ is a uniform distribution over the states in a predefined range $[0, B)$. The observed transition (x, x') is the length change in one time step. The queue has a closed-form stationary distribution that we can compare to (Serfozo, 2009, Sec.1.11).

Fig. 1 provides the log KL divergence between the estimated and true stationary distributions. We compare VPM to a model-based approach, which estimates the transition matrix $\hat{T}(x'|x)$ from the same set of data, then simulates a long trajectory using \hat{T} . It can be seen that our method can be more effective across different sample sizes and queue configurations.

5.2. Solving SDEs

We next apply VPM to solve a class of SDEs known as the Ornstein-Uhlenbeck process (OUP), which finds many applications in biology (Butler and King, 2004), financial mathematics and physical sciences (Oksendal, 2013). The process is described by the equation:

$$dX = \theta(m - X)dt + \sigma dW$$

where m is the asymptotic mean, $\sigma > 0$ is the deviation, $\theta > 0$ determines the strength, and W is the Wiener process. The OUP has a closed-form solution, which converges to the stationary distribution, a normal distribution $\mathcal{N}(m, \sigma^2/2\theta)$, as $t \rightarrow \infty$. This allows us to conveniently calculate the Maximum Mean Discrepancy (MMD) between the adjusted sample to a true sample. We compare our method with the Euler-Maruyama (EM) method (Gardiner, 2009), which is a standard simulation-based method for solving SDEs. VPM

uses samples from the EM steps to train the ratio network and the learned ratio is used to compute weighted MMD.

The results are shown in Fig. 2, with different configurations of parameters (m, σ, θ) . It can be seen that VPM consistently improves over the EM method in terms of the log MMD to a true sample from the normal distribution. The EM method only uses the most recent data, which can be wasteful since the past data can carry additional information about the system dynamics.

In addition, we perform experiment on real-world phylogeny studies. OUP is widely used to model the evolution of various organism traits. The results of two configurations (Beaulieu et al., 2012; Santana et al., 2012, Tab.3&1 resp.) are shown in Fig. 2d. Notably VPM can improve over the EM method by correcting the sample with learned ratio.

5.3. Post-processing MCMC

In this experiment, we demonstrate how VPM can post-process MCMC to use transition data more effectively in order to learn the target distributions. We use four common potential functions as shown in the first column of Fig. 3 (Neal, 2003; Rezende and Mohamed, 2015; Li et al., 2018). A point is sampled from the uniform distribution $p(x) = \text{Unif}(x; [-6, 6]^2)$, then transitioned through an HMC operator (Neal et al., 2011). The transitioned pairs are used as training set \mathcal{D} .

We compare VPM to a model-based method that explicitly learns a transition model $\hat{T}(x'|x)$, parametrized as a neural network to produce Gaussian mean (with fixed standard deviation of 0.1). Then, we apply \hat{T} to a hold-out set drawn from $p(x)$ sufficiently many times, and use the final instances as limiting samples (second column of Fig. 3). As for VPM, since p is uniform, the estimated $\hat{\tau}$ is proportional to the true stationary distribution. To obtain limiting samples (third column of Fig. 3), we resample from a hold-out set drawn from $p(x)$ with probability proportional to $\hat{\tau}$.

The results are shown in Fig. 3. Note that the model-based method quickly collapses all training data into high-probability regions as stationary distributions, which is an inevitable tendency of restricted parametrized \hat{T} . Our learned ratio faithfully reconstructs the target density as shown in the right-most column of Fig. 3. The resampled data of VPM are much more accurate and diverse than that of the model-based method. These experiments show that VPM can indeed effectively use a fixed set of data to recover the stationary distribution without additional information.

To compare the results quantitatively, Fig. 4 shows the MMD of the estimated sample to a “true” sample. Since there is no easy way to sample from the potential function, the “true” sample consists of data after $2k$ HMC steps with rejection sampler. After each MCMC step, VPM takes the transition

Batch Stationary Distribution Estimation

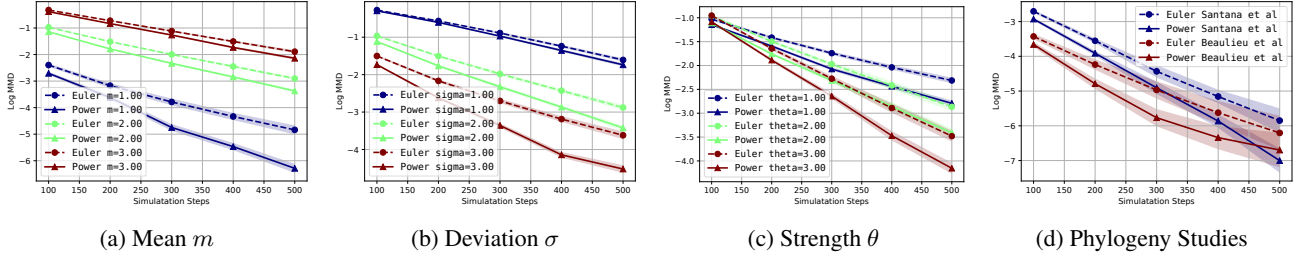


Figure 2. Log MMD versus number of EM steps across different settings, default $(m, \sigma, \theta) = (2, 2, 2)$. (d) is based on the real-world phylogeny studies (Beaulieu et al., 2012; Santana et al., 2012) with $(m, \sigma, \theta) = (0.618, 1.584, 3.85), (0.661, 0.710, 8.837)$ respectively.

