From Monte Carlo to Las Vegas: Improving Restricted Boltzmann Machine Training Through Stopping Sets

Pedro H. P. Savarese
Toyota Technological Institute at Chicago
Chicago, IL, 60637
savarese@ttic.edu

Mayank Kakodkar
Department of Computer Science
Purdue University
West Lafayette, IN, 47903
mkakodkar@purdue.edu

Bruno Ribeiro
Department of Computer Science
Purdue University
West Lafayette, IN, 47903
ribeiro@cs.purdue.edu

Abstract

We propose a Las Vegas transformation of Markov Chain Monte Carlo (MCMC) estimators of Restricted Boltzmann Machines (RBMs). We denote our approach *Markov Chain Las Vegas* (MCLV). MCLV gives statistical guarantees in exchange for random running times. MCLV uses a stopping set built from the training data and has maximum number of Markov chain steps K (referred as MCLV-K). We present a MCLV-K gradient estimator (LVS-K) for RBMs and explore the correspondence and differences between LVS-K and Contrastive Divergence (CD-K), with LVS-K significantly outperforming CD-K training RBMs over the MNIST dataset, indicating MCLV to be a promising direction in learning generative models.

Introduction

Despite the significant recent advances in training discriminative neural network models, training generative models has proven more elusive. As with most neural network training methods, algorithms for training Restricted Boltzmann Machines (RBMs) (Hinton 2002; 2012; Smolensky 1986), a class of energy-based generative neural network models, are unreasonably effective. Though, some argue, not yet effective enough for modern applications. In this work we seek to better understand and improve the training of RBMs.

RBM is a family of energy-based models with probability distribution over a state vector $\mathbf{x} = (\mathbf{v}, \mathbf{h})$ (assumed discrete w.l.o.g.), $\mathbf{v} \in \{0,1\}^{n_V}$ (e.g. an image) and binary latent variables $\mathbf{h} \in \{0,1\}^{n_H}$,

$$p(\mathbf{x}; \mathbf{W}) = \frac{1}{Z(\mathbf{W})} e^{-E(\mathbf{x};\mathbf{W})}, \quad (1)$$

where $Z(\mathbf{W}) = \sum_{\mathbf{x}} e^{-E(\mathbf{x};\mathbf{W})}$ is a partition function with finite mean, $\mathbb{E}[Z(\mathbf{W})] < \infty$, and $E(\mathbf{x}; \mathbf{W})$ is an energy function given by

$$E\Big(\mathbf{x} = (\mathbf{v}, \mathbf{h}); \mathbf{W} = (\mathbf{W}', \mathbf{b}, \mathbf{a})\Big) = -\mathbf{v}^{\mathsf{T}}\mathbf{W}'\mathbf{h} - \mathbf{b}^{\mathsf{T}}\mathbf{v} - \mathbf{a}^{\mathsf{T}}\mathbf{h}.$$

RBMs proved highly successful in many tasks, such as data generation (Hinton 2002; Hinton, Osindero, and Teh 2006; Hinton 2012) and as a pre-training step for feedforward neural networks (Salakhutdinov and Hinton 2009), among others (see Bengio and Delalleau (2009) and Erhan et al. (2010)).

As computing $Z(\mathbf{W})$ directly is intractable for large state spaces, Markov Chain Monte Carlo (MCMC) methods are widely used to compute statistics of these models (including estimating the gradient $\partial p(\mathbf{x}; \mathbf{W})/\partial \mathbf{W}$). MCMC works by running a Markov chain (MC) $\Phi(\mathbf{W})$ with steady state $p(\mathbf{x}; \mathbf{W})$ to equilibrium. Metropolis-Hastings and Gibbs sampling are two general such approaches.

However, in the real world, one is expected to run the MC $\Phi(\mathbf{W})$ for only K steps, returning a state $\mathbf{x} \sim \hat{\pi}(K)$, "approximately sampled" from the Markov chain's true steady state distribution $p(\mathbf{x}; \mathbf{W})$. Starting from a random state, K needs to be quite large for this method to work.

Contrastive Divergence (CD-K) (Hinton 2002; 2012), improves this procedure by starting the MC from the visible states of the training data. Empirically, CD-K works tremendously well to train RBMs with few hidden units (n_H small) even for K as low as $K = 1$ (Carreira-Perpiñán and Hinton 2005; Hinton 2002; 2012).

For high-dimensional RBMs, CD-K is less efficient and the reason is conjectured to be the longer mixing times (Sutskever and Tieleman 2010), although concrete evidence is anecdotal as mixing times are hard to assess in high dimensions. While not the main focus of our paper, armed with our techniques, we will further empirically explore possible reasons for this high-dimensional difficulty.

A Las Vegas transformation of RBM training. The main focus of this paper is to recast MCMC estimation of RBMs as a Markov chain algorithm with stopping sets obtained from the training data. The size of the stopping set is a hyperparameter that can be dynamically adapted during training based on computational trade-offs.

In standard RBM training using MCMC, the MC stops after a predefined number of K steps. In our approach, the MCMC can also stop if it reaches one of the states in the stopping set. Thus, MCMC running times are random (and are, in average, shorter than K). This approach is closer to a Las Vegas algorithm than a Monte Carlo algorithm: we aim to get *perfect samples* of a quantity with an algorithm that has random running times. We denote this approach Markov Chain Las Vegas with K maximum steps (MCLV-K).

We show that, by dynamically adapting K, MCLV-K can find unbiased estimates of the direction of the RBM gradient. Moreover, in contrast to standard MCMC, MCLV-K has an extra piece of information: whether or not the stopping set has been reached. We show that this knowledge provides novel ways to estimate gradients and partition functions of energy-based models.

And perhaps, one of the most interesting observations in this paper comes from the **correspondence between CD-K and MCLV-K**. MCLV-1 is quite similar to CD-1 except for an added $\mathcal{S}$*-stopped* flag, where $\mathcal{S}$ is a set of *stopping* states defined later. Clearly, for $K \geq 2$, MCLV-K is distinct from CD-K, as the MC of MCLV-K may stop before performing all K steps.

Analyzing CD-K through our Las Vegas transformation, it is clear that CD-K has an unintended inspection paradox bias that can be corrected to further improve the RBM learning. Using the *reached-stopping-set* flag of MCLV-K, we design a new gradient estimator, denoted *Las Vegas Slope* (LVS), that empirically gives significantly better parameter estimates than standard CD-1 and CD-10 over the MNIST dataset according to the model's likelihood. MNIST is used in our experiments due to the long history of RBM development over this dataset.

Contributions. We claim the following contributions: **(1)** We introduce Markov Chain Las Vegas (MCLV-K). We show MCLV-K gives finite-sample unbiased and asymptotically consistent estimates of a variety of statistics of RBMs; further, we give two convergence bounds. We also show how to theoretically and empirically reduce the MCLV-K random running times using the training examples. **(2)** We show how MCLV-K can be used to design new ways to train Restricted Boltzmann Machines; we use MCLV-K to propose a novel RBM gradient estimator, *Las Vegas Slope* (LVS), which our empirical results show (for $K \in \{1, 3, 10\}$) improves parameter estimates of RBM over CD-1 and CD-10, over the MNIST dataset.

MCLV-K Estimation with Statistical Guarantees

In what follows we introduce some of the definitions used throughout our paper. We introduce the concept of a tour (a MC which returns to the same state) and show that the return probability can be increased by collapsing a set of stopping states into a single state in Definition 1. The MC stops when it either reaches K steps or one of the states in the stopping set. Corollary 1 describes how this collapsing can be performed while preserving the statistical properties of the MC.

Theorem 1 introduces the MCLV-K estimator (that, among others, can estimate the partition function) and proves it is consistent, giving error bounds. And Theorem 2 shows that this estimator is also finite-running-time unbiased. The first results provides unbiased estimates of the partition function, and generalize these unbiased estimates to a broad family of functions.

The reader only interested in RBM gradient estimates can safely skip to the next section on training RBMs, after reading the preliminaries and the definition of the RBM stopping sets.

Preliminaries. We define state $\mathbf{x} = (\mathbf{v}, \mathbf{h})$ to consist of a visible vector, $\mathbf{v} \in V$, and a hidden vector $\mathbf{h} \in H$, where H and V are the set of all hidden and visible states, respectively. Let $\Phi(\mathbf{W})$ be an irreducible Markov chain with steady state $p(\mathbf{x}; \mathbf{W})$ over the states $\Omega \coloneqq V \times H$. A MC is irreducible if all states communicate, which is trivially true for RBMs since the co-domain of the logistic function is $(0, 1)$ for any input in $\mathbb{R}$. If the Markov chain $\Phi(\mathbf{W})$ starts in equilibrium (or runs until equilibrium), the next transition gives us *one* independent sample $\mathbf{x}$ from the steady state $p(\mathbf{x}; \mathbf{W})$. The set $\{\mathbf{v}_n\}_{n=1}^N$ denotes the N visible examples of the training data. We often use $(\cdot)$, as in $g(\cdot)$, to denote that the statement over g is true for any valid input value.

