Unbiased Gradient Estimation for Variational Auto-Encoders using Coupled Markov Chains

Francisco J. R. Ruiz\(^1\)  \hspace{1cm}  \text{Michalis K. Titsias}\(^1\)  \hspace{1cm}  \text{Taylan Cemgil}\(^1\)  \hspace{1cm}  \text{Arnaud Doucet}\(^1\)  \\
\(^1\)DeepMind

Abstract

The variational auto-encoder (VAE) is a deep latent variable model that has two neural networks in an autoencoder-like architecture; one of them parameterizes the model’s likelihood. Fitting its parameters via maximum likelihood (ML) is challenging since the computation of the marginal likelihood involves an intractable integral over the latent space; thus the VAE is trained instead by maximizing a variational lower bound. Here, we develop a ML training scheme for VAEs by introducing unbiased estimators of the log-likelihood gradient. We obtain the estimators by augmenting the latent space with a set of importance samples, similarly to the importance weighted auto-encoder (IWAE), and then constructing a Markov chain Monte Carlo coupling procedure on this augmented space. We provide the conditions under which the estimators can be computed in finite time and with finite variance. We show experimentally that VAEs fitted with unbiased estimators exhibit better predictive performance.

1 INTRODUCTION

The variational auto-encoder (VAE) [Kingma and Welling, 2014] is a deep latent variable model that uses a joint distribution \( p_\theta(x, z) \), parameterized by \( \theta \), over an observation \( x \) and the corresponding latent variable \( z \). The marginal log-likelihood involves an integral over the latent space,

\[
\mathcal{L}(\theta) := \log p_\theta(x) = \log \left( \int p_\theta(x, z)dz \right).  \tag{1}
\]

As for any other latent variable model, fitting the VAE requires finding the parameters \( \theta \) that best describe the observations. One (intractable) way to find \( \theta \) would be via maximum likelihood, for which the gradient of Eq. 1 is required. Using Fisher’s identity, this gradient can be written as an expectation w.r.t. the posterior \( p_\theta(z \mid x) \),

\[
\nabla_\theta \mathcal{L}(\theta) = \mathbb{E}_{p_\theta(z \mid x)} \left[ \nabla_\theta \log p_\theta(x, z) \right]. \tag{2}
\]

The gradient in Eq. 2 could be approximated unbiasedly if we had access to samples from \( p_\theta(z \mid x) \); however, the posterior is intractable. Although we could use Markov chain Monte Carlo (MCMC) to sample approximately from it [Hoffman, 2017, Naesseth et al., 2020], this would provide a biased estimate, and the bias is difficult to quantify.

Instead, VAEs introduce an encoder \( q_\phi(z \mid x) \) and learn the parameters \( \theta \) by maximizing a variational evidence lower bound (ELBO) [Wainwright and Jordan, 2008, Blei et al., 2017], for which unbiased gradients are readily available [Kingma and Welling, 2014]. Burda et al. [2016] form such a bound using a set of importance samples using the encoder as proposal distribution, leading to the so-called importance weighted auto-encoder (IWAE). The standard ELBO in variational inference can be thought of as a particular instance of the IWAE bound with one importance sample, where the importance distribution is given by the encoder. However, it remains difficult to quantify the difference between the true log-likelihood and the corresponding bound.

We develop here unbiased gradient estimators of the log-likelihood for VAEs by exploiting the coupling estimators developed by Jacob et al. [2020b]. Coupling estimators allow us to obtain unbiased estimates of expectations w.r.t. an intractable target distribution by running two coupled MCMC chains for a finite number of iterations. This approach does not require that the MCMC chains converge to the target, so the unbiased estimator can be computed in a finite (but random) time. However, MCMC couplings are a generic methodology that is not readily applicable to VAEs, since it requires an MCMC kernel that mixes well and provides a suitable coupling mechanism, while at the same time yielding a low-variance estimator.

We address these issues by building coupling estimators based on the iterated sampling importance resampling (ISIR) algorithm [Andrieu et al., 2010]. ISIR is in turn based on
importance sampling, and thus it uses multiple samples to reduce the variance of the estimator, similarly to the IWAE. Like the IWAE, we simultaneously fit an encoder as the proposal distribution for the importance sampling algorithm. Unfortunately, the variance of ISIR is still not small enough for fitting VAEs. To that end, we develop an extension of ISIR, called dependent iterated sampling importance resampling (DISIR), that combines the use of dependent importance samples and reparameterization ideas. DISIR drastically reduces the running time and the variance of the resulting coupling estimator. We develop a DISIR-based coupling estimator to estimate the gradient of the VAE log-likelihood.

**Contributions.** Our contributions are as follows.

- We show that the unbiased gradient estimators based on ISIR are not practical for VAEs, since they require to run the Markov chains for a long time even for only moderately high-dimensional models.
- We develop DISIR, an extension of ISIR that reduces the running time and the variance of the estimators.
- We use DISIR to form unbiased gradient estimators for VAEs. The resulting estimator is widely applicable as it can be used wherever the IWAE bound is applicable.
- We prove that the unbiased gradient estimates have finite variance and can be computed in finite expected time under some regularity conditions.
- We demonstrate experimentally that VAEs fitted with the DISIR-based gradient estimators exhibit better predictive log-likelihood on binarized MNIST, fashion-MNIST, and CIFAR-10, when compared to IWAE.

## 2 BACKGROUND

Here, we review the IWAE bound [Burda et al., 2016] and show that it can be seen as a standard ELBO on an augmented model. Consider a model \( p_\theta(x, z) \) of data \( x \) and latent variables \( z \in \mathcal{Z} \), and a proposal distribution \( q_\phi(z | x) \), where \( \theta \) and \( \phi \) denote the model and proposal parameters, respectively. Let the proposal satisfy the assumption below.