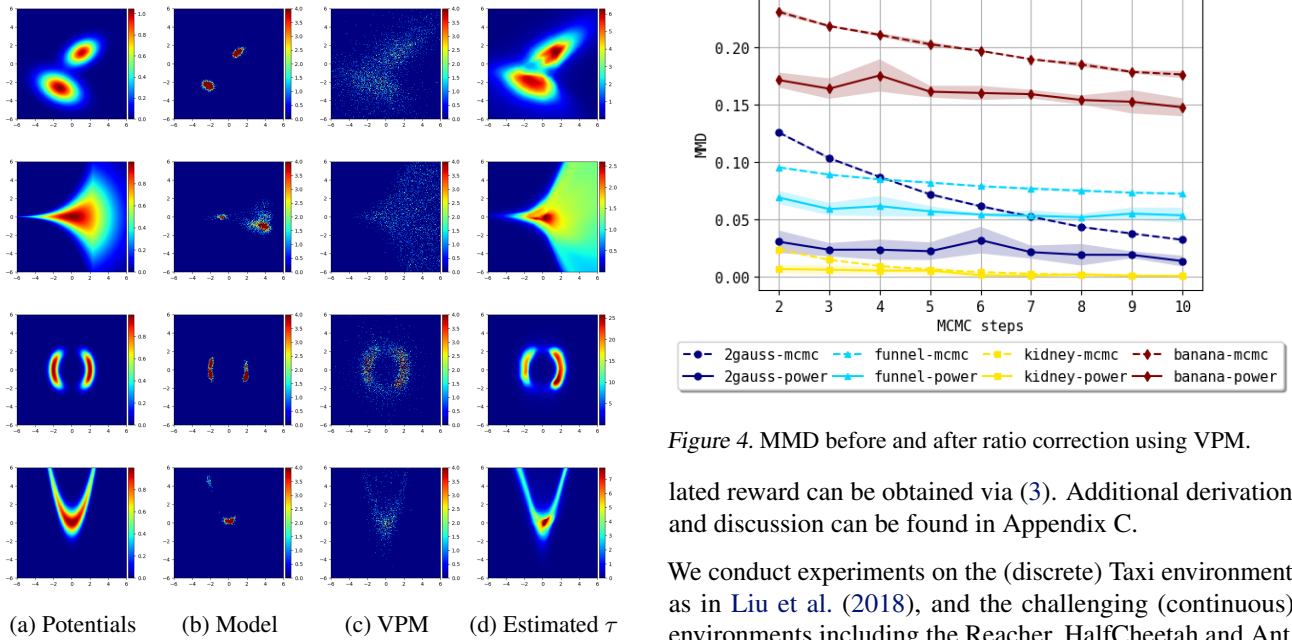


Figure 3. The 2nd and 3rd columns are samples from the model-based method and VPM respectively. Rows (from top to bottom) correspond to data sets: 2gauss, funnel, kidney, banana.

pairs as input and adjusts the sample importance according to the learned ratio. As we can see, after each MCMC step, VPM is able to post-process the data and further reduce MMD by applying the ratio. The improvement is consistent along different MCMC steps across different datasets.

5.4. Off-Policy Evaluation

Finally, we apply our method to behavior-agnostic off-policy evaluation outlined in Section 1, in which only the transition data and the target policy are given, while the behavior policy is *unknown*. Concretely, given a sample $\mathcal{D} = \{(s, a, r, s')_{i=1}^n\}$ from the behavior policy, we compose each transition in \mathcal{D} with a target action $a' \sim \pi(\cdot|s')$. Denoting $x = (s, a)$, the data set can be expressed as $\mathcal{D} = \{(x, x')_{i=1}^n\}$. Applying the proposed VPM with $\mathcal{T}(x'|x)$, we can estimate $\frac{\mu(s,a)}{p(s,a)}$, hence the average accumu-

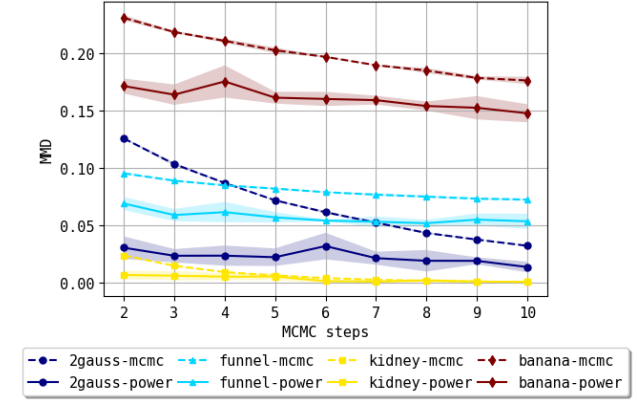


Figure 4. MMD before and after ratio correction using VPM.

lated reward can be obtained via (3). Additional derivation and discussion can be found in Appendix C.

We conduct experiments on the (discrete) Taxi environment as in Liu et al. (2018), and the challenging (continuous) environments including the Reacher, HalfCheetah and Ant.

Taxi is a gridworld environment in which the agent navigates to pick up and drop off passengers in specific locations. The target and behavior policies are set as in Liu et al. (2018). For the continuous environments, the Reacher agent tries to reach a specified location by swinging an robotic arm, while the HalfCheetah/Ant agents are complex robots that try to move forward as much as possible. The target policy is a pre-trained PPO or A2C neural network, which produces a Gaussian action distribution $\mathcal{N}(m_t, \Sigma_t)$. The behavior policy is the same as target policy but using a larger action variance $\Sigma_b = (1 - \alpha)\Sigma_t + 2\alpha\Sigma_t, \alpha \in (0, 1]$. We collect T trajectories of n steps each, using the behavior policy.

We compare VPM to a model-based method that estimates both the transition \mathcal{T} and reward R functions. Using *behavior cloning*, we also compare to the trajectory-wise and step-wise weighted importance sampling (WIST, WISS) (Precup et al., 2001), as well as Liu et al. (2018) with their public code for the Taxi environment.