RBMs can be trained by optimizing its parameters $\mathbf{W}$ in order to maximize the likelihood of the training data. Taking partial derivatives with respect to the weights results in a surprisingly simple update rule for $\mathbf{W}$:

$$\begin{aligned}
&\frac{1}{N} \sum_{n=1}^{N} \frac{\partial \log(\sum_{\mathbf{h}} p(\mathbf{x} = (\mathbf{v}_n, \mathbf{h}); \mathbf{W}))}{\partial \mathbf{W}} \\
&= \sum_{\mathbf{h} \in H} \Big(\frac{1}{N} \sum_{n=1}^{N} p(\mathbf{h}|\mathbf{v}_n; \mathbf{W}) \mathbf{v}_n \mathbf{h}^T \\
&\qquad - \sum_{\mathbf{v} \in V} p((\mathbf{v}, \mathbf{h}); \mathbf{W}) \mathbf{v} \mathbf{h}^T \Big) \\
&= \frac{1}{N} \sum_{n=1}^{N} \mathbf{v}_n \mathbb{E}_{\mathbf{W}}[\mathbf{h}|\mathbf{v}_n]^T - \mathbb{E}_{\mathbf{W}}[\mathbf{v}\mathbf{h}^T],
\end{aligned} \qquad (2)$$

where the l.h.s. term of eq. (2) (also called *positive statistics*) is easily calculated from the training data. However, the r.h.s. term of eq. (2) (*negative statistics*) corresponds to the gradient of the partition function $Z(\mathbf{W})$, which is generally intractable to compute. More specifically, computing $E[\mathbf{v}\mathbf{h}^T]$ requires collecting model statistics $p(\mathbf{v}, \mathbf{h})$, either by running the MCMC Markov chain $\Phi(\mathbf{W})$ to equilibrium from any starting state or by direct computation of the expected value if we know the partition function $Z(\mathbf{W})$.

If the Markov chain $\Phi(\mathbf{W})$ is not run until equilibrium the gradient estimates have an unknown bias. In what follows we use Markov chain tours to take care of this bias.

Tours and Stopping Sets. Define a *tour* to be a sequence of ξ steps of the Markov chain $(\mathbf{X}(1), \ldots, \mathbf{X}(\xi))$ s.t. the state of the $(\xi + 1)$-st step is the same as the starting state, i.e., $\mathbf{X}(1) = \mathbf{X}(\xi + 1)$. Let $\Xi^{(r)} = (\mathbf{X}^{(r)}(1), \ldots, \mathbf{X}^{(r)}(\xi^{(r)}))$ denote the r-th tour, $r \geq 1$. The strong Markov property guarantees that if $s \neq r$, the sequences $\Xi^{(r)}$ are independent of $\Xi^{(s)}$. This independence guarantees that both $\Xi^{(r)}$ and $\Xi^{(s)}$ are sample paths obtained from the equilibrium distribution of the Markov chain. We will later use this property to obtain unbiased estimators of the partition function.

However, as is, tours are not a practical concept for RBMs because in such a large state space Ω, the tour is unlikely to return to the same starting state. We will, however, use a

Markov chain property common to Metropolis-Hastings and Gibbs sampling Markov chains to significantly increase the probability of return by collapsing a large number of states into a single state.

RBM stopping set. Our stopping set $\mathcal{S}$ uses sampled hidden states from the training data, $\mathcal{H}_N^{(m)} = \{\mathbf{h}_n^{(1)}, \ldots, \mathbf{h}_n^{(m)} : \mathbf{h}_n \sim p(\mathbf{h}|\mathbf{v}_n; \mathbf{W})\}$, where $\{\mathbf{v}_n\}_{n=1}^N$ is the training data. Often we will use $m = 1$, but we can change the size of $\mathcal{H}_N^{(m)}$ by changing m. The stopping set contains all hidden states in $\mathcal{H}_N^{(m)}$ and all possible visible states

$$\mathcal{S}_{\text{HN}}^{(m)} = \bigcup_{\mathbf{h}\in\mathcal{H}_N^{(m)}, \mathbf{v}\in V} \{(\mathbf{v}, \mathbf{h})\}, \tag{3}$$

and $p(\mathbf{h}|\mathbf{v}; \mathbf{W})$ is the conditional probability of $\mathbf{h}$ given $\mathbf{v}$ using model parameters $\mathbf{W}$. Most of our theoretical results apply to any stopping set that is a proper subset of the state space, $\mathcal{S} \subset \Omega$. In practice, note that we *do not* store $\mathcal{S}_{\text{HN}}^{(m)}$ in memory, rather we *only keep* $\mathcal{H}_N^{(m)}$ *in memory*, as reaching a hidden state in $\mathcal{H}_N^{(m)}$ is enough to guarantee we need to stop. This requires only $O(mN)$ space, where N is the number of training observations.

Definition 1 (Stopping-set-Collapsed MC)**.** *Consider an arbitrary stopping set $\mathcal{S} \subset \Omega$. A state-collapsed MC is a transformation of MC $\Phi(\mathbf{W})$ with state space Ω, into a new MC $\Phi'(\mathbf{W})$ with state space $\Omega' = \Omega \backslash \mathcal{S} \cup \{\mathsf{S}\}$, where S is a new state formed by collapsing all the states in $\mathcal{S}$. The transition probabilities between states $\Omega' \cap \Omega$ are the same as in $\Phi(\mathbf{W})$. The transition probabilities from S to states in $\Omega' \backslash \{\mathsf{S}\}$ are*

$$p_{\Phi'}(\mathsf{S}, \mathbf{x}) = \frac{\sum_{\mathbf{y}\in\mathcal{S}} e^{-E(\mathbf{y};\mathbf{W})} p_\Phi(\mathbf{y}, \mathbf{x})}{Z_\mathcal{S}(\mathbf{W})}, \quad \forall \mathbf{x} \in \Omega' \backslash \{\mathsf{S}\},$$

where $Z_\mathcal{S}(\mathbf{W}) = \sum_{\mathbf{y}\in\mathcal{S}} e^{-E(\mathbf{y};\mathbf{W})}$, and p_a indicates the probability transition matrix of MC a.

The transitions from states $\Omega' \backslash \{\mathsf{S}\}$ to state S are

$$p_{\Phi'}(\mathbf{x}, \mathsf{S}) = \sum_{\mathbf{y}\in\mathcal{S}} p_\Phi(\mathbf{x}, \mathbf{y}), \quad \forall \mathbf{x} \in \Omega' \backslash \{\mathsf{S}\}.$$

It is important to distinguish the MC in Definition 1 from general MC state aggregation methods such as lumpability (Buchholz 1994) and interactive aggregation-disaggregation methods (Stewart 1994). In the following corollary, we see that the MC in Definition 1 affects the steady state, unlike general MC aggregation methods that leave the steady state undisturbed. Thankfully, later we will be able to correct the distortion imposed by Definition 1 because we know the steady state distribution of the states inside $\mathcal{S}$ up to a normalizing constant.

Corollary 1 (Simulating $\Phi'(\mathbf{W})$ from $\Phi(\mathbf{W})$)**.** *For any MC $\Phi(\mathbf{W})$ resulting from standard Gibbs sampling or Metropolis-Hastings (MH) MCMCs, we can cheaply simulate the transitions in and out of S of Definition 1 by: (a) $p_{\Phi'}(\mathsf{S}, \mathbf{x})$, we first sample a state $\mathbf{y}$ with replacement from $\mathcal{S}$ with probability $e^{-E(\mathbf{y};\mathbf{W})}/Z_\mathcal{S}(\mathbf{W})$ and then perform a transition $p_\Phi(\mathbf{y}, \mathbf{x})$; (b) $p_{\Phi'}(\mathbf{x}, \mathsf{S})$ is also simulated by performing a transition $p_\Phi(\mathbf{x}, \mathbf{y})$, and stopping the MC if $\mathbf{y} \in \mathcal{S}$. The simulated $\Phi'(\mathbf{W})$ is ergodic and time-reversible.*

The proof is in the appendix. It follows from the fact that $\Phi(\mathbf{W})$ is the MC of Gibbs sampling and MH and, thus, time-reversible (Aldous and Fill 2002). Time reversibility imposes a set of necessary and sufficient conditions in the form of detailed balance equations (Gallager 2013, Theorem 6.5.2). A little algebra shows that the sampling procedure in Corollary 1 using $\Phi(\mathbf{W})$ is stochastically equivalent to $\Phi'(\mathbf{W})$.

MCLV-K Estimator

Following Corollary 1, a tour starts by sampling the initial tour state $\mathbf{x}$ and stopping when the tour reaches the stopping set $\mathcal{S}$. We now want to truncate all return times of tours greater than some value $K \geq 1$, i.e., we will only observe the complete r-th tour $(\mathbf{x}, \mathbf{X}^{(r)}(2), \ldots, \mathbf{X}^{(r)}(\xi^{(r)}))$ if $\xi^{(r)} \leq K$. Otherwise, we observe only the first K states of the tour: $(\mathbf{x}, \mathbf{X}^{(r)}(2), \ldots, \mathbf{X}^{(r)}(K))$. The $\mathcal{S}$*-stopped* flag for tour r is **true** if $\xi^{(r)} \leq K$, otherwise it is **false**.

