ANNEALING GAUSSIAN INTO RELU: A NEW SAMPLING STRATEGY FOR LEAKY-RELU RBM

Chun-Liang Li Siamak Ravanbakhsh Barnabás Póczos
Department of Machine Learning
Carnegie Mellon University
Pittsburgh, PA 15213, USA
{chunlial,mravanba,bapoczos}@cs.cmu.edu

ABSTRACT

Restricted Boltzmann Machine (RBM) is a bipartite graphical model that is used as the building block in energy-based deep generative models. Due to its numerical stability and quantifiability of its likelihood, RBM is commonly used with Bernoulli units. Here, we consider an alternative member of the exponential family RBM with leaky rectified linear units – called leaky RBM. We first study the joint and marginal distributions of the leaky RBM under different leakiness, which leads to interesting interpretation of the leaky RBM model as truncated Gaussian distribution. We then propose a simple yet efficient method for sampling from this model, where the basic idea is to anneal the leakiness rather than the energy; – *i.e.*, start from a fully Gaussian/Linear unit and gradually decrease the leakiness over iterations. This serves as an alternative to the annealing of the temperature parameter and enables numerical estimation of the likelihood that are more efficient and far more accurate than the commonly used annealed importance sampling (AIS). We further demonstrate that the proposed sampling algorithm enjoys relatively faster mixing than contrastive divergence algorithm, which improves the training procedure without any additional computational cost.

1 INTRODUCTION

In this paper, we are interested in deep generative models. One may naively classify these models into a family of directed deep generative models trainable by back-propagation (*e.g.*, Kingma & Welling, 2013; Goodfellow et al., 2014), and deep energy-based models, such as deep belief network (Hinton et al., 2006) and deep Boltzmann machine (Salakhutdinov & Hinton, 2009). The building block of deep energy-based models is a bipartite graphical model called restricted Boltzmann machine (RBM). The RBM model consists of two layers, visible and hidden. The resulting graphical model which can account for higher-order interactions of the visible units (visible layer) using the hidden units (hidden layer). It also makes the inference easier that there are no interactions between the variables in each layer.

The conventional RBM uses Bernoulli units for both the hidden and visible units (Smolensky, 1986). One extension is using Gaussian visible units to model general natural images (Freund & Haussler, 1994). For hidden units, we can also generalize Bernoulli units to the exponential family (Welling et al., 2004; Ravanbakhsh et al., 2016).

Nair & Hinton (2010) propose a variation using Rectified Linear Unit (ReLU) for the hidden layer with a heuristic sampling procedure, which has promising performance in terms of reconstruction error and classification accuracy. Unfortunately, due to its lack of strict monotonicity, ReLU RBM does not fit within the framework of exponential family RBMs (Ravanbakhsh et al., 2016). Instead we study leaky-ReLU RBM (leaky RBM) in this work and address two important issues **i**) a better training (sampling) algorithm for ReLU RBM and; **ii**) a better quantification of leaky RBM –*i.e.*, evaluation of its performance in terms of likelihood.

We study some of the fundamental properties of leaky RBM, including its joint and marginal distributions (Section 2). By analyzing these distributions, we show that the leaky RBM is a *union of* *truncated Gaussian distributions*. In this paper, we show that training leaky RBM involves underlying positive definite constraints. Because of this, the training can diverge if these constrains are not satisfied. This is an issue that was previously ignored in ReLU RBM, as it was mainly used for pre-training rather than generative modeling.

Our **contribution** in this paper is three-fold: **I**) we systematically identify and address model constraints in leaky RBM (Section 3); **II**) for the training of leaky RBM, we propose a meta algorithm for sampling, which anneals leakiness during the Gibbs sampling procedure (Section 3) and empirically show that it can boost contrastive divergence with faster mixing (Section 5); **III**) We demonstrate the power of the proposed sampling algorithm on *estimating the partition function*. In particular, comparison on several benchmark datasets shows that the proposed method outperforms the conventional AIS (Salakhutdinov & Murray, 2008) in terms of efficiency and accuracy (Section 4). Moreover, we provide an incentive for using leaky RBM by showing that the leaky ReLU hidden units perform better than the Bernoulli units in terms of the model log-likelihood (Section 4).