The results are shown in Fig. 5. The x -axes are different configurations and the y -axes are the log Mean Square

Batch Stationary Distribution Estimation

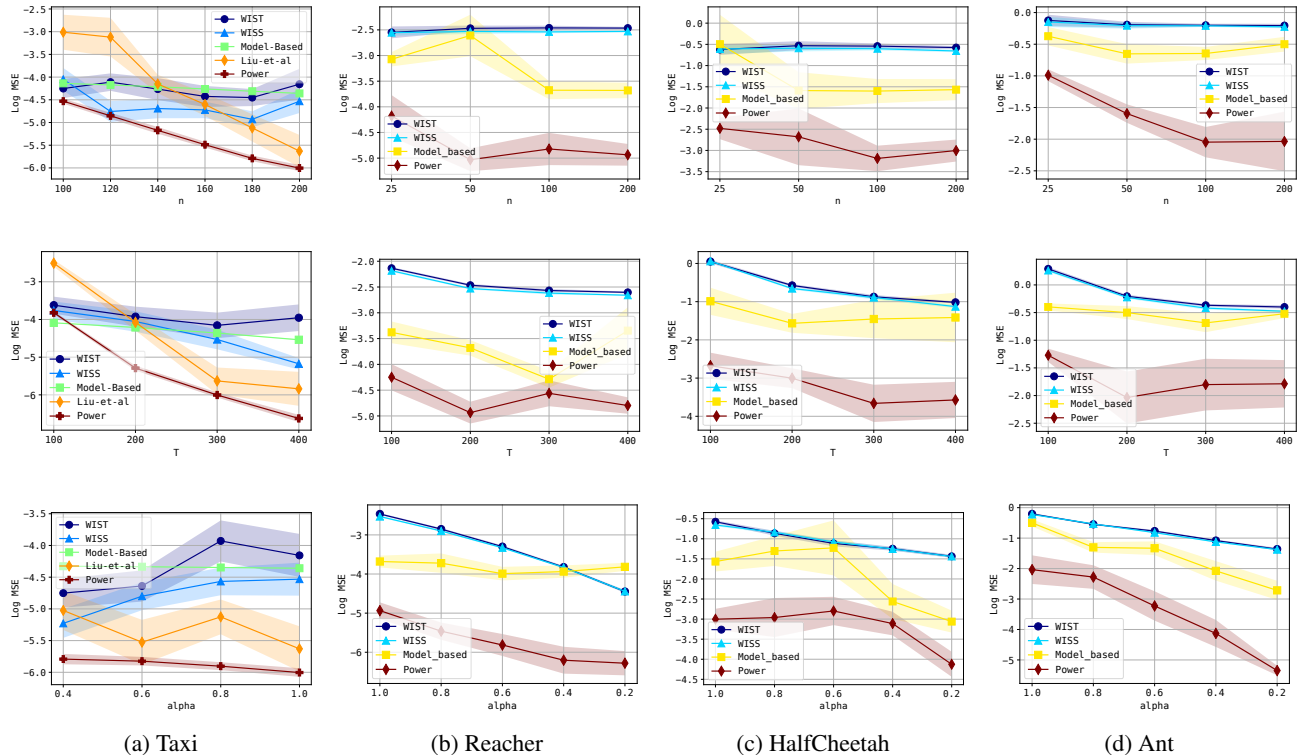


Figure 5. Log MSE of different methods for various datasets and settings.

Error (MSE) to the true average target policy reward, estimated from abundant on-policy data collected from the target policy. As we can see, VPM outperforms all baselines significantly across different settings, including number of trajectories, trajectory length and behavior policies. The method by Liu et al. (2018) can suffer from not knowing the behavior policy, as seen in the Taxi environment. Weighted importance sampling methods (WIST,WISS) also require access to the behavior policy.

5.5. Ablation Study

In this section, we conduct an ablation study to show that VPM is robust to different choices of the parameters. Fig. 6 shows the MMD curves for the MCMC funnel dataset in Section 5.3, using different learning rates, number of inner optimization steps and the regularization λ . Other datasets show similar trends.

Learning rates. In all experiments, we use Adam optimizer (Kingma and Ba, 2014). Fig. 6a shows the convergent behaviour with different learning rates in $\{0.0003, 0.0006, 0.001, 0.003\}$. The algorithm can take a longer time to converge when using a small learning rate. Even though large learning rate (e.g., 0.003) seems to converge faster, its final solution can be noisy. We can see that VPM can work using different learning rates around the default Adam learning rate of 0.001.

Number of inner optimization steps. Recall that in each power iteration, VPM solves an inner optimization Eq. (15). Fig. 6b shows the the effect of different number of inner optimization steps M . Larger M can produce more accurate power iterator and converge faster in terms of number of power iterations, but the time per iteration will also increase accordingly. If M is too small (e.g., 3), the learning can be unstable and the final ratio network can be inaccurate. Due to the damped update, the error in each power iteration can be controlled effectively and VPM can converge to the optimal ratio as long as M is reasonably large.

Regularization. Finally, we investigate the effect of the regularization parameter λ . Intuitively, λ controls the capability of the dual variable v in Eq. (13). The results are shown in Fig. 6c. Although different λ values can have different convergence speeds, their final solutions can achieve low MMD given sufficient iterations, as suggested by Theorem 1.

6. Conclusion

We have formally considered the problem of estimating stationary distribution of an ergodic Markov chain using a fixed set of transition data. We extended a classical power iteration approach to the batch setting, using an equivalent variational reformulation of the update rule to bypass the agnosticity of transition operator and the intractable operations in a functional space, yielding a new algorithm *Variational*

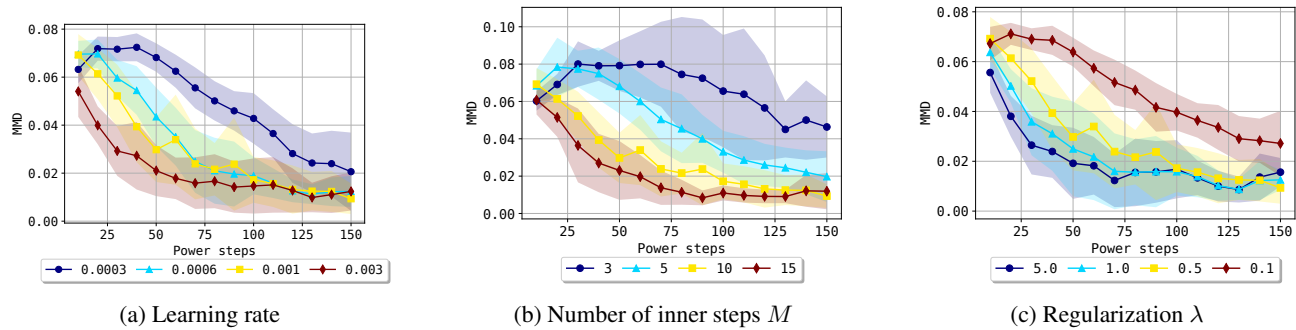


Figure 6. Ablation study. MMD versus number of power iterations for the funnel dataset. Default $(lr, M, \lambda) = (0.001, 10, 0.5)$.