**Assumption 1** (The proposal and posterior have the same support). For any \( z \in \mathcal{Z} \), we have \( q_\phi(z | x) > 0 \) if and only if \( p_\theta(x, z) > 0 \), so that \( 0 < w_{\theta, \phi}(z) < \infty \), where the importance weights are

\[
    w_{\theta, \phi}(z) := \frac{p_\theta(x, z)}{q_\phi(z | x)}.
\]

The IWAE bound. The IWAE is a lower bound of the marginal log-likelihood in Eq. 1 formed with \( K \geq 1 \) importance samples \( z_1:K \) from the proposal, i.e., \( \mathcal{L}(\theta) \geq \mathcal{L}_{\text{IWAE}}(\theta, \phi) \), with

\[
    \mathcal{L}_{\text{IWAE}}(\theta, \phi) = \mathbb{E}_{q_\phi(z_1:K | x)} \left[ \log \left( \frac{1}{K} \sum_{k=1}^{K} w_{\theta, \phi}(z_k) \right) \right], \tag{4}
\]

where \( q_\phi(z_1:K | x) := \prod_{k=1}^{K} q_\phi(z_k | x) \). Eq. 4 monotonically increases with \( K \), converging towards \( \mathcal{L}(\theta) \) as \( K \to \infty \). For the case \( K = 1 \), it recovers the standard ELBO,

\[
    \mathcal{L}_{\text{ELBO}}(\theta, \phi) = \mathbb{E}_{q_\phi(z | x)} \left[ \log w_{\theta, \phi}(z) \right]. \tag{5}
\]

The importance samples \( z_1:K \) also provide an approximation of the posterior \( p_\theta(z | x) \). Specifically, if we define the importance weights \( \hat{w}_{\theta, \phi}(z_k) := w_{\theta, \phi}(z_k) \) and the normalized importance weights \( \tilde{w}_{\theta, \phi}(k) \propto w_{\theta, \phi}(k) \), with \( \sum_{k=1}^{K} \tilde{w}_{\theta, \phi}(k) = 1 \), then the approximation is

\[
    \hat{p}_\theta(z | x) = \sum_{k=1}^{K} \tilde{w}_{\theta, \phi}(k) \delta_{z_k}(z), \tag{6}
\]

where \( \delta_{z_k}(\cdot) \) is the delta Dirac measure located at \( z_k \).

**Fitting VAEs.** VAEs parameterize the likelihood \( p_\theta(x | z) \) using a distribution whose parameters are given by a neural network (decoder) that inputs the observation \( x \). Fitting a VAE involves maximizing the bound (either Eq. 4 or Eq. 5) w.r.t. both \( \theta \) and \( \phi \) using stochastic optimization. For that, the VAE uses unbiased gradient estimators of the objective. To form such gradients, we typically assume that \( q_\phi(z | x) \) is parameterizable [Kingma and Welling, 2014, Rezende et al., 2014, Titsias and Lázaro-Gredilla, 2014]. For convenience, we additionally assume that we can reparameterize in terms of a Gaussian (but we can easily relax this latter assumption).

**Assumption 2** (The variational distribution is parameterizable in terms of a Gaussian). There exists a mapping \( q_\phi(\xi, x) \) such that by sampling \( \xi \sim q(\xi) \), where \( q(\xi) = \mathcal{N}(\xi; 0, I) \), and setting \( z = g_\phi(\xi, x) \), we obtain \( z \sim q_\phi(z | x) \).

The IWAE bound as a standard ELBO. The IWAE bound in Eq. 4 can be interpreted as a regular ELBO on an augmented latent space [Cremer et al., 2017, Domke and Sheldon, 2018], and we use this perspective in Sections 3 and 4. Indeed, consider the \( K \) importance samples \( z_1:K \) and an indicator variable \( \ell \in \{1, \ldots, K\} \). We next define a generative model with latent variables \( (z_1:K, \ell) \), as well as a variational distribution, such that its ELBO recovers Eq. 4.

The augmented generative model posits that the indicator \( \ell \sim \text{Cat}(\frac{1}{K}, \ldots, \frac{1}{K}) \), where \( \text{Cat} \) denotes the categorical distribution. Given \( \ell \), each \( z_\ell \) is distributed according to \( q_{\phi}(z | x) \), except the \( \ell \)-th one, which follows the prior:

\[
    p_{\theta, \phi}(x, z_1:K, \ell) = \frac{1}{K} p_{\theta}(x, z_\ell) \prod_{k=1, k \neq \ell}^{K} q_{\phi}(z_k | x). \tag{7}
\]

Under the corresponding augmented posterior distribution, \( p_{\theta, \phi}(z_1:K, \ell | x) \propto p_{\theta, \phi}(x, z_1:K, \ell) \), the random variable \( z_\ell \)
follows the posterior \( p_\theta(z \mid x) \). We next define a variational distribution on the same augmented space,

\[
q_{\theta,\phi}(z_{1:K}, \ell \mid x) = \text{Cat}(\ell \mid \tilde{w}_{1,\phi}^{(1)}, \ldots, \tilde{w}_{K,\phi}^{(K)}) \prod_{k=1}^{K} q_\phi(z_k \mid x).
\]

(8)

We recover the IWAE bound (Eq. 4) as the ELBO of the augmented model, i.e., as \( \mathbb{E}_{\theta,\phi(z_{1:K}, \ell \mid x)} [\log p_\theta(x, z_{1:K}, \ell)] \).

Towards unbiased gradient estimation. The gradient w.r.t. \( \theta \) of the IWAE bound in Eq. 4 can be interpreted as a (biased) approximation of \( \nabla_\theta \mathcal{L} \) from Eq. 2. To see this, note that \( \nabla_\theta \mathcal{L}_{\text{IWAE}} = \mathbb{E}_{\theta}(z \mid x) [\nabla_\theta \log p_\theta(x, z)] \); i.e., \( \nabla_\theta \mathcal{L}_{\text{IWAE}} \) can be seen as an approximation of \( \nabla_\theta \mathcal{L} \) where we replace the posterior \( p_\theta(z \mid x) \) with the approximation \( \hat{p}_\theta(z \mid x) \) in Eq. 6.

To obtain an unbiased estimate, we need an alternative approximation \( \hat{p}_\theta(z \mid x) \) of \( p_\theta(z \mid x) \) satisfying \( \mathbb{E}_{\hat{p}_\theta(z \mid x)} [\nabla_\theta \log p_\theta(x, z)] = \nabla_\theta \mathcal{L} \). One such example is the empirical measure of exact samples from \( p_\theta(z \mid x) \). However, as mentioned earlier, it is typically impossible to obtain such samples, as a finite run with an MCMC kernel only provides biased estimates when the chain is initialized out of equilibrium. We can obtain an approximation \( \hat{p}_\theta(z \mid x) \) that leads to unbiased estimation using two coupled MCMC chains [Jacob et al., 2020b]. In Sections 3 and 4 we build on this idea to develop a method for unbiased gradient estimation.