Lemma 1 (Perfect sampling of tours)**.** *Let*

$$\mathcal{C}_k = \{(\mathbf{x}, \mathbf{X}^{(i)}(2), \ldots, \mathbf{X}^{(i)}(k))\}_i$$

be a set of tours of length $k \leq K$, with $\mathbf{x}$ sampled from $\mathcal{S}$ according to some distribution.

Then, there exists a distribution G_k such that the random variables

$$\mathcal{G}_k \equiv \{g(\sigma) : \forall \sigma \in \mathcal{C}_k\}$$

are i.i.d. samples of G_k, with g defined over the appropriate σ-algebra (e.g., k RBM states) with $\|g(\cdot)\|_1 \leq \infty$.

Moreover, if we perform M tours, these tours finish in finite time and $\{\xi^{(r)}\}_{r=1}^M$ is an i.i.d. sequence with a well-defined probability distribution $p(\xi^{(\cdot)} = k)$.

The Las Vegas parallel is observed when we notice that any MCMC metric can be perfectly sampled from the tours. The tour lengths are sampled from a distribution $p(\xi^{(\cdot)} = k)$. And, for any given tour length k, the metric of interest g is perfectly sampled from G_k. The maximum tour length K only cuts off the tail of $p(\xi^{(\cdot)} = k)$ beyond $k > K$, which allows us to bound the sampling error.

Theorem 1 (MCLV-K RBM Estimator)**.** *Let $p(\mathbf{x}; \mathbf{W})$, $E(\mathbf{x}; \mathbf{W})$, and $Z(\mathbf{W})$ be as described in eq.*(1)*. Let*

$$F(\mathbf{W}, f) = Z(\mathbf{W}) \sum_{\mathbf{x}\in\Omega} f(\mathbf{x}) p(\mathbf{x}; \mathbf{W}), \tag{4}$$

where $f : \Omega \to \mathbb{R}^n$, $n \geq 1$, $\|f(\cdot)\|_1 < \infty$, and $\|\cdot\|_1$ is the l_1 norm. Let $\Phi(\mathbf{W})$ be a time-reversible MC with state space Ω and steady state distribution $\{p(\mathbf{x}; \mathbf{W})\}_{\mathbf{x}\in\Omega}$. Let $\mathcal{S} \subset \Omega$ be a proper subset of the states of $\Phi(\mathbf{W})$.

Sample $\mathbf{x}' \in \mathcal{S}$ with probability $e^{-E(\mathbf{x}';\mathbf{W})}/Z_\mathcal{S}(\mathbf{W})$ and let $(\mathbf{X}^{(r)}(1) = \mathbf{x}', \mathbf{X}^{(r)}(2), \ldots, \mathbf{X}^{(r)}(\xi^{(r)}))$ be a sequence of discrete states of the r-th $\mathcal{S}$-stopped tour, where we stop the tour if one of two conditions are met: (a) we have reached K steps, or (b) when we reach any state in $\mathcal{S}$, i.e., $\mathbf{X}^{(r)}(\xi + 1) \in \mathcal{S}$. Then, for $R \geq 1$ tours, let $\mathcal{C}_k^{(R)}$ be the set of finished tours in $k \leq K$ steps, (as defined in Corollary 1). For the sake *of simplicity, we henceforth refer to $\mathcal{C}_k^{(R)}$ simply as $\mathcal{C}_k$. The estimator*

$$\hat{F}^{(K,R)}(\mathbf{W}, f) = \frac{1}{\sum_{k=1}^{K} |\mathcal{C}_k|} \sum_{\mathbf{y} \in \mathcal{S}} e^{-E(\mathbf{y};\mathbf{W})} \times \sum_{k=1}^{K} \sum_{(\mathbf{X}(1),\mathbf{X}(2),\ldots,\mathbf{X}(k)) \in \mathcal{C}_k} \sum_{h=1}^{k} f(\mathbf{X}(h)) \tag{5}$$

is an estimate of $F(\mathbf{W}, f)$ in eq. (4) *with a bias upper bounded by $B \cdot (E[\xi] - \sum_{k=1}^{K-1} p(\xi > k))$, where $p(\xi > k)$ is the probability that a tour has length greater than k and $B \geq \sup_{\mathbf{x} \in \Omega} \|f(\mathbf{x})\|_1$.*

Theorem 1 gives a basic estimator from the MCMC tours. The gradient estimates will be explicitly derived in the next section. In our experiments we show how to estimate $p(\xi > k)$. For the partition function and gradient estimates, it is also trivial to obtain a bound on B using the RBM weights $\mathbf{W}$ (Bengio and Delalleau 2009).

Theorem 2 (Geometrically Decaying Tour Length Tails)**.** *Let $p(\xi > k)$ be the probability that a tour has length greater than k. If there exists a constant $\epsilon > 0$ s.t. $\inf_{\mathbf{x} \in \Omega \backslash \mathcal{S}} \sum_{\mathbf{y} \in \mathcal{S}} p_\Phi(\mathbf{x}, \mathbf{y}) \geq \epsilon$ then, there exists $0 < \alpha < 1$, $\log p(\xi > k) = k \log \alpha + o(k)$, i.e., ξ has a geometrically decaying tail.*

Theorem 2 shows conditions of a geometric decay in the tail of $p(\xi > k)$. And in practice it means that tours cannot be “heavy tail” long and, thus, making the bound in Theorem 1 tighter.

MCLV-K Finite-Sample Unbiasedness

In what follows we dynamically increase K until the MC reaches a state in the stopping set.

The following theorem shows that this procedure gives unbiased estimates of $F(\mathbf{W}, f)$.

Theorem 3 (Unbiased Partition-scaled Function Estimates by Dynamic Adaptation of K)**.** *Consider the estimator in Theorem 1 and let us dynamically grow K (denoted K_{dyn}) until the MC reaches a stopping state in $\mathcal{S}$. Then, for $R \geq 1$ tours,*

$$\mathbb{E}[\hat{F}^{(K_{dyn},R)}(\mathbf{W}, f)] = F(\mathbf{W}, f), \tag{6}$$

is an **unbiased estimator** *and the estimator is consistent, i.e., almost surely $\lim_{R \to \infty} \hat{F}^{(K_{dyn},R)}(\mathbf{W}, f) = F(\mathbf{W}, f)$, and K_{dyn} is finite.*

Moreover, for $\epsilon > 0$,

$$p\left(\left\|\hat{F}^{(K_{dyn},R)}(\mathbf{W}, f) - F(\mathbf{W}, f)\right\|_1 \geq \epsilon\right) \leq \alpha_{R,Z_\mathcal{S}(\mathbf{W})},$$

where, R is the number of tours, $\alpha_{R,Z_\mathcal{S}(\mathbf{W})} = \frac{B^2}{\epsilon^2 R}\left(\frac{(Z(\mathbf{W}))^2}{(Z_\mathcal{S}(\mathbf{W}))^2 \delta} + 1\right)$, $B \geq \sup_{\mathbf{x} \in \Omega} \|f(\mathbf{x})\|_1$ is an upper bound on the absolute value of $f(\cdot)$ over the state space Ω, δ is the spectral gap of the transition probability matrix of $\Phi(\mathbf{W})$.

Corollary 2 (Unbiased Partition Function Estimation)**.** *Let $f_1(x) = 1$, then*

$$\mathbb{E}[\hat{F}^{(K_{dyn},R)}(\mathbf{W}, f_1)] = Z(\mathbf{W}),$$

is an unbiased estimator of the partition function.

Training Restricted Boltzmann Machines

In what follows we explore the connections between MCLV-K and learning RBMs using MCMC methods. First, we show how MCLV-K can provide a finite-sample unbiased and asymptotically consistent estimate of the direction of the RBM gradient.

MCLV-K Gradient Estimates

In what follows we will provide an estimate of the gradient of the negative log-likelihood of RBMs using MCLV-K. Our gradient will have a scaling factor but the gradient direction is the same as the original gradient:

$$\nabla_\mathbf{W} \mathcal{L}_Z = \frac{Z(\mathbf{W})}{Z_\mathcal{S}(\mathbf{W})} \left(\frac{1}{N} \sum_{n=1}^{N} \mathbf{v}_n^\mathsf{T} \mathbb{E}_\mathbf{W}[\mathbf{h}|\mathbf{v}_n] - \mathbb{E}_\mathbf{W}[\mathbf{v}^\mathsf{T}\mathbf{h}] \right).$$

The scaling $Z(\mathbf{W})/Z_\mathcal{S}(\mathbf{W})$ is constant given $\mathbf{W}$. In our current implementation, we use Corollary 2 to estimate $Z(\mathbf{W})/Z_\mathcal{S}(\mathbf{W})$ and divide the gradient by it, compensating for the scaling at essentially no computational or memory cost.