Power Method (VPM). We characterized the convergence of VPM theoretically, and demonstrated its empirical advantages for improving existing methods on several important problems such as queueing, solving SDEs, post-processing MCMC and behavior-agnostic off-policy evaluation.

References

- Christophe Andrieu, Nando de Freitas, Arnaud Doucet, and Michael I. Jordan. An introduction to mcmc for machine learning. *Machine Learning*, 50:5–43, 2003.
- Ivan Atencia and Pilar Moreno. The discrete-time geo/geo/1 queue with negative customers and disasters. *Computers & Operations Research*, 31(9):1537–1548, 2004.
- Francis R. Bach. Breaking the curse of dimensionality with convex neural networks. *CoRR*, abs/1412.8690, 2014.
- Akshay Balsubramani, Sanjoy Dasgupta, and Yoav Freund. The fast convergence of incremental pca. In *Advances in Neural Information Processing Systems*, pages 3174–3182, 2013.
- Jeremy M Beaulieu, Dwueng-Chwuan Jhwueng, Carl Boettiger, and Brian C O’Meara. Modeling stabilizing selection: expanding the ornstein–uhlenbeck model of adaptive evolution. *Evolution: International Journal of Organic Evolution*, 66(8):2369–2383, 2012.
- Léon Bottou and Olivier Bousquet. The tradeoffs of large scale learning. In *Advances in neural information processing systems*, pages 161–168, 2008.
- Marguerite A Butler and Aaron A King. Phylogenetic comparative analysis: a modeling approach for adaptive evolution. *The American Naturalist*, 164(6):683–695, 2004.
- Bo Dai, Niao He, Yunpeng Pan, Byron Boots, and Le Song. Learning from conditional distributions via dual embeddings. In *Artificial Intelligence and Statistics*, pages 1458–1467, 2017.
- Crispin Gardiner. *Stochastic methods*, volume 4. Springer Berlin, 2009.
- Carles Gelada and Marc G Bellemare. Off-policy deep reinforcement learning by bootstrapping the covariate shift. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3647–3655, 2019.
- Arthur Gretton, Alex Smola, Jiayuan Huang, Marcel Schmittfull, Karsten Borgwardt, and Bernhard Schölkopf. Covariate shift by kernel mean matching. *Dataset shift in machine learning*, 3(4):5, 2009.
- Zhaohan Guo, Philip S Thomas, and Emma Brunskill. Using options and covariance testing for long horizon off-policy policy evaluation. In *Advances in Neural Information Processing Systems*, pages 2492–2501, 2017.
- Assaf Hallak and Shie Mannor. Consistent on-line off-policy evaluation. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1372–1383. JMLR. org, 2017.
- Moritz Hardt and Eric Price. The noisy power method: A meta algorithm with applications. In *Advances in Neural Information Processing Systems*, pages 2861–2869, 2014.
- Moshe Haviv. Queues—a course in queueing theory. *The Hebrew University, Jerusalem*, 91905, 2009.
- Matthew D Hoffman and Andrew Gelman. The no-u-turn sampler: adaptively setting path lengths in hamiltonian monte carlo. *Journal of Machine Learning Research*, 15(1):1593–1623, 2014.
- Nan Jiang and Lihong Li. Doubly robust off-policy value evaluation for reinforcement learning. In *International Conference on Machine Learning*, pages 652–661, 2016.
- Leo P Kadanoff. *Statistical physics: statics, dynamics and renormalization*. World Scientific Publishing Company, 2000.