### 3 Importance Sampling-Based MCMC Schemes

#### 3.1 Iterated Sampling Importance Resampling (ISIR)

Here we review ISIR [Andrieu et al., 2010], an MCMC scheme that samples from the augmented posterior \( p_{\theta,\phi}(z_{1:K}, \ell \mid x) \) given by Eq. 7. Alternatively, ISIR can be interpreted as an algorithm of which targets the posterior \( p_\theta(z \mid x) \). Indeed, if \( (z_{1:K}, \ell) \sim p_{\theta,\phi}(z_{1:K}, \ell \mid x) \) is a sample from the augmented posterior, then \( z_{\ell} \sim p_\theta(z \mid x) \).

The ISIR transition kernel, \( K_{\text{ISIR}}(\cdot \mid z_{1:K}, \ell) \), takes the current state \( (z_{1:K}, \ell) \) and outputs a new state by following Algorithm 1. ISIR requires a proposal distribution; we use \( q_\phi(z \mid x) \). It also requires to compute the importance weights \( w_{\theta,\phi}(\cdot) \) in Eq. 3. Note that ISIR has a resampling step (Line 4) that is distinct from the resampling step of sequential Monte Carlo, which would involve resampling multiple times from the categorical before mutating the samples.

The kernel is invariant w.r.t. \( p_{\theta,\phi}(z_{1:K}, \ell \mid x) \), as formalized in Proposition 5 in Appendix C.1, if Assumption 3 below is satisfied [Andrieu et al., 2010, 2018]. Assumption 3 holds when the proposal is at least as heavy-tailed as the target.

**Assumption 3** (The importance weights are bounded). There exists \( w_{\theta,\phi}^{\text{max}} < \infty \) such that \( w_{\theta,\phi}(z) \leq w_{\theta,\phi}^{\text{max}} \forall z \in \mathcal{Z} \).

#### 3.2 Dependent Iterated Sampling Importance Resampling (DISIR)

For moderately high-dimensional \( z \), ISIR can be inefficient. Indeed, if the importance weights become dominated by the weight of a single sample, then the corresponding Markov chain will typically get “stuck” for a large number of iterations. We illustrate this in Figure 1, which shows a one-dimensional illustrative comparison. In this toy experiment, we can observe that the ISIR chains get stuck when the state \( z^{(t)} \) corresponds to a high-weight region.

To mitigate this problem, here we develop DISIR, an extension of ISIR that uses dependent importance samples. Intuitively, this scheme proposes dependent samples \( z_{1:K} \sim \tilde{w}_{\text{isr}} \cdot z_{1:K} \) that are close to the current sample \( z_t \) with high probability, in the spirit of Shestopaloff and Neal [2018]. This modification increases the probability that the chain transitions to one of the new proposed values. As a result, in practice, the gradient estimators based on DISIR have smaller variance than the ones based on ISIR.

To make samples dependent, we use the reparameterization property (see Assumption 2) and introduce dependencies among the auxiliary variables \( \xi_{1:K} \); this induces dependen-
We refer to the coefficient \(\beta\) as the \emph{correlation strength}. When \(\beta = 0\), we have \(\xi_* = \xi_{\text{new}}\), and this approach is simply a reparameterized version of ISIR, which favours exploration of new regions of the space. When \(\beta\) approaches 1, all the proposed values \(\xi_*\) become closer to the current state \(\xi_t\) (which may correspond to the sample whose importance weight currently dominates). This dependency among the samples \(\xi_*\) induces dependencies among \(z_{1:K}\), resulting in more uniform importance weights. Thus, we say that this approach favours exploitation.

As given in Algorithm 2, DISIR is an MCMC scheme that targets the augmented density provided below.

**Algorithm 1: ISIR kernel, \(K_{\text{ISIR}}(\cdot, \cdot | z_{1:K}, \ell)\)**

**Input:** Current state of the chain, \((z_{1:K}, \ell)\)

**Output:** Next state of the chain

1. Sample \(\ell_{\text{aux}} \sim \text{Cat}(\frac{1}{K}, \ldots, \frac{1}{K})\)
2. Set \(z^*_{\text{aux}} = z_\ell\)
3. Sample \(z^*_{\text{aux}} \sim q_\ell(x)\) for \(k \in \{1, \ldots, K\}\) \(\setminus \{\ell_{\text{aux}}\}\)
4. Sample \(\ell^* \sim \text{Cat}(p_{1:K}\) with \(p_k \propto w_{\theta, \phi}(z^*_k)\)
5. Return \((z^*_{1:K}, \ell^*)\)

**Algorithm 2: DISIR kernel, \(K_{\text{DISIR}}(\cdot, \cdot | \xi_{1:K}, \ell)\)**

**Input:** Current state of the chain \((\xi_{1:K}, \ell)\) and correlation strength \(\beta\)

**Output:** New state of the chain

1. Sample \(\ell_{\text{aux}} \sim \text{Cat}(\frac{1}{K}, \ldots, \frac{1}{K})\)
2. Set \(\xi^*_{\text{aux}} = \xi_\ell\)
3. Sample \(\xi_{\text{new}}(x) \sim q_k(x)\) for \(k \in \{1, \ldots, K\}\) \(\setminus \{\ell_{\text{aux}}\}\)
4. Set \(\xi_{k+1} = \beta \xi_{k-1} + \sqrt{1 - \beta^2} \xi_{\text{new}}\) for \(k = \ell_{\text{aux}} + 1, \ldots, K\)
5. Set \(\xi_{k+1} = \beta \xi_{k-1} + \sqrt{1 - \beta^2} \xi_{\text{new}}\) for \(k = \ell_{\text{aux}} - 1, \ldots, 1\)
6. Set \(z^*_k = g_\phi(\xi^*_k, x)\) for \(k = 1, \ldots, K\)
7. Sample \(\ell^* \sim \text{Cat}(p_{1:K}\) with \(p_k \propto w_{\theta, \phi}(z^*_k)\)
8. Return \((\xi^*_{1:K}, \ell^*)\)

as invariant distribution and is ergodic.

The proofs of all propositions and lemmas are provided in Appendix C.