Corollary 3 (LVS-K: The *Las Vegas Slope* Estimator)**.** *Let $\Phi(\mathbf{W})$, $\mathcal{S}$, $\mathbf{x}$, the tour $(\mathbf{x}, \mathbf{X}^{(r)}(2), \ldots, \mathbf{X}^{(r)}(\xi^{(r)}))$, R, K, and $\mathcal{C}_k$ be as defined in Theorem 1. Then, for a learning rate $\eta > 0$,*

$$\widehat{\nabla_\mathbf{W} \mathcal{L}_{LVS}}(K, R) = \eta \left(\frac{\widehat{\mathbb{E}}[\xi]}{N} \sum_{n=1}^{N} \frac{\partial E(\mathbf{x}_n; \mathbf{W})}{\partial \mathbf{W}} - \frac{\sum_{k=1}^{K} \sum_{(\mathbf{X}(1),\ldots,\mathbf{X}(k)) \in \mathcal{C}_k} \sum_{i=1}^{k} \frac{\partial E(\mathbf{X}^{(r)}(k);\mathbf{W})}{\partial \mathbf{W}}}{\sum_{k=1}^{K} |\mathcal{C}_k|} \right), \tag{7}$$

is a consistent $(K, R \to \infty)$ estimator of the energy-model gradient in eq.(2)*, where $\widehat{\mathbb{E}}[\xi] = \frac{\sum_{k=1}^{K} |\mathcal{C}_k| k}{\sum_{k=1}^{K} |\mathcal{C}_k|}$ is the empirical expectation of the tour lengths.*

Moreover, the contribution of a tour of length k to the negative statistics of the gradient is proportional to

$$P[\xi = k] \cdot k \cdot \mathbb{E}[\partial E(\tilde{\mathbf{X}}_k; \mathbf{W})/\partial \mathbf{W}],$$

where $\tilde{\mathbf{X}}_k$ is a random state of a tour of length k. If the Markov chain $\Phi(\mathbf{W})$ satisfies the conditions of Theorem 2, then $P[\xi = k] \cdot k = e^{-O(k)}$, so that extremely long *tours do not influence the gradient.*

Corollary 4 (Unbiased Gradient Direction Estimator: LVS-K_{dyn})**.** *Consider the estimator in Corollary 3 and let us dynamically grow K (denoted K_{dyn}) until the MC reaches a stopping state in $\mathcal{S}$. Then,*

$$\mathbb{E}\left[\widehat{\nabla_\mathbf{W} \mathcal{L}_{LVS}}(K_{dyn}, R)\right] \propto \nabla_\mathbf{W} \mathcal{L}_Z,$$

is an unbiased estimate of the RBM gradient direction.

The proofs of the two above corollaries follow directly from Theorems 1 and 3, respectively.

Correspondence and Differences Between LVS-K and CD-K

In this section we explore a correspondence between LVS-K (proposed in Corollary 3) and CD-K to train RBMs. We will also emphasize some differences that will give us some new insights into CD-K. The correspondence is as follows: **(a)** consider a mini-batch of training examples $\{\mathbf{v}_i\}_{i=1}^{N}$; **(b)** the stopping set is $\mathcal{S}_{HN}^{(m)}$, described in eq.(3); **(c)** the number of tours R of LVS-K is the number of training examples in the mini-batch N, i.e., $R = N$.

One can readily verify that the Gibbs sampling updates of LVS-K and CD-K are similar except for the following key differences: **(i)** LVS-K starts at a state $\mathbf{x}$ of $\mathcal{S}_{HN}^{(m)}$ with probability proportional to $\exp(-E(\mathbf{x};\mathbf{W}))$, CD-$K$ starts uniformly over the training examples. Thus, the negative phase of LVS-K tends to push the model away from unbalanced probabilities over the training examples. **(ii)** at every Markov chain step, LVS-K stops early if it has reached a state in $\mathcal{S}_{HN}^{(m)}$, while CD-K will always perform all K steps. **(iii)** the gradient estimates of LVS-K use only the completed tours, while CD-K uses all tours; **(iv)** the gradient estimates of LVS-K use all states visited by the MC during a tour, while CD-K uses only the last visited state.

A long sequence of states visited by the CD-K Gibbs sampler can be broken up into tours if the stopping state contains only the starting state. Figure 1 illustrates three MCMC runs starting at visible states representing "7", "3", and "4", broken up into tours whenever the starting hidden state is sampled again. Starting from visible state "7", CD-K ignores the completed tour *Tour 1*, which LVS-K uses for its gradient estimate; and CD-K proceeds to use the state in the middle of *Tour A* for its gradient estimate. CD-K also uses a state in the incomplete *Tour 2*, which LVS-K ignores as incomplete. Finally, CD-K ignores *Tour 3* and proceeds to use the state in the beginning of *Tour B* for its gradient estimate.

This means that, for $K \geq 2$, CD-K is more likely to sample states from longer tour than shorter tours. This bias is the inspection paradox (Wilson 1983). Interestingly, this bias makes CD-K, $K \geq 2$, significantly different from CD-1, which has no such bias. Note that LVS-K has the opposite bias: it ignores tours longer than K; the bias of LVS is measurable (Theorem 1) if we can estimate the average tour length.

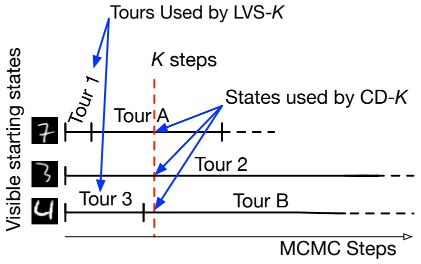


Figure 1: CD-k bias towards longer tours for $k \geq 2$;

Computational Complexity

In this section we give the time and space complexities of LVS-K and CD-K. Let $|\mathbf{W}|$ denote the number of elements in $\mathbf{W}$, $= n_V * n_H$ and $n_X = n_V + n_H$. In terms of space, LVS-K needs $O(N\, m\, n_X)$ space to store the $\mathcal{H}_N^{(m)}$ which is m times the requirement for CD-K. At every epoch, LVS-K samples a stopping set, which involves a matrix multiplication followed by algebraic operations over a matrix. The matrix multiplication which takes $O(Nm|\mathbf{W}|)$ upper bounds the time. Computing the free energies of the hidden state also takes the same time. Adding the states of the stopping set to a heap for easier sampling takes $O(Nm)$ time and allows us to sample starting states for the tours in $O(N\log(Nm))$. Every Gibbs step is again bounded by the time taken for matrix multiplication which takes a total of $O(N|\mathbf{W}|K)$ time. Checking stopping set membership takes $O(NKn_X)$ amortized time assuming that standard algorithms used by hash sets, e.g. MD5, take $O(n_X)$ time to evaluate. Computing the gradient and updating $\mathbf{W}$ takes $O(N|\mathbf{W}|)$ time.

Therefore LVS-K takes $O(NKn_X + Nm|\mathbf{W}| + NK|\mathbf{W}|) \equiv O(N|\mathbf{W}|(m+K))$ time, compared to CD-K which takes $O(NK|\mathbf{W}|)$. In the general case $m \in O(K)$, $\therefore$ the asymptotic complexity of CD-K and LVS-K are the same.

Related Work

AIS, MC changes, and CD-K extensions. Annealed Importance Sampling (AIS) (Neal 2005; 2001) uses two distinct Markov transitions kernels and has been applied by Salakhutdinov and Murray (Salakhutdinov and Murray 2008) to obtain unbiased estimates of the partition function of an RBM. Like AIS, the Russian roulette pseudo-marginal likelihood is also a Markov chain modification to sample from the steady state distribution (Lyne et al. 2015). These modifications cannot be readily applied to the original RBM Markov chain, nor they provide insights into the learning process. MCLV-K is a new tool that can be used from visual inspection of convergence to proposing new gradient estimators, as seen in our empirical results.

RBMs are powerful models (Montúfar and Morton 2015) and the analysis of CD-K has a long history (Hinton 2012). A few past studies have focused on how CD-K learns well RBMs (Carreira-Perpiñán and Hinton 2005; Yuille 2005), have some fixable issues learning RBMs (Schulz, Müller, and Behnke 2010; Fischer and Igel 2014; Prats, Romero Merino, and Castrillejo 2014), may approximate some objective function (Hinton 2002), or do not approximate any objective function (Bengio and Delalleau 2009; Sutskever and Tieleman 2010). Orthogonally, Persistent Contrastive Divergence (PCD) (Tieleman 2008) improves CD-K in some problems by using the starting state of the CD-K Markov chain as the end state of the previous training epoch (simulating a single sample path, assuming the MC does not change much between epochs, which is not always true (Schulz, Müller, and Behnke 2010)). Clearly, PCD could be adapted as a MCLV method, which we see as future work.