- Kwang In Kim, Matthias O. Franz, and Bernhard Schölkopf. Iterative kernel principal component analysis for image modeling. *IEEE transactions on pattern analysis and machine intelligence*, 27(9):1351–1366, 2005.
- Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.
- David A Levin and Yuval Peres. *Markov chains and mixing times*, volume 107. American Mathematical Soc., 2017.
- Chunyuan Li, Ke Bai, Jianqiao Li, Guoyin Wang, Changyou Chen, and Lawrence Carin. Adversarial learning of a sampler based on an unnormalized distribution. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 3302–3311, 2019.
- Ji-hong Li and Nai-shuo Tian. Analysis of the discrete time geo/geo/1 queue with single working vacation. *Quality Technology & Quantitative Management*, 5(1):77–89, 2008.
- Lihong Li, Remi Munos, and Csaba Szepesvari. Toward minimax off-policy value estimation. In *Artificial Intelligence and Statistics*, pages 608–616, 2015.
- Wenliang Li, Dougal Sutherland, Heiko Strathmann, and Arthur Gretton. Learning deep kernels for exponential family densities. *arXiv preprint arXiv:1811.08357*, 2018.
- Jun S. Liu. *Monte Carlo strategies in scientific computing*. Springer, 2001. ISBN 0387952306.
- Qiang Liu and Jason Lee. Black-box importance sampling. In *Artificial Intelligence and Statistics*, pages 952–961, 2017.
- Qiang Liu, Jason Lee, and Michael Jordan. A kernelized stein discrepancy for goodness-of-fit tests. In *International conference on machine learning*, pages 276–284, 2016.
- Qiang Liu, Lihong Li, Ziyang Tang, and Dengyong Zhou. Breaking the curse of horizon: Infinite-horizon off-policy estimation. In *Advances in Neural Information Processing Systems*, pages 5356–5366, 2018.
- Sean Meyn, Richard L. Tweedie, and Peter W. Glynn. *Markov Chains and Stochastic Stability*. Cambridge Mathematical Library. Cambridge University Press, 2 edition, 2009. doi: 10.1017/CBO9780511626630.
- Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of Machine Learning*. The MIT Press, 2012. ISBN 026201825X, 9780262018258.
- Ofir Nachum, Yinlam Chow, Bo Dai, and Lihong Li. Dualdice: Behavior-agnostic estimation of discounted stationary distribution corrections. *CoRR*, abs/1906.04733, 2019. URL <http://arxiv.org/abs/1906.04733>.
- Radford M Neal. Slice sampling. *Annals of statistics*, pages 705–741, 2003.
- Radford M Neal et al. Mcmc using hamiltonian dynamics. *Handbook of markov chain monte carlo*, 2(11):2, 2011.
- X.L. Nguyen, M. Wainwright, and M. Jordan. Estimating divergence functionals and the likelihood ratio by penalized convex risk minimization. In *Advances in Neural Information Processing Systems 20*, pages 1089–1096. MIT Press, Cambridge, MA, 2008.
- Chris J Oates, Mark Girolami, and Nicolas Chopin. Control functionals for monte carlo integration. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 79(3):695–718, 2017.
- Bernt Oksendal. *Stochastic differential equations: an introduction with applications*. Springer Science & Business Media, 2013.
- Doina Precup, Richard S Sutton, and Sanjoy Dasgupta. Off-policy temporal-difference learning with function approximation. In *ICML*, pages 417–424, 2001.
- Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.
- Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows. *arXiv preprint arXiv:1505.05770*, 2015.
- C. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer, second edition, 2004.
- Reuven Y Rubinstein and Dirk P Kroese. *Simulation and the Monte Carlo method*, volume 10. John Wiley & Sons, 2016.
- Ernest K Ryu and Stephen Boyd. Primer on monotone operator methods. *Appl. Comput. Math*, 15(1):3–43, 2016.
- Sharlene E Santana, Ian R Grosse, and Elizabeth R Dumont. Dietary hardness, loading behavior, and the evolution of skull form in bats. *Evolution: International Journal of Organic Evolution*, 66(8):2587–2598, 2012.
- Richard Serfozo. *Basics of applied stochastic processes*. Springer Science & Business Media, 2009.
- H. Shimodaira. Improving predictive inference under covariance shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90, 2000.

- Jiaming Song, Shengjia Zhao, and Stefano Ermon. A-nice-mc: Adversarial training for mcmc. In *Advances in Neural Information Processing Systems*, pages 5140–5150, 2017.
- Masashi Sugiyama and Motoaki Kawanabe. *Machine learning in non-stationary environments: Introduction to covariate shift adaptation*. MIT press, 2012.
- Masashi Sugiyama, Taiji Suzuki, Shinichi Nakajima, Hisashi Kashima, Paul von Bünau, and Motoaki Kawanabe. Direct importance estimation for covariate shift adaptation. *Annals of the Institute of Statistical Mathematics*, 60(4):699–746, 2008.
- Masashi Sugiyama, Taiji Suzuki, and Takafumi Kanamori. *Density ratio estimation in machine learning*. Cambridge University Press, 2012.
- R.S. Sutton and A.G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 1998.
- Philip Thomas and Emma Brunskill. Data-efficient off-policy policy evaluation for reinforcement learning. In *International Conference on Machine Learning*, pages 2139–2148, 2016.
- Fang Wang, Jinting Wang, and Feng Zhang. Equilibrium customer strategies in the geo/geo/1 queue with single working vacation. *Discrete Dynamics in Nature and Society*, 2014, 2014.
- Yu-Xiang Wang, Alekh Agarwal, and Miroslav Dudik. Optimal and adaptive off-policy evaluation in contextual bandits. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3589–3597. JMLR.org, 2017.
- Max Welling and Yee-Whye Teh. Bayesian learning via stochastic gradient langevin dynamics. In *International Conference on Machine Learning (ICML)*, pages 681–688, 2011.
- Bo Xie, Yingyu Liang, and Le Song. Scale up nonlinear component analysis with doubly stochastic gradients. *arXiv preprint arXiv:1504.03655*, 2015.
- Tengyang Xie, Yifei Ma, and Yu-Xiang Wang. Towards optimal off-policy evaluation for reinforcement learning with marginalized importance sampling. In *Advances in Neural Information Processing Systems 32*, pages 9665–9675, 2019.
- Lin F Yang, Vladimir Braverman, Tuo Zhao, and Mengdi Wang. Online factorization and partition of complex networks from random walks. *arXiv preprint arXiv:1705.07881*, 2017.

Appendix

A. Consistency of the Objectives

Theorem 1 (Normalization of solution) *If $\mathbb{E}_p [\tau_t] = 1$, then for any $\lambda > 0$, the estimator (13) has the same solution as (12), hence $\mathbb{E}_p [\tau_{t+1}] = 1$.*

Proof Taking derivative of the objective function in (12) and setting it to zero, we can see that the unconstrained solution is $\frac{\mathcal{T}_p \tau_t}{p}$. Moreover, it satisfies the constraint when $\mathbb{E}_p [\tau_t] = 1$: we can rewrite $\tau_t = \frac{\mu_t}{p}$ for some distribution μ_t and $\mathbb{E}_p \left[\frac{\mathcal{T}_p \tau_t}{p} \right] = \int \mathcal{T}(x'|x) \mu_t(x) dx dx' = 1$.