This target distribution has two desired properties. First, when the correlation strength \(\beta \in [0, 1)\), the \(\ell\)-th sample \(z_\ell\) is distributed according to the posterior \(p_\theta(z | x)\). That is, DISIR defines a Markov chain that targets Eq. 10, and \((z^{(l)}_{1:K})_{l \geq 0}\) is a Markov chain that converges to \(p_\theta(z | x)\). Second, when \(\beta = 0\), DISIR becomes identical to a reparameterized version of ISIR. That is, it simulates a Markov chain \((\xi^{(l)}_{1:K}, \ell^{(l)})_{l \geq 0}\) such that, setting each \(z^*_k = g_\phi(\xi^{(l)}_k, x)\), the Markov chain \((z^{(l)}_{1:K}, \ell^{(l)})_{l \geq 0}\) obeys a law that is identical to the one simulated by ISIR. This is formalized below.

**Lemma 1 (Distribution of DISIR samples).** Let Assumptions 1 and 2 hold. For any \(\beta \in [0, 1)\), we have \(z_\ell = g_\phi(\xi_{\ell}, x) \sim p_\theta(z | x)\) under \(K_{\text{DISIR}}(\cdot, \cdot | \xi_{1:K}, \ell)\). Moreover, for \(\beta > 0\), if \((\xi_{1:K}, \ell) \sim p_\theta(z_{1:K}, \ell | x)\), then we have \((z_{1:K}, \ell) \sim p_{\theta, \phi}(z_{1:K}, \ell | x)\), where each \(z_k = g_\phi(\xi_k, x)\).

In practice, we interleave DISIR steps for which \(\beta > 0\) with steps for which \(\beta = 0\). Specifically, we define a composed kernel that consists of the consecutive application of two steps of Algorithm 2. The first step has \(\beta = 0\) and favours exploration; the second step has \(\beta > 0\) and favours exploitation. It is possible to interleave the two kernels since both can be reinterpreted as MCMC kernels targeting \(p_\theta(z | x)\). We denote the composed kernel as \(K_{\text{DISIR,DISIR}}\).

**Choice of the correlation strength.** For the second step of the composed kernel, we wish to use a value \(\beta\) close to 1 to achieve exploitation, but not too close because then we will effectively have one importance sample repeated \(K\) times. We set \(\beta\) following a heuristic that is based on the effective sample size (ESS), defined as \(\text{ESS} = (\sum_{k=1}^{K} w_{k, \theta, \phi}^2)^{-1}\). As \(\beta\) becomes closer to 1, the ESS becomes closer to \(K\). We set the target ESS to 0.3\(K\), and we update \(\beta\) after each
step of the kernel, so that the ESS becomes closer to the target value. In particular, we apply the update rule $\beta \leftarrow \beta - 0.01(\text{ESS} - 0.3K)$. We also constrain the resulting $\beta \in [10^{-6}, 1 - 10^{-6}]$ to avoid numerical issues.

Since we only modify $\beta$ between iterations of the algorithm (and not while running the algorithm), the invariance of the Markov kernel w.r.t. to the target still holds. This also implies that the estimators that we derive in Section 4 are unbiased despite the adaptation of $\beta$.

### 3.3 DISIR ESTIMATES OF EXPECTATIONS

The DISIR samples $(z^{(t)}_{\phi(x)})_{t \geq 0}$ are distributed asymptotically as $p_{\theta}(z \mid x)$ and, in the limit of infinite samples, we can estimate expectations under $p_{\theta}(z \mid x)$, as $\frac{1}{T+1} \sum_{t=0}^{T} h(z^{(t)}_{\phi(x)}) \to E_{p_{\theta}(z \mid x)}[h(z)]$ almost surely.

However, it may seem wasteful to generate $K-1$ proposals at each iteration of Algorithm 2 and then use only the $\ell$-th importance sample to estimate an expectation. The proposition below shows that it is possible to use all the $K$ importance samples: we can estimate an expectation $E_{p_{\theta}(z \mid x)}[h(z)]$ with a weighted average of the importance samples.

**Proposition 2** (The $K$ importance samples can be used for estimating expectations). For any function $h : \mathcal{Z} \to \mathbb{R}$ such that $E_{p_{\theta}(z \mid x)}[\mid h(z) \mid ] < \infty$, we have the identity

$$E_{p_{\theta}(z \mid x)}[h(z)] = E_{\pi_{\theta, \phi}}(\xi_{1,K}, \ell \mid x) \sum_{k=1}^{K} \bar{w}_{\theta, \phi}(z_{k}) h(z_{k})$$

where $z_{k} = g_{\phi}(\xi_{k}, x)$ and the normalized importance weights are $\bar{w}_{\theta, \phi}(z_{k}) \propto w_{\theta, \phi}(z_{k})$ with $\sum_{k=1}^{K} \bar{w}_{\theta, \phi}(z_{k}) = 1$. Setting $h(z) = \nabla_{\theta} \log p_{\theta}(x, z)$, and given that the DISIR kernel is ergodic, it follows from Eq. 2 that

$$\frac{1}{T+1} \sum_{t=0}^{T} \sum_{k=1}^{K} \bar{w}_{\theta, \phi}(z_{k}) \nabla_{\theta} \log p_{\theta}(x, z^{(t)}_{\phi}) \to \nabla_{\theta} \mathcal{L}$$

almost surely as $T \to \infty$ for any $K \geq 2$, where $z^{(t)}_{\phi} = g_{\phi}(\xi^{(t)}_{k}, x)$ and $\bar{w}_{\theta, \phi}(z_{k})$.

### 3.4 CHOICE OF THE PROPOSAL

Both ISIR and DISIR require a proposal distribution to sample the states $z^*_{t}$, for which we use $q_{\phi}(z \mid x)$ (DISIR additionally requires the proposal to be reparameterizable). Here we discuss how to set the parameters $\phi$ of the proposal.

Like for IAWE, in our case $q_{\phi}(z \mid x)$ is a proposal distribution rather than a variational posterior. We fit $\phi$ via stochastic optimization of the IAWE bound in Eq. 4. To estimate the gradient w.r.t. $\phi$, we use the doubly reparameterized estimator [Tucker et al., 2019], which addresses some issues of the estimator of Burda et al. [2016] for large values of $K$ [Rainforth et al., 2019].

Alternatively, we could fit the encoder using the forward Kullback-Leibler (KL) divergence. This would imply to maximize $E_{p_{u}(z \mid x)}[\log q_{\phi}(z \mid x)]$ w.r.t. $\phi$, for which we can apply the unbiased estimators of Section 4 to estimate the expectation w.r.t. $p_{u}(z \mid x)$. We leave this for future work.