The presence of training data is key to the practicality of MCLV. Without training data, obtaining error bounds with

MCLV can be prohibitively expensive. In the worst-case, there is no polynomial time algorithm that can estimate the probabilities of an RBM model within a constant factor (Long and Servedio 2010), assuming P≠NP. But most real-world machine learning problems are *supposed* to be much easier than general MCMC results would have us believe. We are *given* a good hint of what should be a larger number of high-probability states in the steady state: the states containing the training examples. Unfortunately, vanilla MCMC methods do not incorporate this extra information to speed up convergence in a principled way. We believe the lessons learned in this paper will be invaluable to design new classes of Markov chain methods tailored to machine learning applications.

Las Vegas algorithms for Markov chain sampling. Perfect Sampling (Corcoran and Tweedie 2002; Fill 1997; Fill et al. 2000; Propp and Wilson 1996; 1998; Wilson 2000) is an example of a Las Vegas algorithm for MCMC applications. Unfortunately, energy-based models can easily reach trillions of states while perfect sampling methods rarely scale well unless some specific MC structure can be exploited. We are unaware of clever CFTP constructions for arbitrary energy-based models.

Mykland et al. (Mykland, Tierney, and Yu 1995) with a few follow-up works first proposed the use of regeneration in the context of MCMC to estimate mixing times, however these techniques are mostly of theoretical interest (Baxendale 2005; Gilks, Roberts, and Sahu 1998; Hobert et al. 2002; Roberts and Tweedie 1999) rather than of practical utility for energy-based models. Path coupling is another alternative to estimate mixing times (Bubley and Dyer 1997). More recently, path coupling was used to develop a theory of Ricci curvature for Markov chains (Ollivier 2009). The connections between Ricci curvature estimation and MCLV-K are worth exploring in future work.

Empirical Results

Our experiments use the MNIST dataset, which consists of 70,000 images of digits ranging from 0 to 9, each image having 28×28 pixels (a total of 784 pixels per image), divided into 55,000 training examples and 15,000 test examples. We use this dataset for historical reasons. MNIST is arguably the most extensively studied dataset in RBM training, e.g. (Hinton 2002; 2012; Hinton, Osindero, and Teh 2006; Carreira-Perpiñán and Hinton 2005; Salakhutdinov and Hinton 2009; Tieleman 2008; Bengio and Delalleau 2009). Our goal is to show that MCLV-K is able to give new insights into RBM training (and improved performance) even in a studied-to-death dataset such as MNIST. The experimental details of our empirical results are presented in the appendix. We use LVS-1 to train the RBM model used in the following experiments (CD-K tends to give very high probability to a few examples in the training data). We observe little difference between LVS-1, LVS-3, and LVS-10 (for reasons that will be clear soon).

RBM learning. Our first set of empirical results compares LVS-K, $K \in \{1, 3, 10\}$, CD-K, $K \in \{1, 10\}$ and PCD-K, $K \in \{1, 10\}$ by training an RBM using stochastic gradient descent, where the gradient estimates are computed using the respective methods. We train RBMs with $n_H = 32$ hidden neurons for a total of 100 epochs (inclusive of 15 warm-up epochs of CD-1 for LVS-K), using a learning rate of 0.1 which decays according to a Robbins-Monro schedule. Weight decay and momentum were not used. The initial $\mathbf{W}$ weights are sampled uniformly from $U\left(\frac{-0.1}{\sqrt{n_V+n_H}}, \frac{0.1}{\sqrt{n_V+n_H}}\right)$, where n_V and n_H denote the number of visible and hidden neurons, respectively. Hidden biases are initialized as zero, while visible biases are initialized as $\log(p_i/(1-p_i))$ (Hinton 2012), where p_i is the empirical probability of feature i being active.

The small number of hidden units is to enable us to evaluate the true performance: we compute the exact partition function of the trained RBM and calculate the average log-likelihood $\frac{1}{N}\sum_{n=1}^{N} \log p(\mathbf{v}_n)$. All results are means calculated from 10 executions. In all LVS-K experiments we use $m = 1$, for simplicity. The negative log-likelihood of LVS-K, PCD-K and CD-K are presented in Table 1.

Subsequently, in order to compare our results with those presented in Tieleman (2008), we train RBMs with $n_H = 25$ and initial learning rates between 10^{-4} and 1. We observe that larger learning rates (10^{-1} to 1) are more appropriate for LVS-K, resulting in faster convergence and increased performance. Small rates (e.g. 10^{-4}) cause tours to rarely finish, severely slowing down the training. On the other hand, CD-K and PCD-K fail to converge with learning rates slightly larger than 10^{-2}. The results for this experiment, along with the best learning rates for each method, are presented in Table 2.

In conclusion, LVS-K drastically (and paired t-test significantly) outperforms CD-K and PCD-K w.r.t. the log-likelihood in all settings, even LVS-1 performs significantly better than PCD-10. However we were unable to reproduce the likelihood of ≈ -130 for PCD achieved by Tieleman (2008).

Method	**Training**	**Testing**
CD-1	-167.3 (2.7)	-166.6 (2.8)
CD-10	-154.3 (3.3)	-153.4 (3.3)
PCD-1	-153.0 (4.9)	-152.1 (4.7)
PCD-10	-139.3 (3.2)	-138.5 (3.3)
LVS-1	**-134.0 (1.0)**	**-133.3 (1.0)**
LVS-10	**-133.3 (1.0)**	**-132.6 (1.0)**
LVS-3	**-133.7 (0.8)**	**-132.9 (0.7)**

Table 1: (Higher is better) Average log-likelihood on the MNIST dataset using a RBM with 32 hidden neurons. Results are means over 10 executions after 100 epochs.

Tours lengths and stopping state. We now analyze the tour lengths as a function of: (a) n_H, the number of hidden units, and (b) the size of the stopping set $|\mathcal{S}_{NH}^{(m)}|$, where $\mathcal{S}_{NH}^{(m)}$ is built from the training data as defined in eq.(3). Note that the r-th tour ends at state $\mathbf{X}^{(r)}(\xi) = \left(\mathbf{v}^{(r)}(\xi), \mathbf{h}^{(r)}(\xi)\right)$ whenever $\mathbf{X}^{(r)}(\xi + 1) \in \mathcal{S}_{NH}$, and that the stopping criteria

Method	Learning Rate	Training	Testing
CD-1	0.01	-169.8 (2.6)	-169.0 (2.6)
CD-10	0.01	-156.4 (0.5)	-155.6 (0.5)
PCD-1	0.01	-147.8 (0.5)	-147.0 (0.5)
PCD-10	0.01	-147.4 (0.5)	-146.7 (0.5)
LVS-1	**0.1**	**-138.3 (1.3)**	**-137.5 (1.4)**
LVS-10	**0.1**	**-138.1 (1.1)**	**-137.4 (1.2)**
LVS-3	**0.1**	**-138.2 (1.0)**	**-137.5 (1.1)**

Table 2: (Higher is better) Average log-likelihood on the MNIST dataset using a RBM with 25 hidden neurons. Results are means over 10 executions after 100 epochs, using appropriate learning rates for each method.

only truly depends on $\mathbf{h}^{(r)}(\xi + 1)$ since $\mathcal{S}_{NH}^{(m)}$ contains all possible visible states.

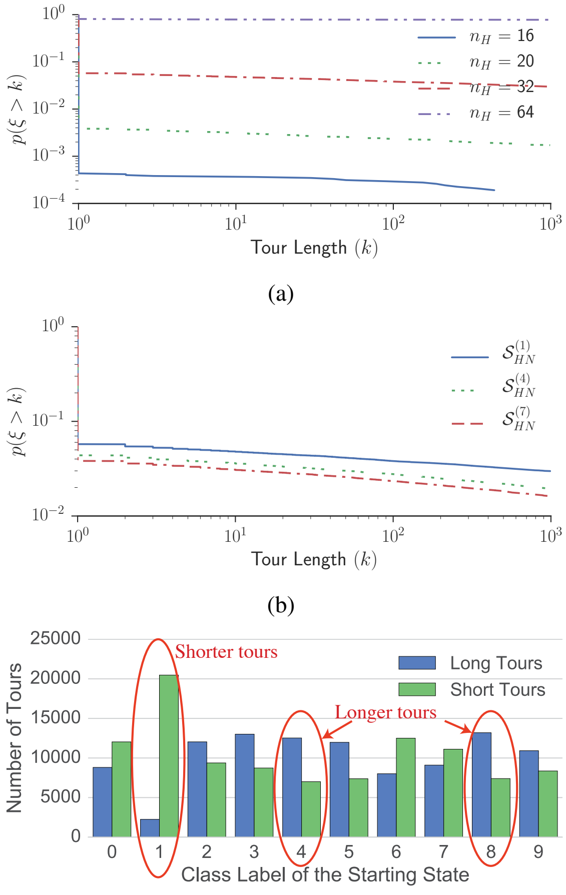
Figure 2a shows the CCDF of the tour lengths for different values of n_H. Most tours are extremely short for RBMs with few hidden neurons (for $n_H = 16$, more than 99% have length one), but significantly increase as we increase n_H with a very heavy tail. Thus, it is expected that we see little difference between LVS-1, LVS-3, and LVS-10. Moreover, these heavy tails likely causes strong inspection-paradox biases for CD-K in high-dimensional RBMs.