We just need to show $\frac{\mathcal{T}_p \tau_t}{p}$ is also the solution to (13). First, note that for any primal τ , the optimal dual v can be attained at $v = \frac{1}{\lambda} (\mathbb{E}_p [\tau] - 1)$. Plugging it to (13), we have for any $\lambda > 0$

$$\begin{aligned} & \min_{\tau \geq 0} \frac{1}{2} \mathbb{E}_{p(x')} [\tau^2(x')] - \mathbb{E}_{\mathcal{T}_p(x, x')} [\tau_t(x) \tau(x')] + \frac{1}{2\lambda} (\mathbb{E}_p [\tau] - 1)^2 \\ & \geq \min_{\tau \geq 0} \frac{1}{2} \mathbb{E}_{p(x')} [\tau^2(x')] - \mathbb{E}_{\mathcal{T}_p(x, x')} [\tau_t(x) \tau(x')] + \min_{\tau \geq 0} \frac{1}{2\lambda} (\mathbb{E}_p [\tau] - 1)^2 \\ & = -\frac{1}{2} \mathbb{E}_p \left[\left(\frac{\mathcal{T}_p \tau_t}{p} \right)^2 \right]. \end{aligned} \quad (18)$$

This lower bound is attainable by plugging $\tau = \frac{\mathcal{T}_p \tau_t}{p}$ in (13). Finally, we conclude the proof by noticing that (13) is strictly convex so the optimal solution is unique. ■

B. Convergence Analysis

Let (X, Σ, ν) be a measure space. The $\mathcal{L}^2(X)$ space consists of measurable functions $f : X \mapsto \mathbb{R}$ such that $\|f\| = (\int |f|^2 d\nu)^{1/2} < \infty$. Suppose the initial $\mu_0 \in \mathcal{L}^2(X)$, we want to show the converging behavior of the following damped iteration:

$$\begin{aligned} \mu_t &= (1 - \alpha_t) \mu_{t-1} + \alpha_t \widehat{\mathcal{T}} \mu_{t-1} \\ &= (1 - \alpha_t) \mu_{t-1} + \alpha_t \mathcal{T} \mu_{t-1} + \alpha_t \epsilon \end{aligned} \quad (19)$$

with suitable step-sizes $\alpha_t \in (0, 1)$, where $\epsilon \in \mathcal{L}^2(X)$ is a random field due to stochasticity in $\widehat{\mathcal{T}}$. To this end, we will use the following lemma.

Lemma 3 *For $\alpha \in \mathbb{R}$, $f, g \in \mathcal{L}^2(X)$*

$$\|(1 - \alpha)f + \alpha g\|^2 = (1 - \alpha)\|f\|^2 + \alpha\|g\|^2 - \alpha(1 - \alpha)\|f - g\|^2.$$

This can be proved by expanding both sides. Now we state our main convergence result.

Theorem 2 *Suppose $\mu_0 \in \mathcal{L}^2(X)$, the step size is $\alpha_t = 1/\sqrt{t}$, $\epsilon \in \mathcal{L}^2(X)$ is a random field and \mathcal{T} has a unique stationary distribution μ . After t iterations, define the probability distribution over the iterations as*

$$\Pr(R = k) = \frac{\alpha_k(1 - \alpha_k)}{\sum_{k'=1}^t \alpha_{k'}(1 - \alpha_{k'})}$$

Then there exist some constants $C_1, C_2 > 0$ such that

$$\mathbb{E} \left[\|\mu_R - \mathcal{T} \mu_R\|_2^2 \right] \leq \frac{C_1}{\sqrt{t}} \|\mu_0 - \mu\|_2^2 + \frac{C_2 \ln t}{\sqrt{t}} \|\epsilon\|_2^2,$$

where the expectation is taken over R . Consequently, μ_R converges to μ for ergodic \mathcal{T} .

Proof Using Lemma 3 and the fact that \mathcal{T} is non-expansive, we have

$$\begin{aligned} \|\mu_t - \mu\|^2 &= \|(1 - \alpha_t)(\mu_{t-1} - \mu) + \alpha_t(\mathcal{T}\mu_{t-1} - \mu) + \alpha_t\epsilon\|^2 \\ &\leq \|(1 - \alpha_t)(\mu_{t-1} - \mu) + \alpha_t(\mathcal{T}\mu_{t-1} - \mu)\|^2 + \alpha_t^2\|\epsilon\|^2 \\ &\leq (1 - \alpha_t)\|\mu_{t-1} - \mu\|^2 + \alpha_t\|\mathcal{T}\mu_{t-1} - \mu\|^2 - \alpha_t(1 - \alpha_t)\|\mu_{t-1} - \mathcal{T}\mu_{t-1}\|^2 + \alpha_t^2\|\epsilon\|^2 \\ &\leq \|\mu_{t-1} - \mu\|^2 - \alpha_t(1 - \alpha_t)\|\mu_{t-1} - \mathcal{T}\mu_{t-1}\|^2 + \alpha_t^2\|\epsilon\|^2. \end{aligned}$$

Then telescoping sum gives

$$0 \leq \|\mu_t - \mu\|^2 \leq \|\mu_0 - \mu\|^2 + \sum_{k=1}^t \alpha_k^2 \|\epsilon\|^2 - \sum_{k=1}^t \alpha_k(1 - \alpha_k) \|\mu_k - \mathcal{T}\mu_k\|^2$$

So

$$\sum_{k=1}^t \alpha_k(1 - \alpha_k) \|\mu_k - \mathcal{T}\mu_k\|^2 \leq \|\mu_0 - \mu\|^2 + \sum_{k=1}^t \alpha_k^2 \|\epsilon\|^2.$$

Divide both sides by $\sum_{k=1}^t \alpha_k(1 - \alpha_k)$ (taking expectation over iterations) gives

$$\mathbb{E}[\|\mu_R - \mathcal{T}\mu_R\|^2] = \frac{\sum_{k=1}^t \alpha_k(1 - \alpha_k)}{\sum_{k=1}^t \alpha_{k'}(1 - \alpha_{k'})} \|\mu_k - \mathcal{T}\mu_k\|^2 \leq \frac{\|\mu_0 - \mu\|^2 + \sum_{k=1}^t \alpha_k^2 \|\epsilon\|^2}{\sum_{k=1}^t \alpha_k(1 - \alpha_k)}.$$

When $\alpha_t = 1/\sqrt{t}$, we have

$$\begin{aligned} \sum_{k=1}^t \alpha_k^2 \|\epsilon\|^2 &= \sum_{k=1}^t \frac{1}{k} \|\epsilon\|^2 \leq (\ln t + 1) \|\epsilon\|^2 \\ \sum_{k=4}^t \alpha_k(1 - \alpha_k) &= \sum_{k=4}^t \frac{1}{\sqrt{k}} - \frac{1}{k} \geq \int_4^t \left(\frac{1}{\sqrt{k+1}} - \frac{1}{k+1} \right) dk = \Omega\left(t^{\frac{1}{2}}\right) \end{aligned}$$