### 4 UNBIASED GRADIENT ESTIMATION WITH MCMC COUPLINGS

#### 4.1 UNBIASED ESTIMATION WITH COUPLINGS

In this section, we review how to obtain an unbiased gradient estimator using two coupled Markov chains. We use the notation $u \in \mathcal{U}$ to refer to a generic random variable, keeping in mind that we will later set $u = [\xi_{1,K}, \ell]$, i.e., the latent variables in the augmented space.

Consider the estimation of the expectation

$$H := E_{\pi_{u}}[h(u)]$$

for some distribution $\pi_{u}$ and function $h(u)$. As discussed in Section 2, a direct approximation $\hat{\pi}(u)$ of $\pi_{u}$ via MCMC leads to a biased estimator.

We can obtain an unbiased estimator based on two coupled MCMC chains, each with invariant distribution $\pi_{u}$ [Glynn and Rhee, 2014, Jacob et al., 2020b]. The two chains have the same marginals at any time instant $t$, but they evolve according to a joint transition kernel $K_{C}$. Let $K_{C}(- \mid u)$ be the marginal transition kernel of each chain, and let $K_{C}(\cdot \mid u, \bar{u})$ be a joint kernel that takes the state of both chains (denoted $u$ and $\bar{u}$) and produces the state of both chains.

We next review the unbiased estimator of Vanetti and Doucet [2020], which reduces the variance of the estimator of Jacob et al. [2020b]. The main idea is to consider a lag $L \geq 1$ and jointly sample the states of both chains $(u^{(t)}, \bar{u}^{(t-L)})$ conditioned on their previous states, i.e., $(u^{(t-1)}, \bar{u}^{(t-L-1)})$.

We then use (a finite number of) the samples from each chain to obtain the unbiased estimator (Appendix A shows how to derive it). Practically, we initialize the first Markov chain from some (arbitrary) initial distribution $\pi_{0}(-)$, i.e., $u^{(0)} \sim \pi_{0}(u)$. We then sample this Markov chain using the marginal kernel $K_{C}$, i.e., $u^{(t)} \sim K_{C}(u \mid u^{(t-1)})$ for $t = 1, \ldots, L$. After $L$ steps, we draw the initial state of the second Markov chain $\bar{u}^{(0)}$ (potentially conditionally upon $u^{(L-1)}$, $\bar{u}^{(L)}$), such that marginally $\bar{u}^{(0)} \sim \pi_{0}(u)$.

Afterwards, for $t > L$, we draw both states jointly as $(u^{(0)}, \bar{u}^{(t-L)}) \sim K_{C}(u \mid \bar{u}^{(t-L)}, \bar{u}^{(t-L-1)})$.

The joint kernel $K_{C}$ is chosen such that, after some time, both chains produce the same exact realizations of the random

---

1 The joint kernel is such that $K_{C}(A, U \mid u, \bar{u}) = K_{C}(A \mid u)$ and $K_{C}(U, A \mid u, \bar{u}) = \mathcal{K}(A \mid \bar{u})$ for any measurable set $A$. 711
variables, i.e., \( u(t) = \tilde{u}(t-L) \) for \( t \geq \tau \). Here, \( \tau \) is the meeting time, defined as the first time instant in which both chains meet, \( \tau = \inf\{t \geq L : u(t) = \tilde{u}(t-L)\} \) (it could be infinite, but we design the joint kernel so that \( \tau \) is a random variable of finite expected value).

Based on this coupling procedure, the unbiased estimator of Eq. 14 is

\[
\hat{H} = \frac{1}{L} \left( \sum_{t=t_0+L}^{t_0+L-1} h(u(t)) + \sum_{t=t_0+L}^{t_0+L} \left( h(u(t)) - h(\tilde{u}(t-L)) \right) \right)
\]

where \( t_0 \) is a constant that plays the role of the burn-in period, although it is not a burn-in period in the usual sense, since we do not require the Markov chains to converge. Indeed, the estimator in Eq. 14 requires us to run the coupled Markov chains until they meet each other. Given that we design the joint kernel such that the meeting time \( \tau \) is finite, this implies that we obtain the unbiased estimator in finite time.

We can also think of this unbiased coupling procedure as providing an empirical approximation\(^2\) \( \hat{\pi}(\cdot) \) of \( \pi(\cdot) \),

\[
\hat{\pi}(u) = \frac{1}{L} \left( \sum_{t=t_0}^{t_0+L-1} \delta_{u(t)}(u) + \sum_{t=t_0+L}^{t_0+L} \left( \delta_{u(t)}(u) - \delta_{\tilde{u}(t-L)}(u) \right) \right).
\]

We next provide sufficient conditions that ensure that the estimator in Eq. 14 can be computed in expected finite time and has finite variance. These conditions are similar as for the original estimator of [Jacob et al., 2020b; Middleton et al., 2020] but the proof is slightly different.

**Proposition 3** (The unbiased estimator can be computed in finite time and has finite variance). Assume the following conditions hold:

a. (Convergence of the Markov chain.) Each of the two chains marginally starts from a distribution \( \pi_0 \), evolves according to a transition kernel \( \mathcal{K} \) and is such that \( \mathbb{E}[h(u(t))] \rightarrow \mathbb{E}_{\pi_0}[h(u)] \) as \( t \to \infty \).

b. (Finite high-order moment.) There exist \( \eta > 0 \) and \( D < \infty \) such that \( \mathbb{E}[h(u(t))^{2+\eta}] \leq D \) \( \forall t \geq 0 \).

c. (Distribution of the meeting time.) There exists an almost surely finite meeting time \( \tau = \inf\{t \geq L : u(t) = \tilde{u}(t-L)\} \) such that \( \mathbb{P}(\tau > t) \leq C \tau^{-\alpha} \) for some \( C < \infty \) and \( \alpha > 2(\eta + 1) \), where \( \eta \) appears in Condition (b).

d. (The chains stay together after meeting.) We have \( u(t) = \tilde{u}(t-L) \) for all \( t \geq \tau \).

Then, Eq. 14 is an unbiased estimator of \( \mathbb{E}_{\pi(u)}[h(u)] \) that can be computed in finite expected time and has finite variance.

Conditions (a) and (d) can be satisfied by careful design of the joint kernel \( \mathcal{K}_C \). Condition (b) is a mild integrability condition. Condition (c) can be satisfied if the marginal kernel \( \mathcal{K} \) is (only) polynomially ergodic and some additional mild irreducibility and aperiodicity conditions on the joint kernel \( \mathcal{K}_C \) hold [Middleton et al., 2020].