Most importantly, Figure 2a shows that tours either return within one step or are unlikely to return for a very long time. A closer inspection at these one-step tours, shows that over 99% of the cases have the hidden state being the starting state. *Thus, it seems that RBMs (even with few hidden neurons) are just memorizing the training data, not learning how to generate new digits.* We conjecture, however, that if our training could force the tours to stop at distinct hidden states, and requires the tours to be possibly longer (but not too-long), the RBM might be taught how to generate new digits.

Using $n_H = 32$ hidden neurons, Figure 2b shows the probability that a tour takes more than k steps, as we increase the number of stopping states by setting the values of $m \in \{1, 4, 7\}$ in $\mathcal{S}_{NH}^{(m)}$. We see that the probability of tours finishing in a single step increases as we add more states to the stopping state. Thus, increasing the stopping set size can significantly shorten the tours, which in turn improves the estimates of MCLV-K and LVS-K, and is an avenue to ameliorate MCMC issues in high-dimensional RBMs.

Distribution modes of the learned RBM. Overall, we may want to ask which digits (pictures) the model is learning to reproduce well.

Figure 2c shows the length of the tours split by the type of digit starting the tour. Note that the RBM seems to learn well digits that are more consistent across the training data (e.g., numbers one and six) and have more trouble with digits that have more variability (e.g., numbers four and eight).



(a)

(b)

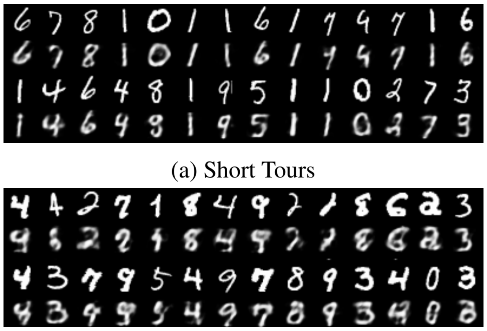
(c)

Figure 2: (a) Tour lengths CCDF for $n_H = \log_2 |H| \in \{16, 20, 32, 64\}$ for LVS-1; (b) Tour lengths CCDF variation for LVS-10 with $n_H = 32$, using larger Stopping Sets; (c) Comparison of frequencies of short and long tours starting from labeled states on a trained RBM

As a visual inspection, Figure 3 shows the next visible states of extremely short (length = 1) and long (unfinished after 99,999 steps) tours, for $n_H = 32$ hidden neurons. There is a clear relation between long tours and not-so-common examples in the training data. The first and third rows show the training examples; the next row shows their first visible state after one Gibbs sampling step. Note that the majority of the training examples are easy-to-recognize digits, with still recognizable digits after sampling.

The second part of Figure 3 shows the training example and first visible samples of long tours. Note that the long tours tend to be digits that are either thicker (rarer in the data), or come in a not-so-standard shape than the digits in the first row. Note that in half of the examples, their first Gibbs samples are not too similar to the original digit. This shows that the model is having trouble learning these less standard digits. That long tours tend to start in odd-looking-examples, should help us better understand and avoid *fantasy particles* (visible states $\mathbf{v} \in V$ that are not characteristic of the dataset but have high probability nonetheless (Tieleman 2008)).

Estimating the partition function. We use MCLV-K_{dyn} to estimate the partition function $Z(\mathbf{W}) =$



(a) Short Tours

(b) Long Tours

Figure 3: Visible states of tours for $n_H = 32$ neurons. The first and third rows of each image show the visible states from the training data, whereas the second and fourth show the next visible state obtained through Gibbs Sampling

$\sum_{\mathbf{v}}\sum_{\mathbf{h}} e^{-E((\mathbf{v},\mathbf{h});\mathbf{W})}$ as specified in Corollary 2 using an RBM with $n_H = 32$, so that we can easily compute the true partition function for comparison. We note that computing $Z_{\mathcal{S}}(\mathbf{W})$ is fast as it is in the order of the number of training examples (as stated earlier).

Following Corollary 2, we estimate $Z(\mathbf{W})$ with $\hat{F}^{(K_{\text{dyn}},R)}(\mathbf{W}, f_1)$, with $f_1(x) = 1$. The average tour length in this example is estimated to be close to one (see Figure 1). Thus, $\hat{F}^{(K_{\text{dyn}},R)}(\mathbf{W}, f_1) \approx Z_{\mathcal{S}}(\mathbf{W}) = 1.46 \times 10^{100}$ in this example. In fact, $\hat{F}^{(K_{\text{dyn}},R)}(\mathbf{W}, f_1)$ and the true partition function $Z(\mathbf{W})$ report the same value up to nearly machine precision (10-th decimal place).

Conclusions

This paper proposes a Las Vegas transformation of Markov Chain Monte Carlo (MCMC) for RBMs, denoted *Markov Chain Las Vegas* (MCLV). MCLV gives statistical guarantees in exchange for random running times. Our empirical results show MCLV-K is a powerful tool to learn and understand RBMs, with a gradient estimator LVS-K that can better fit RBMs to the MNIST dataset than standard MCMC methods such as Contrastive Divergence (CD-K).

Acknowledgments

This work was partially sponsored by the ARO, under the U.S. Army Research Laboratory contract number W911NF-09-2-0053. The authors would like to also thank NVidia for their kind hardware support.

Appendix

Proof of Corollary 1

Proof. Collapse the states of $\mathcal{S}$ into a single state $\mathscr{S}$ to form a state-collapsed MC $\Phi'(\mathbf{W})$, with transition probabilities given by Definition 1. Let $(\mathscr{S}, \mathbf{X}^{(\cdot)}(2), \dots, \mathbf{X}^{(\cdot)}(\xi^{(\cdot)}))$ be a sequence of discrete states of the r-th tour of the state-collapsed MC $\Phi'(\mathbf{W})$. Note that $\mathscr{S}$ is the renewal state of the tour $\Xi^{(\cdot)}$, i.e., $\mathbf{X}^{(\cdot)}(1) = \mathscr{S}$.

The time reversibility of $\Phi(\mathbf{W})$ implies that $p(\mathbf{x};\mathbf{W})p_\Phi(\mathbf{x},\mathbf{y}) = p(\mathbf{y};\mathbf{W})p_\Phi(\mathbf{y},\mathbf{x})$, where p_a indicates the probability transition matrix of MC a. Let $Z_{\mathcal{S}}(\mathbf{W}) = \sum_{\mathbf{y}\in\mathcal{S}} e^{-E(\mathbf{y};\mathbf{W})}$. We now show that $\Phi'(\mathbf{W})$ is time-reversible using the fact that the steady state distribution of $\Phi(\mathbf{W})$ is known up to a constant factor. Thus, we "guess" the steady state distribution in $\Phi'(\mathbf{W})$ of $\mathscr{S}$ as $p(\mathscr{S};\mathbf{W}) = Z_{\mathcal{S}}(\mathbf{W})/Z(\mathbf{W})$ and verify that, because $\mathcal{S}$ is a proper subset of Ω, the balance equations of $\Phi'(\mathbf{W})$ are time reversible:

$$\begin{aligned}
p(\mathscr{S};\mathbf{W})p_{\Phi'}(\mathscr{S},\mathbf{x}) &:= \frac{Z_{\mathcal{S}}(\mathbf{W})}{Z(\mathbf{W})}\sum_{\mathbf{y}\in\mathcal{S}}\frac{e^{-E(\mathbf{y};\mathbf{W})}}{Z_{\mathcal{S}}(\mathbf{W})}p_\Phi(\mathbf{y},\mathbf{x}) \\
&= \sum_{\mathbf{y}\in\mathcal{S}} p(\mathbf{y};\mathbf{W})p_\Phi(\mathbf{y},\mathbf{x}) \\
&= \sum_{\mathbf{y}\in\mathcal{S}} p(\mathbf{x};\mathbf{W})p_\Phi(\mathbf{x},\mathbf{y}) \qquad \text{see}^\dagger \\
&= p(\mathbf{x};\mathbf{W})p_{\Phi'}(\mathbf{x},\mathscr{S}),
\end{aligned}$$

†from the time reversibility of $\Phi(\mathbf{W})$. Thus, all states $\mathbf{x} \in \Omega'\backslash\{\mathscr{S}\}$ in $\Phi'(\mathbf{W})$ have the same steady state distribution as in $\Phi(\mathbf{W})$: $p(\mathbf{x};\mathbf{W})$. □

Proof of Lemma 1

Lemma (Perfect sampling of tours)**.** *Let*

$$\mathcal{C}_k = \{(\mathbf{x}, \mathbf{X}^{(i)}(2), \dots, \mathbf{X}^{(i)}(k))\}_i$$

be a set of tours of length $k \le K$, with $\mathbf{x}$ sampled from $\mathcal{S}$ according to some distribution.

