Batch Stationary Distribution Estimation

Junfeng Wen *1  Bo Dai *2  Lihong Li 2  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 queuing, 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 $X$, transition kernel $T$, and a unique stationary distribution $\mu$ that satisfies

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

Assume we are given a fixed sample of state transitions, $D = \{(x, x')_{i=1}^n\} \sim T(x'|x) p(x)$, 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, $x'|x \sim T(x'|x)$. Below we investigate procedures for estimating the point-wise ratios, $\tilde{\mu}(x_i) \approx \frac{\mu(x_i)}{p(x_i)}$, such that the weighted empirical distribution

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

can be used to approximate $\mu$ 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 \( \mu \). 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 \( T \), 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

\[
dX = f(X) \, dt + \sigma(X) \, dW,
\]

where \( f(X) \) is a drift term, \( \sigma(X) \) a diffusion term, and \( W \) the Wiener process. Given data \( D = \{ (x, x') \}_{i=1}^{p} \) 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 \( \mu \) 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 \( M = (S, \mathcal{A}, P, R) \), such that \( S \) 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 \( \pi \) that maps \( s \in S \) to a distribution over \( \mathcal{A} \), a random trajectory can be generated starting from an initial state \( s_0: (s_0, a_0, r_0, s_1, a_1, r_1, \ldots) \), 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 \( \pi \) is defined to be its long-term average per-step reward:

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

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

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

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 \( T \) and \( \mu \) in (1). We make the following assumption about \( T \) and \( \mu \) throughout the paper.

**Assumption 1 (ergodicity)** The transition operator \( T \) has a unique stationary distribution, denoted \( \mu \).

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 \( \mu \) 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

\[
\mu(x') = \int T(x'|x) \mu(x) \, dx = \int T(x'|x) p(x) \frac{\mu(x)}{p(x)} \, dx
\]

\[
:= \int T_p(x, x') \tau(x) \, dx,
\]

which implies

\[
p(x') \tau(x') = \int T_p(x, x') \tau(x) \, dx \quad \text{which implies}
\]

\[
p(x') \tau(x') = \int T_p(x, x') \tau(x) \, dx \quad := \int T_p(x, x') \quad (4)
\]

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

### 2.1. Variational Power Iteration

To develop a practical algorithm for recovering \( \tau \) from the constraint (4), in function space, we first consider the classical power method for recovering the \( \mu \) that satisfies (1).
Batch Stationary Distribution Estimation

From (1) it can be seen that the stationary distribution \( \mu \) is an eigenfunction of \( T \). Moreover, it is the principal eigenfunction, corresponding to the largest eigenvalue \( \lambda_1 = 1 \).

In the simpler case of finite \( X \), the vector \( \mu \) is the principal (right) eigenvector of the transposed transition matrix. A standard approach to computing \( \mu \) is then the power method:

\[
\mu_{t+1} = T\mu_t,
\]

where the division is element-wise. Clearly the fixed point whose iterates converge to \( \mu \) at a rate linear in \( |\lambda_2| \), where \( \lambda_2 \) is the second largest eigenvalue of \( T \). 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 \( X \) to be finite. This can be naturally achieved by the update

\[
\tau_{t+1} = \frac{T_p\tau_t}{p},
\]

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 \( \mu_t \) in (5), \( \tau_t \) in (6) also converges to \( \tau \) at a linear rate for finite \( X \). 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 \( T_p \), but only samples from \( T_p \). Second, the operator \( T_p \) is applied to a function \( \tau_t \), which typically involves an intractable integral over \( X \) 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 \( \phi \) be a lower semicontinuous, convex function satisfying \( \phi(1) = 0 \), and consider the induced \( f \)-divergence,

\[
D_\phi(\tilde{p}||\tilde{q}) = \int \tilde{p}(x) \phi \left( \frac{\tilde{q}(x)}{\tilde{p}(x)} \right) dx
\]

\[
= -\left( \min_{\nu} E_p[\phi^*(\nu)] - E_q[\nu] \right),
\]

where \( \phi^*(x) = \sup_{y \in \mathbb{R}} x^T y - \phi(y) \) is the conjugate function of \( \phi \). The key property of this formulation is that for any distributions \( \tilde{p} \) and \( \tilde{q} \), the inner optimum in \( \nu \) 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:

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

\[
- \mathbb{E}_{T_p(x,x')} \left[ \partial \phi^*(\nu_t(x)) \cdot \nu(x') \right]
\]

\[
= \arg\min_{\nu} \mathbb{E}_{p(x')} \left[ \phi^*(\nu(x')) \right] \]

\[
- \mathbb{E}_{T_p(x,x')} \nu(x) \left[ \nu(x') \right],
\]

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

\[
\partial \phi^*(\nu_{t+1}) = \frac{T_p\tau_t}{p} = \tau_{t+1},
\]

hence the updated ratio \( \tau_{t+1} \) in (6) can be directly recovered from the dual solution \( \nu_{t+1} \), 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

\[
\tau_{t+1} = \arg\min_{\tau \geq 0} \frac{1}{2} \mathbb{E}_{p(x')} \left[ \tau^2(x') \right] - \mathbb{E}_{T_p(x,x')} \left[ \tau(x) \tau(x') \right].
\]

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

We now discuss some practical refinements of the approach.

### 2.2. Normalization

For \( \tau_t \) to be a proper ratio \( \frac{\mu}{\tau_t} \), 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):

\[
\min_{\tau \geq 0} \frac{1}{2} \mathbb{E}_{p(x')} \left[ \tau^2(x') \right] - \mathbb{E}_{T_p(x,x')} \left[ \tau(x) \tau(x') \right],
\]

\[
s.t. \quad \mathbb{E}_p[\tau(x)] = 1.
\]

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

\[
\min_{\tau \geq 0} \max_{v \in \mathbb{R}} J(\tau, v) = \frac{1}{2} \mathbb{E}_{p(x')} \left[ \tau^2(x') \right]
\]

\[
- \mathbb{E}_{T_p(x,x')} \left[ \tau(x) \tau(x') \right] + v \left( \mathbb{E}_p[\tau] - 1 \right) - \frac{\lambda}{2} v^2.
\]

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 \( (\tau, v) \) 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 \( \mathbb{E}_p[\tau_{t+1}] = 1 \).
To control the error due to sampling, we introduce a damped version of the update (solves 2.4. A Practical Algorithm to change the sample from minor modifications to the previous objective. We only need controlled by the stepsize \( \alpha_t \), number of power steps \( T \), number of inner optimization steps \( M \), batch size \( B \).

1. **Input:** Transition data \( D = \{(x, x')_i^d\} \), learning rate \( \alpha_0, \alpha_v \), number of power steps \( T \), number of inner optimization steps \( M \), batch size \( B \)
2. **Initialize** \( \tau_0 \)
3. **for** \( t = 1 \ldots T \) **do**
4. **Update and fix the reference network** \( \tau_t = \tau_0 \)
5. **for** \( m = 1 \ldots M \) **do**
6. **Sample transition data** \( \{(x, x')_i^d\} \)
7. **Compute gradients** \( \nabla_{\theta} J \) and \( \nabla_{v} J \) from (16)
8. **\( \theta = \theta - \alpha_0 \nabla_{\theta} J \) \( >> \) gradient descent
9. **\( v = v + \alpha_v \nabla_{v} J \) \( >> \) gradient ascent
10. **end for**
11. **end for**
12. **Return** \( \tau_0 \)

Hence, we can begin with any \( \tau_0 \) satisfying \( E_p[\tau_0] = 1 \) (e.g., \( \tau_0 = \arg \min_r (E_p[\tau] - 1)^2 \)), and the theorem ensures that the normalization of \( \tau_{t+1} \) 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 \( T_p \) by a stochastic estimate \( \hat{T}_p \) such that \( E[\hat{T}_p] = T_p \). 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 \( \tau_{t+1} = \hat{T}_p \tau_t \), we instead perform a damped update given by

\[
\tau_{t+1} = (1 - \alpha_t + 1) \cdot \tau_t + \alpha_t \cdot \hat{T}_p \tau_t
\]

(14)

where \( \alpha_t \in (0, 1) \) is a stepsize parameter. Intuitively, the update error introduced by the stochasticity of \( \hat{T}_p \) is now controlled by the stepsize \( \alpha_t \). 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 \( T_p \) in (13) by a weighted sample:

\[
\min_{\tau \geq 0} \max_{v \in \mathbb{R}} J(\tau, v) = \frac{1}{2} E_p(x') \left[ \tau^2(x') \right] - (1 - \alpha_t + 1) E_p(x') \left[ \tau_t(x') \tau(x') \right] - \alpha_{t+1} E_{T_p}(x', x') \left[ \tau_t(x') \tau(x') \right] + v (E_p[\tau] - 1) - \frac{\lambda}{2} v^2.
\]