### 4.2 COUPLING DISIR

Here we describe the main algorithm of this paper: an estimator based on two coupled Markov chains, each evolving according to the DISIR transition kernel from Section 3.2. That is, we build a joint kernel \( \mathcal{K}_C \) for unbiased gradient estimation. We denote the joint kernel as \( \mathcal{K}_{\text{C-DISIR}}((\cdot, \cdot), (\xi_{1,K}, \ell), (\xi_{1,K}, \hat{\ell})) \).

**Algorithm 3:** C-DISIR kernel for two coupled chains, \( \mathcal{K}_{\text{C-DISIR}}((\cdot, \cdot), (\xi_{1,K}, \ell), (\xi_{1,K}, \hat{\ell})) \)

**Input:** Current state of both chains, \( (\xi_{1,K}, \ell) \) and \( (\xi_{1,K}, \hat{\ell}) \), and correlation strength \( \beta \)

**Output:** New state of both chains

1. Sample \( \xi^\text{aux}_{l_{\text{aux}}} \sim \text{Cat} \left( \frac{1}{K}, \ldots, \frac{1}{K} \right) \)
2. Set \( \xi^*_{l_{\text{aux}}} = \xi^\beta_{l_{\text{aux}}} \) and \( \xi^\text{new}_{l_{\text{aux}}} = \xi^\beta_{l_{\text{aux}}} \)
3. Sample \( \xi_{l_{\text{new}}}_{k}^{\text{new}} \sim q(\xi) \) for \( k \in \{1, \ldots, K\} \setminus \{l_{\text{aux}}\} \)
4. Set \( \xi_{l_{\text{new}}}_{k}^{\text{new}} = \beta \xi_{l_{\text{new}}}_{k-1} + \sqrt{1 - \beta^2} \xi_{l_{\text{new}}}_{k}^{\text{new}} \) and \( \xi_{l_{\text{new}}}_{k}^{\text{new}} = \beta \xi_{l_{\text{new}}}_{k-1} + \sqrt{1 - \beta^2} \xi_{l_{\text{new}}}_{k}^{\text{new}} \) for \( k = l_{\text{aux}} + 1, \ldots, K \)
5. Set \( \xi_{l_{\text{new}}}_{k}^{\text{new}} = \beta \xi_{l_{\text{new}}}_{k+1} + \sqrt{1 - \beta^2} \xi_{l_{\text{new}}}_{k}^{\text{new}} \) for \( k = l_{\text{aux}} - 1, \ldots, 1 \)
6. Set \( z_{k}^{\text{new}} = g_{\beta}(\xi_{l_{\text{new}}}_{k}, x) \) and \( \bar{z}_{k}^{\text{new}} = g_{\beta}(\xi_{l_{\text{new}}}_{k}, x) \) for \( k = 1, \ldots, K \)
7. Sample \( \ell^*, \bar{\ell}^* \sim \mathcal{K}_{\text{C-Cat}}(\ell, \bar{\ell}, (w_{\theta, \phi}(z_{1}^{\star}), \ldots, w_{\theta, \phi}(z_{K}^{\star})), (w_{\theta, \phi}(\bar{z}_{1}^{\star}), \ldots, w_{\theta, \phi}(\bar{z}_{K}^{\star}))) \) from the maximal coupling kernel (Algorithm 5 in Appendix B)
8. Return \( (\xi_{1,K}^{\text{C}}, \ell^*), (\xi_{1,K}^{\text{C}}, \bar{\ell}^*) \)

\(^2\)A coupling procedure is maximal if it maximizes the probability that both chains meet.
Algorithm 4: Unbiased estimation with C-ISIR-DISIR

Input: The constant $t_0$ and the lag $L$

Output: An unbiased estimator of $\nabla_{\theta} \mathcal{L}$

\begin{itemize}
\item Initialize $\xi_k \sim q(\xi)$ and $\bar{\xi}_k \sim q(\xi)$ for $k = 1, \ldots, K$
\item Initialize $\ell \sim \text{Cat}(\frac{1}{K}, \ldots, \frac{1}{K})$ and $\bar{\ell} \sim \text{Cat}(\frac{1}{K}, \ldots, \frac{1}{K})$
\item for $t = 1, \ldots, L$ do
\begin{itemize}
\item Sample $\left(\xi_{K+1}^{t}, \ell(t)\right) \sim \mathcal{K}_{\text{C-ISIR-DISIR}}(\cdot, \cdot | \xi_{K}^{t-1}, \ell(t-1))$ (two steps of Algorithm 2, with $\beta = 0$ then $\beta > 0$)
\end{itemize}
\item end
\item Set the iteration $t = L$
\item while $t < t_0 + L - 1$ or the two chains have not met do
\begin{itemize}
\item Sample $\left(\xi_{K+1}^{t+1}, \ell(t+1)\right), \left(\xi_{K+1}^{t+L+1}, \ell(t+L+1)\right) \sim \mathcal{K}_{\text{C-ISIR-DISIR}}(\cdot, \cdot, \cdot | \xi_{K}^{t}, \ell(t), \xi_{K}^{t-L}, \ell(t-L))$
\item Increase $t \leftarrow t + 1$
\end{itemize}
\item end
\item Return the estimator from Eq. 14 using the function $h(\xi_{1:K}, \ell) := \sum_{k=1}^{K} \bar{w}_{\theta, \phi}(k) \nabla_{\theta} \log p_{\theta}(x, z_k)$
\end{itemize}

That when $\beta = 0$, Algorithm 3 shares the same values of the noise values generating the importance samples for both chains, i.e., $\xi_k^* = \bar{\xi}_k^*$ for $k \neq \ell_{\text{aux}}$, while the $\ell_{\text{aux}}$-th importance sample is set to the current state of each chain, i.e., $\xi_{\ell_{\text{aux}}}^* = \xi_\ell$ and $\bar{\xi}_{\ell_{\text{aux}}}^* = \bar{\xi}_\ell$. Thus, if the indicators sampled in Line 7 take the same value (i.e., $\ell^* = \bar{\ell}^*$) and this value is different from $\ell_{\text{aux}}$, then both chains meet. After meeting, for any future iteration of the joint kernel, the states of both chains are guaranteed to be identical to each other. On the contrary, when the correlation strength $\beta \neq 0$, coupling cannot occur. However, for any $\beta \in [0, 1)$, Algorithm 3 guarantees that the chains remain equal to each other once they have previously met.