2 Restricted Boltzmann Machine and ReLU

The Boltzmann distribution is defined as $p(x) = e^{-E(x)}/Z$ where $Z = \sum_x e^{-E(x)}$ is the partition function. Restricted Boltzmann Machine (RBM) is a Boltzmann distribution with a bipartite structure It is also the building block for many deep models (*e.g.*, Hinton et al., 2006; Salakhutdinov & Hinton, 2009; Lee et al., 2009), which are widely used in numerous applications (Bengio, 2009). The conventional *Bernoulli* RBM, models the joint probability $p(v, h)$ for the visible units $v \in [0, 1]^I$ and the hidden units $h \in [0, 1]^J$ as $p(v, h) \propto \exp(-E(v, h))$, where

$$E(v, h) = a^\top v - v^\top Wh + b^\top h.$$

The parameters are $a \in \mathbb{R}^I$, $b \in \mathbb{R}^J$ and $W \in \mathbb{R}^{I \times J}$. We can derive the conditional probabilities as

$$p(v_i = 1|h) = \sigma\left(\sum_{j=1}^{J} W_{ij} h_j + a_i\right) \quad \text{and} \quad p(h_j = 1|v) = \sigma\left(\sum_{i=1}^{I} W_{ij} v_i + b_j\right), \tag{1}$$

where $\sigma(x) = (1 + e^{-x})^{-1}$ is the sigmoid function.

One extension of Bernoulli RBM is replacing the binary visible units by linear units $v \in \mathbb{R}^I$ with independent Gaussian noise. The energy function in this case is given by

$$E(v, h) = \sum_{i=1}^{I} \frac{(v_i - a_i)^2}{2\sigma_i^2} - \sum_{i=1}^{I} \sum_{j=1}^{J} \frac{v_i}{\sigma_i} W_{ij} h_j + b^\top h.$$

To simplify the notation, we assume a normalized data so that a_i and σ_i is no longer required. The energy function is accordingly simplified to $E(v, h) = \frac{\|v\|^2}{2} - v^\top Wh + b^\top h$ (Note that the elimination does not influence the discussion and one can easily extend all the results in this paper to the model that includes a_i and σ_i.).

The conditional distributions are as follows:

$$p(v_i|h) = \mathcal{N}\left(\sum_{j=1}^{J} W_{ij} h_j, 1\right) \quad \text{and} \quad p(h_j = 1|v) = \sigma\left(\sum_{i=1}^{I} W_{ij} v_i + b_j\right), \tag{2}$$

where $\mathcal{N}(\mu, V)$ is a Gaussian distribution with mean μ and variance V.

2.1 ReLU RBM with Continuous Visible Units

From (1) and (2), we can see that the mean of the $p(h_j|v)$ is the nonlinearity of the hidden unit at $\sum_{i=1}^{I} W_{ij} v_i + b_j$ – *e.g.*, mean of the Bernoulli unit is the sigmoid function. From this perspective, we can extend the sigmoid function to other functions and thus allow RBM to have more expressive power (Ravanbakhsh et al., 2016). One special case of this is *rectified linear unit* (ReLU Nair & Hinton, 2010) activation function, defined as $\max(0, x)$.

However, only the strictly monotonic activation functions can derive feasible joint and conditional distributions in the exponential familly RBM Ravanbakhsh et al. (2016)[1]. One implication is that, even if ReLU RBM defines a valid distribution, we do not have access to its joint distribution and corresponding energy function via the formalism of exponential family RBM and therefore would not be able to "evaluate" its generative performance in terms of likelihood. That is why we consider the leaky ReLU (Maas et al., 2013) in this paper. The activation function of leaky ReLU is defined as $\max(cx, x)$, where $c \in (0, 1)$ is the leakiness parameter.