So for big enough t , there exists $C_0 > 0$ such that

$$\mathbb{E}[\|\mu_R - \mathcal{T}\mu_R\|^2] \leq \frac{\|\mu_0 - \mu\|^2 + \ln(t+1)\|\epsilon\|^2}{C_0\sqrt{t}},$$

which leads to the the bound in the theorem and $\mathbb{E}[\|\mu_R - \mathcal{T}\mu_R\|^2] = \tilde{\mathcal{O}}(t^{-1/2})$. Additionally, since \mathcal{T} has a unique stationary distribution $\mu = \mathcal{T}\mu$, we have μ_R converges to μ . \blacksquare

C. Application to Off-policy Stationary Ratio Estimation

We provide additional details describing how the variational power method we have developed in the main body of the paper can be applied to the behavior-agnostic off-policy estimation problem (OPE). The general framework has been introduced in Section 1 and the implementation for the undiscounted case ($\gamma = 1$) is demonstrated in Section 5.4. Specifically, given a sample $\mathcal{D} = \{(s, a, r, s')_{i=1}^n\}$ from the behavior policy, we compose each transition in \mathcal{D} with a target action $a' \sim \pi(\cdot|s')$. Denoting $x = (s, a)$, the data set can be expressed as $\mathcal{D} = \{(x, x')_{i=1}^n\}$. Applying the proposed VPM with $\mathcal{T}(x'|x)$, we can estimate $\frac{\mu(s, a)}{p(s, a)}$. Here the $\mu(s, a) = d_\pi(s)\pi(a|s)$ consists of the stationary state occupancy d_π and the target policy π , while $p(s, a)$ is the data-collecting distribution. Then the average accumulated reward can be obtained via (3).

Here we elaborate on how the discounted case (*i.e.*, $\gamma \in (0, 1)$) can be handled by our method. We first introduce essential quantities similar to the undiscounted setting. For a trajectory generated stochastically using policy π from an initial state s_0 : $(s_0, a_0, r_0, s_1, a_1, r_1, \dots)$, where $a_t \sim \pi(\cdot|s_t)$, $s_{t+1} \sim P(\cdot|s_t, a_t)$ and $r_t \sim R(s_t, a_t)$, the the policy value is

$$\rho_\gamma(\pi) := (1 - \gamma) \mathbb{E}_{s_0 \sim \mu_0, a \sim \pi, s' \sim P} [\sum_{t=0}^{\infty} \gamma^t r_t],$$

where μ_0 is the initial-state distribution. Denote

$$d_t^\pi(s, a) = \mathbb{P} \left(s_t = s, a_t = a \left| \begin{array}{l} s_0 \sim \mu_0, \forall i < t, \\ a_i \sim \pi(\cdot | s_i), \\ s_{i+1} \sim P(\cdot | s_i, a_i) \end{array} \right. \right).$$

The discounted occupancy distribution is

$$\mu_\gamma(s, a) := (1 - \gamma) \sum_{t=0}^{\infty} \gamma^t d_t^\pi(s, a). \quad (20)$$

Then, we can re-express the discounted accumulated reward via μ_γ and the stationary density ratio,

$$\rho_\gamma(\pi) = \mathbb{E}_{(s,a) \sim \mu_\gamma(s,a)} [r(s, a)] = \mathbb{E}_{(s,a) \sim p(s,a)} \left[\frac{\mu_\gamma(s, a)}{p(s, a)} r(s, a) \right]. \quad (21)$$

The proposed VPM is applicable to estimating the density ratio in this discounted case. Denoting $x = (s, a)$, $x' = (s', a')$ respectively for notational consistency, we expand μ_γ and use the definition of d_t^π :

$$\begin{aligned} \mu_\gamma(s', a') &= (1 - \gamma) \mu_0(s') \pi(a' | s') + \gamma \int \pi(a' | s') P(s' | s, a) \mu_\gamma(s, a) ds da \\ \implies p(x') \tau^*(x') &= (1 - \gamma) \mu_0 \pi(x') + \gamma \int \mathcal{T}_p(x, x') \tau^*(x) dx, \end{aligned} \quad (22)$$

where $\mu_0 \pi(x') = \mu_0(s') \pi(a' | s')$ and $\mathcal{T}_p(x, x') = \pi(a' | s') P(s' | s, a) p(s, a)$.

It has been shown that the RHS of (22) is contractive (Sutton and Barto, 1998; Mohri et al., 2012), therefore, the fix-point iteration,

$$p(x') \tau_{t+1}(x') = (1 - \gamma) \mu_0 \pi(x') + \gamma \int \mathcal{T}_p(x, x') \tau_t(x) dx, \quad (23)$$

converges to the true τ as $t \rightarrow \infty$, provided the update above is carried out exactly. Compared to (6), we can see that the RHS of (23) is now a mixture of $\mu_0 \pi$ and \mathcal{T}_p , with respective coefficients $(1 - \gamma)$ and γ .

Similarly, we construct the $(t + 1)$ -step variational update as

$$\tau_{t+1} = \arg \min_{\tau \geq 0} \frac{1}{2} \mathbb{E}_{p(x')} [\tau^2(x')] - \gamma \mathbb{E}_{\mathcal{T}_p(x, x')} [\tau_t(x) \tau(x')] - (1 - \gamma) \mathbb{E}_{\mu_0 p(x')} [\tau(x')] + \lambda (\mathbb{E}_p[\tau] - 1)^2. \quad (24)$$

Compared to (11), we see that the main difference is the third term of (24) involves the initial distribution. As $\gamma \rightarrow 1$, (24) reduces to (11).

D. Experiment Details

Here we provide additional details about the experiments. In all experiments, the regularization $\lambda = 0.5$ and the optimizer is Adam with $\beta_1 = 0.5$. The τ model is a neural network with 2 hidden layers of 64 units each with ReLU activation and softplus activation for the output.