We use a composed kernel that consists of the consecutive application of two steps of Algorithm 3. The first step has $\beta = 0$; in this step the chains may meet each other. The second step has $\beta > 0$, which favours exploitation. As discussed in Section 3.2, it is valid to combine these two kernels. We denote this composed joint kernel as $\mathcal{K}_{\text{C-ISIR-DISIR}}$.

Unbiased gradient estimation with C-ISIR-DISIR. Algorithm 4 describes the procedure that provides an unbiased estimator of $\nabla_{\theta} \mathcal{L}$. It samples two Markov chains, $\left(\xi_{K}^{t}, \ell(t)\right)$ and $\left(\xi_{K}^{t}, \bar{\ell}(t)\right)$, by inducing a coupling between the state of the first chain at time $t$ and the state of the second chain at time $t - L$, where $L \geq 1$ is the lag. After both chains meet, it returns the unbiased gradient estimator using Eq. 14 for the function $h(\xi_{1:K}, \ell) := \sum_{k=1}^{K} \bar{w}_{\theta, \phi}(k) \nabla_{\theta} \log p_{\theta}(x, z_k)$ (applying Proposition 2), where $z_k = g_{\theta}(\xi_k, x)$ and the normalized importance weights are $\bar{w}_{\theta, \phi}(k) \propto w_{\theta, \phi}(z_k)$. Algorithm 4 provides a practical unbiased estimator, as we show next.

Algorithm 4 returns an unbiased estimator of $\nabla_{\theta} \mathcal{L}$ of finite variance that can be computed in finite expected time. Additionally, $\mathbb{E}[\tau]$ can be upper bounded by a quantity decreasing with $K$.

5 RELATED WORK

Our estimator builds on previous work discussed in the former sections. We now review other related works.

ISIR, as well as other particle MCMC algorithms, has been previously used for smoothing in state-space models [Andrieu et al., 2010] and for (biased) estimation of $\nabla_{\theta} \mathcal{L}$ [Naesseth et al., 2020]. Coupled variants of these algorithms have also been previously developed for unbiased smoothing [Jacob et al., 2020a, Middleton et al., 2019]. Indeed, Algorithm 3 for $\beta = 0$ has been used by Jacob et al. [2020a] (without reparameterization). However, we found experimentally that the unbiased estimators based on coupled ISIR suffer from high variance for moderately high dimensions, making them impractical for VAEs. The estimators based on coupled DISIR with $\beta \approx 1$ address this issue.

An unbiased estimator based on a coupled Gibbs sampler has also been presented for restricted Boltzmann machines [Qiu et al., 2019], but this method is not applicable for VAEs. An alternative unbiased gradient estimator for VAEs, based on Russian roulette ideas, was developed by Luo et al. [2020]. However, this estimator suffers from high variance (potentially infinite), and requires additional variance reduction methods such as gradient clipping, which defeats the purpose of unbiased gradient estimation. In our experiments, we use RMSProp and no gradient clipping is needed.

Dieng and Paisley [2019] maximize the marginal log-likelihood of the data using an expectation maximization scheme that gives a consistent (but not unbiased) estimator.

Finally, note that the coupling estimators approximate the gradient w.r.t. $\theta$, and are orthogonal to the methods that improve the expressiveness of the encoder $q_{\phi}(z | x)$, such as semi-implicit methods [Yin and Zhou, 2018, Titsias and Ruiz, 2019] or normalizing flows [Rezende and Mohamed, 2015, Kingma et al., 2016, Papamakarios et al., 2017, Tomczak and Welling, 2016, 2017, Dinh et al., 2017], to name a few. These methods could be used together with the coupling estimators to obtain a more flexible proposal distribution, which could improve the mixing of the Markov chains.

6 EXPERIMENTS

In Section 6.1, we study the bias of different estimators in an experiment where we have access to the exact gradient $\nabla_{\theta} \mathcal{L}$. In Section 6.2, we study the predictive performance of VAEs trained with coupled DISIR and show that models fitted with unbiased estimators outperform those...
We first consider probabilistic principal component analysis
w.r.t. the intercept term. (In Appendix D.1, we show a
coupled ISIR, i.e., a variant of Algorithm 4 where
(C-ISIR-DISIR) exhibits lower variance.

The fourth estimator is based on Algorithm 4. For all the
estimators, we use the same (fixed) distribution
of the ELBO and IWAE bounds (\( \mathcal{L}_{\text{ELBO}} \) and \( \mathcal{L}_{\text{IWAE}} \)) = \( \sum_{n=1}^{N} \nabla_{\theta} \log q_{\phi}(x_n) \), for a batch of \( N = 100 \) datapoints, and we compare it
against four gradient estimators. Two estimators are the
gradients of the ELBO and IWAE
\( \nabla_{\theta} \mathcal{L}_{\text{ELBO}} \) and \( \nabla_{\theta} \mathcal{L}_{\text{IWAE}} \).
The third one is the unbiased estimator obtained with coupled
ISIR, i.e., a variant of Algorithm 4 where
we replace the \( \mathcal{K}_{\text{C-ISIR-DISIR}} \) kernel with \( \mathcal{K}_{\text{C-ISIR}} \) (which
is equivalent to \( \mathcal{K}_{\text{C-DISIR}} \) with correlation strength \( \beta = 0 \).
The fourth estimator is based on Algorithm 4. For all the
estimators, we use the same (fixed) distribution \( q_{\phi}(z | x) \).
For the coupling estimators, we set \( t_0 = 1 \) and the lag \( L = 10 \).

We obtain 50,000 samples from each estimator, and compute the (signed) error \( \nabla_{\theta} \mathcal{L} - \nabla_{\theta} \mathcal{L} \) for each sample. We show in Figure 2 the boxplot representation of the
error for a randomly chosen component of the gradient
w.r.t. the intercept term. (In Appendix D.1, we show a
randomly chosen weight term in Figure 5, and the average
over components in Figure 6.) As expected, the estimators
of the ELBO and IWAE gradients are biased. The boxplots
for C-ISIR and C-ISIR-DISIR are consistent with the
unbiasedness of the estimators, and the one based on
C-ISIR-DISIR has smaller variance. This property is key
for fitting more complex models such as VAEs.

6.2 Variational auto-encoder

Now we apply the coupling estimators to fit VAEs and compare
the predictive performance to the maximization of the
ELBO and IWAE objectives. (We also implemented the
method of Luo et al. [2020], but we found it led to unstable
optimization despite using gradient clipping.) We provide
further details on the experimental setup in Appendix D.2.