Then, there exists a distribution G_k such that the random variables

$$\mathcal{G}_k := \{g(\sigma) : \forall\sigma \in \mathcal{C}_k\} \tag{8}$$

are i.i.d. samples of G_k, with g defined over the appropriate σ-algebra (e.g., k RBM states) with $\|g(\cdot)\|_1 \le \infty$.

Moreover, if we perform M tours, these tours finish in finite time and $\{\xi^{(r)}\}_{r=1}^M$ is an i.i.d. sequence with a well-defined probability distribution $p(\xi^{(\cdot)} = k)$.

Proof. Consider an infinite run of the MCMC $\Phi'(\mathbf{W})$: $\mathbf{X}(1), \mathbf{X}(2), \dots$, starting at state $\mathbf{X}(1) = \mathscr{S}$. Divide this infinite run into tours, the longest segments of consecutive states that start at state $\mathscr{S}$ but do not contain $\mathscr{S}$ in any other states in the segment. Let $\xi^{(r)}$ be the length of the r-th tour. Because $\Phi'(\mathbf{W})$ is an irreducible Markov chain, it is positive recurrent (Gallager 2013, Theorem 6.3.8), and we can use Kac's theorem (Meyn and Tweedie 2012, Theorem 10.2.2) to assert that $\mathbb{E}[\xi^{(\cdot)}] < \infty$, which also implies $\xi^{(\cdot)} < \infty$ almost surely (i.e., except for a set of measure zero). Define $R_{r+1} = R_r + \xi^{(r+1)}$, with $R_0 = 0$. Define

$$\begin{aligned}
\mathcal{G}_k = \{g(\mathbf{X}(R_{r-1}), \dots, \mathbf{X}(R_r - 1)) : \\
r = 1, \dots, M, \xi^{(r)} = k\},
\end{aligned}$$

with $M > 1$. By the strong Markov property, there exists a distribution G_k such that $\mathcal{G}_k$ is an iid sequence from G_k. Note that $\{\xi^{(r)}\}_{r=1}^M$ is also iid. Further, note that by Corollary 1 we can equivalently consider the MC $\Phi(\mathbf{W})$, starting at state $\mathbf{x}$ sampled from the stopping set $\mathcal{S}$, which concludes the proof. □

Proof of Theorems 1 and 3

Proof. For simplicity, in what follows we combine the proofs of Theorems 1 and 3, specializing on each case when necessary. Define for all $r \geq 0$, $R_{r+1} = R_r + \xi^{(r+1)}$ and for $t \geq 0$, $N(t) = \operatorname{argmax}_r \mathbf{1}_{\{R_{r-1} < t\}}$, with $R_0 = 0$. $N(t)$ counts how many of the tours in the sequence $\{\xi^{(r)}\}_{r \geq 1}$ are needed to add up to the largest number smaller than t. Let

$$Y_K^{(r)} = \mathbf{1}\{\xi^{(r)} \leq K\} \sum_{t=R_{r-1}+1}^{R_r} f(\mathbf{X}^{(r)}(t - R_{r-1})).$$

By Lemma 1, both $\{Y_K^{(r)}\}_{r \geq 1}$ and $\{\xi^{(r)}\}_{r \geq 1}$ are iid sequences. Also, even in the case $K \to \infty$,

$$\mathbb{E}[\|Y_K^{(\cdot)}\|_1] \leq \mathbb{E}[\sup_{\mathbf{x}} \xi^{(\cdot)} \|f(\mathbf{x})\|_1] = \mathbb{E}[\xi^{(\cdot)}] \sup_{\mathbf{x}} \|f(\mathbf{x})\|_1 < \infty,$$

as by definition $\|f(\cdot)\|_1 < \infty$ and we know $\mathbb{E}[\xi^{(\cdot)}] < \infty$ (see Lemma 1). The Renewal-Reward Theorem (Brémaud 2013, Theorem 4.2) yields, $r \geq 1$

$$\lim_{t \to \infty} \frac{\sum_{r=1}^{N(t)} Y_K^{(r)}}{t} = \frac{\mathbb{E}\left[\mathbf{1}_{\{\xi^{(\cdot)} \leq K\}} \sum_{k=1}^{\xi^{(\cdot)}} f(\mathbf{X}^{(\cdot)}(k))\right]}{\mathbb{E}[\xi^{(\cdot)}]}. \tag{9}$$

Note that,

$$\frac{\sum_{r=1}^{N(t)} Y_K^{(r)}}{t} = \frac{\mathbf{1}_{\{\xi^{(N(t'))} \leq K\}} \sum_{t'=1}^{R_{N(t)}} f(\mathbf{X}^{(N(t'))}(t' - R_{N(t')-1}))}{t} = \frac{\sum_{t'=1}^{R_{N(t)}} \mathbf{1}_{\{\xi^{(N(t'))} \leq K\}} f(\mathbf{X}^{(N(t'))}(t' - R_{N(t')-1}))}{R_{N(t)}} \cdot \frac{R_{N(t)}}{t}.$$

Most importantly, $\lim_{t \to \infty} \frac{R_{N(t)}}{t} = 1$, as by definition $R_{N(t)} + \xi^{(N(t)+1)} > t$, and $\lim_{t \to \infty} \frac{\xi^{(N(t)+1)}}{t} = 0$, otherwise an infinitely large $\xi^{(N(t)+1)}$ would have non-zero measure, contradicting $\mathbb{E}[\xi^{(N(t)+1)}] < \infty$. This yields,

$$\lim_{t \to \infty} \frac{\sum_{r=1}^{N(t)} Y_K^{(r)}}{t} = \lim_{t \to \infty} \frac{\sum_{t'=1}^{R_{N(t)}} \mathbf{1}_{\{\xi^{(N(t'))} \leq K\}} f(\mathbf{X}^{(N(t'))}(t' - R_{N(t')-1}))}{R_{N(t)}}.$$

(Theorem 3) The case of $K_{\textbf{dyn}}$: As K_{dyn} is finite almost surely (see proof of Lemma 1), it makes the condition $\mathbf{1}_{\{\xi^{(N(t'))} \leq K_{\text{dyn}}\}} := 1$. Note that the sequence $\{\mathbf{X}^{(N(t'))}(t' - R_{N(t')-1})\}_{t'=1}^{R_{N(t)}}$ is just a single sample path of our MC starting at state $\mathbf{x}'$, taking $R_{N(t)}$ steps. As our MC is irreducible and time-reversible, there is solution and the solution is unique (Gallager 2013, Theorem 6.3.8), and thus we can use the ergodic theorem to show

$$\lim_{t \to \infty} \frac{\sum_{t'=1}^{R_{N(t)}} f(\mathbf{X}^{(N(t'))}(t' - R_{N(t')-1}))}{R_{N(t)}} = \sum_{\mathbf{x}} f(\mathbf{x}) p(\mathbf{x}; \mathbf{W}),$$

and substituting the above equation in (9), yields

$$\mathbb{E}\left[\sum_{k=1}^{\xi^{(\cdot)}} f(\mathbf{X}^{(\cdot)}(k))\right] = \mathbb{E}[\xi^{(\cdot)}] \sum_{\mathbf{x}} f(\mathbf{x}) p(\mathbf{x}; \mathbf{W}), \quad r \geq 1. \tag{10}$$

Finally, by Kac's theorem (Meyn and Tweedie 2012, Theorem 10.2.2),

$$\mathbb{E}[\xi^{(\cdot)}] = \frac{1}{p(\mathbf{x}'; \mathbf{W})} = \frac{Z(\mathbf{W})}{e^{-E(\mathbf{x}'; \mathbf{W})}}, \tag{11}$$

as $p(\mathbf{x}'; \mathbf{W})$ is the steady state probability of visiting state $\mathbf{x}'$. Replacing (11) into (10) and multiplying it by $e^{-E(\mathbf{x}'; \mathbf{W})}$ on both sides concludes the unbiasedness proof. Thus, if $\hat{F}_r^{(K_{\text{dyn}})}(\mathbf{W}, f)$ denotes the estimator $\hat{F}$ in eq.(5) applied to only a single tour $r = 1, \ldots, R$. Then, $\mathbb{E}[F_r^{(K_{\text{dyn}})}(\mathbf{W}, f)] = F(\mathbf{W}, f)$ and the sequence $\{F_r^{(K_{\text{dyn}})}(\mathbf{W}, f)\}_{r \geq 1}$ is trivially iid by the strong Markov property. This iid sequence guarantees the following convergence properties.