To simplify the notation, we define $\eta_j = \sum_{i=1}^{I} W_{ij} v_i + b_j$ – that is η_j is the input to the j^{th} hidden layer neuron. By Ravanbakhsh et al. (2016), the conditional probability of the activation, assuming the nonlinearity $f(\eta_j)$, is defined as $p(h_j|v) = \exp\left(-D_f(\eta_j \| h_j) + g(h_j)\right)$, where $D_f(\eta_j \| h_j)$ is a *Bregman Divergence* and $g(h_j)$ is the *base* (or carrier) measure in the exponential family which ensures the distribution is well-defined. The Bergman divergence, for strictly monotonic function f, is $D_f(\eta_j \| h_j) = -\eta_j h_j + F(\eta_j) + F^*(h_j)$, where F with $\frac{d}{d\eta_j} F(\eta_j) = f(\eta_j)$ is the anti-derivative (integral) of f and F^* is the anti-derivative of f^{-1} (*i.e.*, $f^{-1}(f(\eta)) = \eta$); Note that due to the strict monotonicity of f, f^{-1} is well-defined, and F and F^* are commonly referred to as conjugate duals.

Considering the leaky ReLU activation function $f(\eta) = \max(c\eta, \eta)$, using this formalism, the conditional distributions of hidden units in the leaky RBM simplifies to (see Appendix A.1 for details)

$$p(h_j|v) = \begin{cases} \mathcal{N}(\eta_j, 1), & \text{if } \eta_j > 0 \\ \mathcal{N}(c\eta_j, c), & \text{if } \eta_j \leq 0. \end{cases} \tag{3}$$

Since the visible units uses the identity function, the corresponding conditional distribution is a Gaussian[2]

$$p(v_i|h) = \mathcal{N}\left(\sum_{j=1}^{J} W_{ij} h_j, 1\right), \tag{4}$$

Having these two conditional distributions is enough for training a leaky RBM model using contrastive divergence (Hinton, 2002) or some other alternatives (*e.g.*, Tieleman, 2008; Tieleman & Hinton, 2009).

3 Training and Sampling from leaky RBM

Given the conditional distributions $p(v|h)$ and $p(h|v)$, the joint distribution $p(v, h)$ from the general treatment for MRF model is (Yang et al., 2012; Ravanbakhsh et al., 2016)

$$p(v, h) \propto \exp\left(v^\top W h - \sum_{i=1}^{I} (\tilde{F^*}(v_i) + g(v_i)) - \sum_{j=1}^{J} (F^*(h_j) + g(h_j))\right), \tag{5}$$

where $\tilde{F^*}(v_i)$ and $F^*(h_j)$ are anti-derivatives of the inverses of the activation functions $\tilde{f}(v_i)$ and $f(h_j)$ for visible units v_i and hidden units h_j, respectively (see Section 2.1). Assuming $f(\eta_j) = \max(c\eta_j, c)$ and $\tilde{f}(\nu_i) = \nu_i$ in leaky-ReLU RBM, the joint distribution above becomes (see Appendix A.2 for details)

$$p(v, h) \propto \exp\left(v^\top W h - \frac{\|v\|^2}{2} - \sum_{\eta_j > 0}\left(\frac{h_j^2}{2} + \log\sqrt{2\pi}\right) - \sum_{\eta_j \leq 0}\left(\frac{h_j^2}{2c} + \log\sqrt{2c\pi}\right) + b^\top h\right),$$

and the corresponding visible marginal distribution is

$$p(v) \quad \propto \quad \exp\left(-\frac{1}{2} v^\top \left(I - \sum_{\eta_j > 0} W_j W_j^\top - c \sum_{\eta_j \leq 0} W_j W_j^\top\right) v + \sum_{\eta_j > 0} b_j W_j^\top v + c \sum_{\eta_j \leq 0} b_j W_j^\top v\right). \tag{6}$$

[1] Note that Nair & Hinton (2010) are interested in the pre-training for classification, rather than generative modeling.

[2] which can also be written as $p(v_i|h) = \exp\left(-D_{\tilde{f}}(\nu_i \| v_i) + g(v_i)\right)$, where $\nu_i = \sum_{j=1} W_{ij} h_j$ and $\tilde{f}(\nu_i) = \nu_i$ and $D_{\tilde{f}}(\nu_i \| v_i) = (\nu_i - v_i)^2$ and $g(v_i) = -\log\sqrt{2\pi}$.

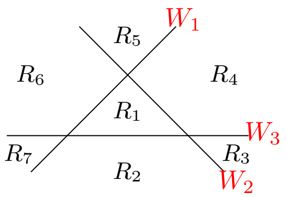


Figure 1: A two dimensional example with 3 hidden units.