Binarized MNIST. We first fit a VAE on the statically
binarized MNIST dataset. We use \( K = 10 \) importance
samples and explore the dimensionality \( D = \{20, 100, 300\} \) of
\( z \in \mathbb{R}^D \). We use RMSProp [Tieleman and Hinton, 2012]
in the stochastic optimization procedure. For the coupling
estimators, we set \( t_0 = 1 \) step and the lag \( L = 10 \).

Following Wu et al. [2017], we estimate the predictive log-
likelihood using annealed importance sampling (AIS) [Neal,
2001]. Specifically, we use 16 independent AIS chains, with
10,000 intermediate annealing distributions, and a transition
operator consisting of one Hamiltonian Monte Carlo (HMC)
trajectory with 10 leapfrog steps and adaptive acceptance
rate tuned to 0.65. For CIFAR-10, we use 4 AIS chains
with 7,500 intermediate distributions and 5 HMC leapfrog
steps. As this procedure is computationally intensive, we
only evaluate the train log-likelihood on the current data
batch, but we evaluate the log-likelihood on the entire test
set at the end of the optimization.

Figure 3(a) shows the evolution of the train log-likelihood;
the error bars correspond to the standard deviation of 10
independent runs. (Figure 7 in Appendix D.3 shows similar
plots for varying \( D \).) Table 1(a) shows the test log-likelihood
after 300 epochs. The VAE models fitted with the unbiased
estimator of Algorithm 4 have better predictive performance.

The \( \mathcal{K}_{\text{C-ISIR-DISIR}} \) kernel in Algorithm 4 is key for obtaining
this improved performance. As a comparison, replacing it
with \( \mathcal{K}_{\text{C-ISIR}} \) leads to a test log-likelihood value of
\(-90.70 \pm 0.08 \) for \( D = 20 \), i.e., it is worse than using the standard
ELBO (and the gap with the ELBO gets larger for increasing
dimensionality \( D \)). Moreover, \( \mathcal{K}_{\text{C-ISIR-DISIR}} \) alleviates the
computational complexity of \( \mathcal{K}_{\text{C-ISIR}} \), as measured by the
number of MCMC iterations it requires. Figure 4 compares
the histograms of the meeting time \( \tau \) for both kernels; C-
ISIR-DISIR requires significantly fewer iterations. (Figure 8
in Appendix D.4 shows that the histograms behave similarly
different values of \( D \).)

The improved performance over IWAE comes at the expense
of computational complexity. The cost of Algorithm 4 is
roughly 10 times the cost of computing \( \nabla_{\theta} \mathcal{L}_{\text{IWAE}} \).

Fashion-MNIST and CIFAR-10. The estimator based on
Algorithm 4 leads to improved models but it is also
computationally more expensive. We now study the effect of
switching to Algorithm 4 after fitting a VAE using the IWAE
objective. That is, we first fit the VAE using the IWAE

\( \nabla_{\theta} \mathcal{L}_{\text{IWAE}} \). The model is
fitting to a Gaussian whose parameters depend linearly on
\( z \), \( \theta \in \mathbb{R}^K \), \( z \in \mathbb{R}^{K+1} \), and \( \theta_0 + \theta_1 z, 0.1I \), where
\( z \in \mathbb{R}^{100} \). We randomly set the model
parameters \( \theta \), and fit a variational distribution
\( q_{\phi}(z | x) \) by
maximizing the IWAE bound w.r.t. \( \phi \) with \( K = 100 \)
importance samples on binarized MNIST [Salakhutdinov and
Murray, 2008]. The distribution \( q_{\phi}(z | x) \) is a fully factorized
Gaussian whose parameters depend linearly on \( x \).

We obtain the exact gradient, \( \nabla_{\theta} \mathcal{L} = \sum_{n=1}^{N} \nabla_{\theta} \log p_{\theta}(x_n) \),
for a batch of \( N = 100 \)datapoints, and we compare it
against four gradient estimators. Two estimators are the
gradients of the ELBO and IWAE bounds
(\( \nabla_{\theta} \mathcal{L}_{\text{ELBO}} \) and
\( \nabla_{\theta} \mathcal{L}_{\text{IWAE}} \)). The third one is the unbiased estimator obtained with coupled
ISIR, i.e., a variant of Algorithm 4 where
we replace the \( \mathcal{K}_{\text{C-ISIR-DISIR}} \) kernel with \( \mathcal{K}_{\text{C-ISIR}} \) (which
is equivalent to \( \mathcal{K}_{\text{C-DISIR}} \) with correlation strength \( \beta = 0 \).
The fourth estimator is based on Algorithm 4. For all the
estimators, we use the same (fixed) distribution \( q_{\phi}(z | x) \).
For the coupling estimators, we set \( t_0 = 1 \) and the lag \( L = 10 \).

We obtain 50,000 samples from each estimator, and compute the (signed) error \( \nabla_{\theta} \mathcal{L} - \nabla_{\theta} \mathcal{L} \) for each sample. We show in Figure 2 the boxplot representation of the
error for a randomly chosen component of the gradient
w.r.t. the intercept term. (In Appendix D.1, we show a
randomly chosen weight term in Figure 5, and the average
over components in Figure 6.) As expected, the estimators
of the ELBO and IWAE gradients are biased. The boxplots
for C-ISIR and C-ISIR-DISIR are consistent with the
unbiasedness of the estimators, and the one based on
C-ISIR-DISIR has smaller variance. This property is key
for fitting more complex models such as VAEs.
We have developed a practical algorithm to obtain unbiased estimators of the gradient of the log-likelihood for intractable models, and we have shown empirically that VAEs fitted with unbiased estimators exhibit better predictive performance. Compared to ELBO or IWAE gradients, the main limitation of this approach is its higher computational cost and the fact that the running time is random. While one could obtain more accurate estimators simply by increasing the number of samples of the IWAE bound, this would significantly increase the memory requirement, making it impractical for datasets like CIFAR-10, and it would also remain biased for any (finite) number of samples.

The topic of coupling estimators is currently an active research field. We expect future work will improve the practical applicability of such estimators using methods like, e.g., control variates [Craiu and Meng, 2020].

7 DISCUSSION

We thank Andriy Mnih for his insightful comments and useful help. We also thank the anonymous reviewers and area chair for their feedback.

References


Igor Babuschkin, Kate Baumli, Alison Bell, Surya Bhupatiraju, Jake Bruce, Peter Buchlovsky, David Budden,