(15)

2.4. A Practical Algorithm

A practical version of VPM is described in Algorithm 1. It solves (15) using a parameterized \( \tau : X' \mapsto \mathbb{R} \) expressed as a neural network \( \tau_\theta \) with parameters \( \theta \). Given the constraint \( T \geq 0 \), we added a softplus activation \( \log(1 + \exp(\cdot)) \) to the final layer to ensure positivity. The expectations with respect to \( p \) and \( T_p \) are directly estimated from sampled data. When optimizing \( \tau_\theta \) by stochastic gradient methods, we maintain a copy of the previous network \( \tau_t \) as the reference network to compute the second and third terms of (15). The gradients of \( J(\tau, v) \) with respect to \( \theta \) and \( v \) are given by

\[
\nabla_\theta J(\tau, v) = E_p [\tau \nabla_\theta \tau] - (1 - \alpha_{t+1}) E_p [\tau_t \nabla_\theta \tau] - \alpha_{t+1} E_{T_p} [\tau_t \nabla_\theta \tau] + v E_p [\nabla_\theta \tau],
\]

(16)

\[
\nabla_v J(\tau, v) = E_p [\tau] - 1 - \lambda v.
\]

After convergence of \( \tau_\theta \) 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 \( \mu_t = \hat{T} \mu_{t-1} \), the damped update can be expressed by

\[
\mu_t = (1 - \alpha_t) \mu_{t-1} + \alpha_t \hat{T} \mu_{t-1} = (1 - \alpha_t + 1) \mu_{t-1} + \alpha_t \epsilon,
\]

(17)

where \( \epsilon \) is the error due to stochasticity in \( \hat{T} \). 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

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

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

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 $\tau$ 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 $\mathcal{O}\left(n^{-\frac{1}{1+\beta}}\right)$ in terms of Hellinger distance ($\beta$ 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 $\tau$ 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 simulations. 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-
We next apply VPM to solve a class of SDEs known as the Ornstein-Uhlenbeck process (OUP), which finds many applications in biology and neural networks. 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) 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 \( p(x) = \text{Unif}([-6, 6]^2] \), then transitioned through an HMC operator (Neal et al., 2011). The transitioned pairs are used as training set \( D \).

We compare VPM to a model-based method that explicitly learns a transition model \( \hat{\mathcal{F}}(x' | x) \), parametrized as a neural network to produce Gaussian mean (with fixed standard deviation of 0.1). Then, we apply \( \hat{\mathcal{F}} \) 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 \( \hat{\mathcal{F}} \) 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 \( \hat{\mathcal{F}} \).

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 \( \hat{\mathcal{F}} \). 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 20k HMC steps with rejection sampler. After each MCMC step, VPM takes the transition...
VPM is able to post-process the data and further reduce MMD before and after ratio correction using VPM. As we can see, after each MCMC step, the data set can be expressed as $\mathcal{D} = \{ (s, a, r, s') \}_{i=1}^{n}$ from the behavior policy, we compose each transition in $\mathcal{D}$ with a target action $a' \sim \pi (-s')$. Denoting $x = (s, a)$, the data set can be expressed as $\mathcal{D} = \{ (x, x') \}_{i=1}^{n}$. Applying the proposed VPM with $T(x'|x)$, we can estimate $\frac{\mu(x, a, r)}{p(x, a)}$, hence the average accumulated 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 $T$ and reward $R$ functions. Using behavior cloning, we also compare to the trajectory-wise and step-wise 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 Error (MSE) of the MMD.
VPM is robust to different choices of the parameters. Even though large learning rate (e.g., 0.003) seems to converge faster, its final solution can be noisy. We can see that 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 effect of different number of inner optimization steps $M$. Larger $M$ can produce more accurate power iteration 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 $\lambda$. Intuitively, $\lambda$ controls the capability of the dual variable $v$ in Eq. (13). The results are shown in Fig. 6c. Although different $\lambda$ 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 agnosticism of transition operator and the intractable operations in a functional space, yielding a new algorithm Variational
**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**


Batch Stationary Distribution Estimation