Error bound: Note that $\sum_{k=1}^{\xi^{(r)}} \frac{\partial E(\mathbf{X}^{(r)}(k); \mathbf{W})}{\partial \mathbf{W}}$ is upper bounded by $\xi^{(r)} B$. As $\Phi(\mathbf{W})$ is time-reversible, it is equivalent to a random walk on a weighted graph. Thus, Lemma 2(i) of Avrachenkov et al. (Avrachenkov, Ribeiro, and Sreedharan 2016) applies with $Z(\mathbf{W}) = 2d_{\text{tot}}$, $Z_{\mathcal{S}}(\mathbf{W}) = d_{\mathcal{S}_n}$, and we have

$$\text{var}(\hat{F}_1^{(K_{\text{dyn}})}(\mathbf{W})) \leq B^2 \left((Z(\mathbf{W}))^2 / (Z_{\mathcal{S}}(\mathbf{W}) \delta) + 1\right).$$

By the strong Markov property the tours are independent, thus, $\text{var}(\hat{F}^{(K_{\text{dyn}}, R)}(\mathbf{W})) = \text{var}(\hat{F}_1^{(K_{\text{dyn}})}(\mathbf{W}))/R$ by the Bienaymé formula. And we have already shown that the estimate of $\hat{F}_1^{(K_{\text{dyn}})}(\mathbf{W})$ is unbiased. Finally, we obtain the bound through the application of Chebyshev's inequality.

(Theorem 1) The case of K: From above, $\hat{F}^{(K_{\text{dyn}}, R)}(\mathbf{W}, f)$ is an unbiased estimate of $F(\mathbf{W}, f)$. The remaining of the proof is straightforward. The tours are independent. Thus $\mathbb{E}[\hat{F}^{(K,R)}(\mathbf{W})] = \mathbb{E}[\hat{F}^{(K,1)}(\mathbf{W})]$. Note that

$$E[\xi^{(\cdot)}] - \sum_{k=1}^{K-1} k P[\xi^{(\cdot)} = k] = \sum_{k=K}^{\infty} k P[\xi^{(\cdot)} = k],$$

and as B upper bounds $\|f(\cdot)\|_1$, then the bias $\mathbb{E}[\hat{F}^{(K_{\text{dyn}}, 1)}(\mathbf{W}) - \hat{F}^{(K,R)}(\mathbf{W})]$ can be at most $(E[\xi^{(\cdot)}] - \sum_{k=1}^{K-1} k P[\xi^{(\cdot)} = k]) \cdot B$.

□

Theorem 2

Proof of Theorem 2. Note that the condition

$$\inf_{\mathbf{x}\in\Omega\setminus\mathcal{S}} \sum_{\mathbf{y}\in\mathcal{S}} p_{\Phi}(\mathbf{x},\mathbf{y}) \geq \epsilon$$

ensures that the MC $\Phi'(\mathbf{W})$ satisfies Doeblin's condition, and therefore $\Phi'(\mathbf{W})$ is geometrically ergodic with convergence rate $(1-\epsilon)$ (Stroock 1995, pp. 30). Finally, by Kendall's theorem (Meyn and Tweedie 2012, Theorem 15.1.1), a geometric ergodicity and a geometric decay in the tail of the return time distribution are equivalent conditions. □

Proof of Corollary 3

Proof. An unbiased estimate of $\nabla_{\mathbf{W}}\mathcal{L}_Z$ for one tour is obtained from $F^{(K_{\text{dyn}},R)}(\mathbf{W},f)$ in eq. (6) of Theorem 3 with $f(\mathbf{y}) = \frac{1}{N}\sum_{n=1}^{N}\frac{\partial E(\mathbf{x}_n;\mathbf{W})}{\partial \mathbf{W}} - \frac{\partial E(\mathbf{y};\mathbf{W})}{\partial \mathbf{W}}$. Averaging the gradient of each tour over $R \geq 1$ tours gives the desired result. □

Source Code

Our source code and detailed results are hosted at `https://github.com/PurdueMINDS/MCLV-RBM`.

References

Aldous, D., and Fill, J. 2002. Reversible markov chains and random walks on graphs.

Avrachenkov, K.; Ribeiro, B.; and Sreedharan, J. K. 2016. Inference in OSNs via Lightweight Partial Crawls. In *SIGMETRICS*.

Baxendale, P. H. 2005. Renewal theory and computable convergence rates for geometrically ergodic Markov chains. *Ann. Appl. Probab.*

Bengio, Y., and Delalleau, O. 2009. Justifying and Generalizing Contrastive Divergence. *Neural Computation*.

Brémaud, P. 2013. *Markov chains: Gibbs fields, Monte Carlo simulation, and queues*. Springer.

Bubley, R., and Dyer, M. 1997. Path coupling: A technique for proving rapid mixing in Markov chains. In *FOCS*.

Buchholz, P. 1994. Exact and ordinary lumpability in finite markov chains. *Journal of Applied Probability*.

Carreira-Perpiñán, M. A., and Hinton, G. E. 2005. On Contrastive Divergence Learning. In *AISTATS*.

Corcoran, J., and Tweedie, R. 2002. Perfect sampling from independent Metropolis-Hastings chains. *J. Stat. Plan. Inference*.

Erhan, D.; Bengio, Y.; Courville, A.; Manzagol, P.-A.; Vincent, P.; and Bengio, S. 2010. Why does unsupervised pre-training help deep learning? *JMLR*.

Fill, J. A.; Machida, M.; Murdoch, D. J.; and Rosenthal, J. S. 2000. Extension of Fill's perfect rejection sampling algorithm to general chains. *RSA*.

Fill, J. A. 1997. An interruptible algorithm for perfect sampling via markov chains. In *STOC*.

Fischer, A., and Igel, C. 2014. Training restricted Boltzmann machines: An introduction. *Pattern Recognit.*

Gallager, R. G. 2013. *Stochastic processes: theory for applications*. Cambridge University Press.

Gilks, W. R.; Roberts, G. O.; and Sahu, S. K. 1998. Adaptive Markov Chain Monte Carlo through Regeneration. *J. AMSTAT*.

Hinton, G. E.; Osindero, S.; and Teh, Y.-W. 2006. A fast learning algorithm for deep belief nets. *Neural Computation*.

Hinton, G. E. 2002. Training products of experts by minimizing contrastive divergence. *Neural Computation*.

Hinton, G. E. 2012. A Practical Guide to Training Restricted Boltzmann Machines. In *Neural networks: Tricks of the trade*. Springer.

Hobert, J. P.; Jones, G. L.; Presnell, B.; and Rosenthal, J. S. 2002. On the applicability of regenerative simulation in Markov chain Monte Carlo. *Biometrika* (4).

Long, P. M., and Servedio, R. 2010. Restricted boltzmann machines are hard to approximately evaluate or simulate. In *ICML*.

Lyne, A.-M.; Girolami, M.; Atchadé, Y.; Strathmann, H.; Simpson, D.; et al. 2015. On Russian roulette estimates for Bayesian inference with doubly-intractable likelihoods. *Statistical science*.

Meyn, S. P., and Tweedie, R. L. 2012. *Markov chains and stochastic stability*. Springer.

Montúfar, G., and Morton, J. 2015. Discrete restricted boltzmann machines. *JMLR*.

Mykland, P.; Tierney, L.; and Yu, B. 1995. Regeneration in Markov chain samplers. *J. AMSTAT* (429).

Neal, R. M. 2001. Annealed Importance Sampling. *Statistics and Computing* 11(2):125–139.

Neal, R. M. 2005. Estimating Ratios of Normalizing Constants using Linked Importance Sampling. *arXiv preprint math/0511216*.

Ollivier, Y. 2009. Ricci curvature of Markov chains on metric spaces. *J. Funct. Anal.*

Prats, D. B.; Romero Merino, E.; and Castrillejo, F. M. 2014. Stopping Criteria in Contrastive Divergence: Alternatives to the Reconstruction Error. In *ICML*.

Propp, J. G., and Wilson, D. B. 1996. Exact sampling with coupled markov chains and applications to statistical mechanics. *RSA*.

Propp, J. G., and Wilson, D. B. 1998. How to get a perfectly random sample from a generic markov chain and generate a random spanning tree of a directed graph. *Journal of Algorithms*.

Roberts, G. O., and Tweedie, R. L. 1999. Bounds on regeneration times and convergence rates for Markov chains. *Stoch. Process. their Appl.*

Salakhutdinov, R., and Hinton, G. 2009. Deep Boltzmann Machines. *JMLR*.

Salakhutdinov, R., and Murray, I. 2008. On the quantitative analysis of deep belief networks. In *ICML*.

Schulz, H.; Müller, A.; and Behnke, S. 2010. Investigating Convergence of Restricted Boltzmann Machine Learning. In *NIPS*.

Smolensky, P. 1986. Information processing in dynamical systems: Foundations of harmony theory, tech. rep. dtic.

Stewart, W. J. 1994. *Introduction to the numerical solutions of Markov chains*. Princeton Univ. Press.

Stroock, D. 1995. *An introduction to Markov processes*. Springer.

Sutskever, I., and Tieleman, T. 2010. On the Convergence Properties of Contrastive Divergence. In *AISTATS*.

Tieleman, T. 2008. Training restricted Boltzmann machines using approximations to the likelihood gradient. In *ICML*.

Wilson, J. R. 1983. The inspection paradox in renewal-reward processes. *Operations Research Letters*.

Wilson, D. B. 2000. Layered multishift coupling for use in perfect sampling algorithms (with a primer on cftp). *Monte Carlo Methods*.

Yuille, A. 2005. The Convergence of Contrastive Divergences. In *NIPS*.