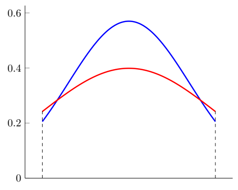


Figure 2: An one dimensional example of truncated Gaussian distributions with different variances.

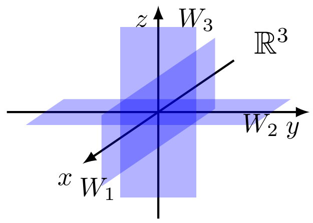


Figure 3: A three dimensional example with 3 hidden units, where W_j are orthogonal to each other.

where W_j is the j-th column of W.

3.1 Leaky RBM as Union of Truncated Gaussian Distributions

From (6) we see that the marginal probability is determined by the affine constraints $\eta_j > 0$ or $\eta_j \leq 0$ for all hidden units j. By combinatorics, these constraints divide $\mathbb{R}^I$ (the visible domain) into at most $M = \sum_{i=1}^{I} \binom{J}{i}$ convex regions $R_1, \cdots R_M$. An example with $I = 2$ and $J = 3$ is shown in Figure 2. If $I > J$, then we have at most 2^J regions.

We discuss the two types of these regions. For bounded regions, such as R_1 in Figure 2, the integration of (6) is also bounded, which results in a valid distribution. Before we discuss the unbounded cases, we define $\Omega = I - \sum_{j=1}^{J} \alpha_j W_j W_j^\top$, where $\alpha_j = \mathbb{1}_{\eta_j > 0} + c\mathbb{1}_{\eta_j \leq 0}$. For the unbounded region, if $\Omega \in \mathbb{R}^{I \times I}$ is a positive definite (PD) matrix, then the probability density is proportional to a multivariate Gaussian distribution with mean $\mu = \Omega^{-1}\left(\sum_{j=1}^{J} \alpha_j b_j W_j\right)$ and precision matrix Ω (covariance matrix Ω^{-1}) but over an affine-constrained region. Therefore, the distribution of each unbounded region can be treated as a truncated Gaussian distribution.

On the other hand, if Ω is not PD, and the region R_i contains the eigenvectors with negative eigenvalues of Ω, the integration of (6) over R_i is divergent (infinite), which can not result in a valid probability distribution. In practice, with this type of parameter, when we do Gibbs sampling on the conditional distributions, the sampling will diverge. However, it is unfeasible to check exponentially many regions for each gradient update.

Theorem 1. *If $I - WW^\top$ is positive definite, then $I - \sum_j \alpha_j W_j W_j^\top$ is also positive definite, for all $\alpha_j \in [0, 1]$.*

The proof is shown in Appendix 1. From Theorem 1 we can see that if the constraint $I - WW^\top$ is PD, then one can guarantee that the distribution of every region is a valid truncated Gaussian distribution. Therefore, we introduce the following projection step for each W after the gradient update.

$$\begin{aligned} \underset{\tilde{W}}{\operatorname{argmin}} \quad & \|W - \tilde{W}\|_F^2 \\ \text{s.t.} \quad & I - \tilde{W}\tilde{W}^\top \succeq 0 \end{aligned} \tag{7}$$

Theorem 2. *The above projection step (7) can be done by shrinking the singular values to be less than 1.*

The proof is shown in Appendix C. The training algorithm of the leaky RBM is shown in Algorithm 1. By using the projection step (7), we could treat the leaky RBM as the *union of truncated Gaussian distributions*, which uses weight vectors to divide the space of visible units into several regions and use a truncated Gaussian distribution to model each region. Note that the leaky RBM model is different from Su et al. (2016), which uses a truncated Gaussian distribution to model the conditional distribution $p(h|v)$ instead of the marginal distribution.

The empirical study about the divergent values and the necessity of the projection step is shown in Appendix D. Without the projection step, when we run Gibbs sampling for several iterations from the model, the sampled values will diverge because the model does not have a valid marginal distribution $p(v)$. It also implies that we cannot train leaky RBM with larger CD steps when we do not do projection; otherwise, we would have the diverged gradients. The detailed discussion is shown in Appendix D.