D.1. Queueing

For Geo/Geo/1 queue, when the arrival and finish probabilities are $q_a, q_f \in (0, 1)$ respectively with $q_f > q_a$, the stationary distribution is $P(X = i) = (1 - \rho) \rho^i$ where $\rho = q_a(1 - q_f) / [q_f(1 - q_a)]$ (Serfozo, 2009, Sec.1.11). The defaults are $(n, q_a, q_f) = (100, 0.8, 0.9)$ for the figures. ρ is called *traffic intensity* in the queueing literature and we set $B = \lceil 40\rho \rceil$ in the experiment. The mean and standard error of the log KL divergence is computed based on 10 runs. We conduct closed-form update for 1000 steps. As for the model-based method, we simulate the transition chain for 200 steps to attain the estimated stationary distribution.

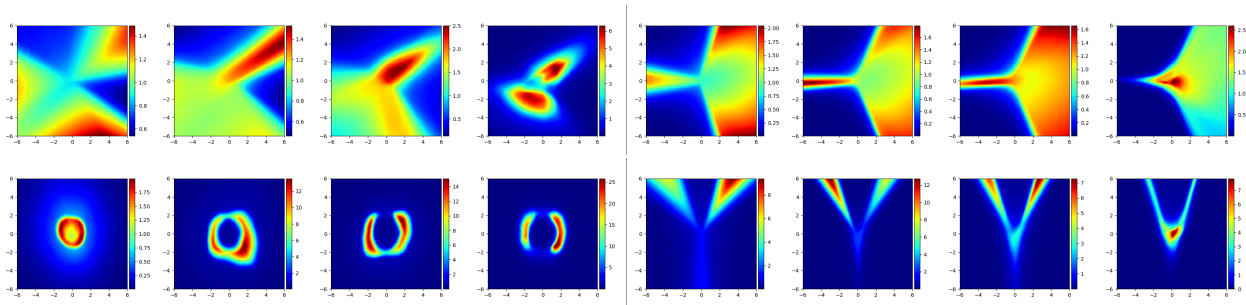


Figure 7. The VPM estimates after $\{10, 20, 30, 150\}$ iterations on the datasets. As we can see, with the algorithm proceeds, the learned stationary density ratio is getting closer to the ground-truth.

D.2. Solving SDEs

Using initial samples are uniformly spaced in $[0, 1]$, we run the Euler-Maruyama (EM) method and evaluate the MMD along the path. The τ model is a neural network with 2 hidden layers of 64 units each with ReLU and Softplus for the final layer. Numbers of outer and inner steps are $T = 50$, $M = 10$. The learning rate is 0.0005. At each evaluation time step t , we use the most recent 1% of evolution data to train our model τ . The plots are reporting the mean and standard deviation over 10 runs. For the phylogeny studies, the number of particles is $1k$ and $dt = 0.0005$ for the EM simulation, while the rest settings using $dt = 0.001$.

D.3. Post-processing MCMC

The potential functions are collected from several open-source projects¹². $50k$ examples are sampled from the uniform distribution $p(x) = \text{Unif}(x; [-6, 6]^2)$, then transition each x through an HMC operator (one leapfrog step of size 0.5). The model-based $\hat{\mathcal{T}}$ has a similar structure as τ except the final layer has 2D output without activation to estimate the Gaussian mean. The mini-batch size is $B = 1k$, the maximum number of power iterations $T = 150$ and the number of inner optimization steps is $M = 10$. The model-based \mathcal{T} is given the same number of iterations ($MT = 1500$). The learning rate is 0.001 for τ and 0.0005 for $\hat{\mathcal{T}}$. To compute the model-based sample, we apply the estimated transition 100 time steps. The MMD plot is based on a “true sample” of size $2k$ from the stationary distribution (estimated by $2k$ HMC transition steps). The numbers are mean and standard deviation over 10 runs. The MMD is computed by the Gaussian kernel with the median pairwise distance as kernel width.

The quality of the transition kernel and the generated data is critical. Since x and x' are supposed to be related, we use an HMC kernel with one leap-frog step. The initial x is effectively forgotten if using too many leap-frog steps. The main point is to show that our method can utilize the intermediate samples from the chain other than the final point. Moreover, to conform with Assumption 2, the potential functions are numerically truncated.

To verify the convergent behavior of our method, Fig. 7 shows how the ratio network improves as we train the model. It can be seen that the our method quickly concentrates its mass to the region with high potentials.

D.4. Off-policy Evaluation

Taxi is a 5×5 gridworld in which the taxi agent navigates to pick up and drop off passengers in specific locations. It has a total of 2000 states and 6 actions. Each step incurs a -1 reward unless the agent picks up or drops off a passenger in the correct locations. The behavior policy is set to be the policy after 950 Q-learning iterations and the target policy is the policy after 1000 iterations. In the Taxi experiment, given a transition (s, a, s') , instead of sampling one single action from the target policy $\pi(a'|s')$, we use the whole distribution $\pi(\cdot|s')$ for estimation. We conduct closed-form update in the power method and the number of steps is $T = 100$.

Continuous experiments. The environments are using the open-source PyBullet engine. The state spaces are in $\mathbb{R}^9, \mathbb{R}^{26}, \mathbb{R}^{28}$ respectively and the action spaces are in $\mathbb{R}^2, \mathbb{R}^6, \mathbb{R}^8$ respectively. the τ model is the same as in the SDE

¹https://github.com/kamenblznashki/normalizing_flows

²<https://github.com/kevin-w-li/deep-kexpfam>

Batch Stationary Distribution Estimation

experiment (except for input, which depends on the environment). $T = 200$, $M = 10$, $B = 1k$ and the learning rate is 0.0003. The model-based method has a similar neural network structure and is trained for $MT = 2k$ steps with a learning rate of 0.0005. The target policy for the Reacher agent is pretrained using PPO while the HalfCheetah and Ant agents are pretrained using A2C (all with two hidden layers of 64 units each).

The results in the plots are mean and standard deviation from 10 runs.