Algorithm 1 Training Leaky RBM

```
for $t = 1, \dots, T$ do
    Estimate gradient $g_\theta$ by CD or other algorithms with (13) and (4), where $\theta = \{W, a, b\}$.
    $\theta^{(t)} \leftarrow \theta^{(t-1)} + \eta g_\theta$.
    Project $W^{(t)}$ by (7).
end for
```

3.2 Sampling from Leaky-ReLU RBM

Gibbs sampling is the core procedure for RBM, including training, inference, and estimating the partition function (Fischer & Igel, 2012; Tieleman, 2008; Salakhutdinov & Murray, 2008). For every task, we start from randomly initializing v by an arbitrary distribution q, and iteratively sample from the conditional distributions. Gibbs sampling guarantees the procedure result in the stationary distribution in the long run for any initialized distribution q. However, if q is close to the target distribution p, it can significantly shorten the number of iterations to achieve the stationary distribution.

If we set the leakiness c to be 1, then (6) becomes a simple multivariate Gaussian distribution $\mathcal{N}\left((I - WW^\top)^{-1}Wb, (I - WW^\top)^{-1}\right)$, which can be easily sampled without Gibbs sampling. Also, the projection step (7) guarantees it is a valid Gaussian distribution. Then we decrease the leakiness with a small ϵ, and use samples from the multivariate Gaussian distribution when $c = 1$ as the initialization to do Gibbs sampling. Note that the distribution of each region is a truncated Gaussian distribution. When we only decrease the leakiness with a small amount, the resulted distribution is a "similar" truncated Gaussian distribution with more concentrated density. From this observation, we could expect the original multivariate Gaussian distribution serves as a good initialization. The one-dimensional example is shown in Figure 2. We then repeat this procedure until we reach the target leakiness. The algorithm can be seen as *annealing the leakiness* during the Gibbs sampling procedure. The meta algorithm is shown in Algorithm 2. Next, we show the proposed sampling algorithm can help both the partition function estimation and the training of leaky RBM.

Algorithm 2 Meta Algorithm for Sampling from Leaky RBM

```
Sample $v$ from $\mathcal{N}\left((I - WW^\top)^{-1}Wb, (I - WW^\top)^{-1}\right)$
$\epsilon = (1 - c)/T$
$c' = 1$
for $t = 1, \dots, T$ do
    if $c' > c$ then
        $c' = c' - \epsilon$
    end if
    Do Gibbs sampling by using (13) and (4) with leakiness $c'$
end for
```

4 Partition Function Estimation

It is known that estimating the partition function of RBM is intractable (Salakhutdinov & Murray, 2008). Existing approaches, including Salakhutdinov & Murray (2008); Grosse et al. (2013); Liu et al. (2015); Carlson et al. (2016) focus on using sampling to approximate the partition function of the conventional Bernoulli RBM instead of the RBM with Gaussian visible units and non-Bernoulli hidden units. In this paper, we focus on extending the classic annealed importance sampling (AIS) algorithm (Salakhutdinov & Murray, 2008) to leaky RBM.

Assuming that we want to estimate the partition function Z of $p(v)$ with $p(v) = p^*(v)/Z$ and $p^*(v) \propto \sum_h \exp(-E(v,h))$, Salakhutdinov & Murray (2008) start from a initial distribution $p_0(v) \propto \sum_h \exp(-E_0(v,h))$, where computing the partition Z_0 of $p_0(v)$ is tractable and we can

	$J=5$	$J=10$	$J=20$	$J=30$
Log partition function	2825.48	2827.98	2832.98	2837.99

Table 1: The true partition function for Leaky-ReLU RBM with different number of hidden units.

	$J=5$	$J=10$	$J=20$	$J=30$
AIS-Energy	1.76 ± 0.011	3.56 ± 0.039	7.95 ± 0.363	9.60 ± 0.229
AIS-Leaky	$\mathbf{0.02 \pm 0.001}$	$\mathbf{0.04 \pm 0.002}$	$\mathbf{0.08 \pm 0.003}$	$\mathbf{0.13 \pm 0.004}$

Table 2: The difference between the true partition function and the estimations of two algorithms with standard deviation.

draw samples from $p_0(v)$. They then use the "geometric path" to anneal the intermediate distribution as $p_k(v) \propto p_k^*(v) = \sum_h \exp\left(-\beta_k E_0(v,h) - (1-\beta_k)E(v,h)\right)$, where they grid β_k from 1 to 0. If we let $\beta_0 = 1$, we can draw samples v_k from $p_k(v)$ by using samples v_{k-1} from $p_{k-1}(v)$ for $k \geq 1$ via Gibbs sampling. The partition function is then estimated via $Z = \frac{Z_0}{M}\sum_{i=1}^M \omega^{(i)}$, where

$$\omega^{(i)} = \frac{p_1^*(v_0^{(i)})}{p_0^*(v_0^{(i)})}\frac{p_2^*(v_1^{(i)})}{p_1^*(v_1^{(i)})}\cdots\frac{p_{K-1}^*(v_{K-2}^{(i)})}{p_{K-2}^*(v_{K-2}^{(i)})}\frac{p_K^*(v_{K-1}^{(i)})}{p_{K-1}^*(v_{K-1}^{(i)})}, \text{ and } \beta_K = 0.$$

Salakhutdinov & Murray (2008) use the initial distribution with independent visible units and without hidden units. We consider application of AIS to the leaky-ReLU case with $E_0(v,h) = \frac{\|v\|^2}{2}$, which results in a multivariate Gaussian distribution $p_0(v)$. Compared with the meta algorithm shown in Algorithm 2 which *anneals between leakiness*, AIS *anneals between energy functions*.

4.1 Study on Toy Examples

As we discussed in Section 3.1, leaky RBM with J hidden units is a union of 2^J truncated Gaussian distributions. Here we perform a study on the leaky RBM with a small number hidden units. Since in this example the number of hidden units is small, we can integrate out all possible configurations of h. However, integrating a truncated Gaussian distribution with general affine constraints does not have analytical solutions, and several approximations have been developed (*e.g.*, Pakman & Paninski, 2014). To compare our results with the exact partition function, we consider a special case that has the following form:

$$p(v) \propto \exp\left(-\frac{1}{2}v^\top\left(I - \sum_{\eta_j>0} W_jW_j^\top - c\sum_{\eta_j\leq 0} W_jW_j^\top\right)v\right). \tag{8}$$

Compared to (6), it is equivalent to the setting where $b=0$. Geometrically, every W_j passes through the origin. We further put the additional constraint $W_i \perp W_j, \forall i \neq j$. Therefore. we divide the whole space into 2^J equally-sized regions. A three dimensional example is shown in Figure 3. Then the partition function of this special case has the analytical form

$$Z = \frac{1}{2^J}\sum_{\alpha_j\in\{1,c\},\forall j}(2\pi)^{-\frac{I}{2}}\left|\left(I - \sum_{j=1}^J \alpha_j W_jW_j^\top\right)^{-\frac{1}{2}}\right|.$$

We randomly initialize W and use SVD to make columns orthogonal. Also, we scale $\|W_j\|$ to satisfy $I - WW^\top \succeq 0$. The leakiness parameter is set to be 0.01. For Salakhutdinov & Murray (2008) (AIS-Energy), we use 10^5 particles with 10^5 intermediate distributions. For the proposed method (AIS-Leaky), we use only 10^4 particles with 10^3 intermediate distributions. In this small problem we study the cases when the model has 5, 10, 20 and 30 hidden units and 3072 visible units. The true log partition function $\log Z$ is shown in Table 1 and the difference between $\log Z$ and the estimates given by the two algorithms are shown in Table 2.

From Table 1, we observe that **AIS-Leaky has significantly better and more stable estimations than AIS-Energy especially and this gap increases as we increase the number of hidden units. AIS-Leaky achieves this with orders magnitude reduced computation** –*e.g.*, here it uses $\sim$.1%

	CIFAR-10	SVHN
Bernoulli-Gaussian RBM	-2548.3	-2284.2
Leaky-ReLU RBN	-1031.1	-182.4

Table 3: The log-likelihood performance of Bernoulli-Gaussian RBM and leaky RBM.

of resources used by conventional AIS. For example, when we increase J from 5 to 30, the bias (difference) of AIS-Leaky only increases from 0.02 to 0.13; however, the bias of AIS-Energy increases from 1.76 to 9.6. We further study the implicit connection between the proposed AIS-Leaky and AIS-Energy in Appendix E, which shows AIS-Leaky is a special case of AIS-Energy under certain conditions.

4.2 Comparison between leaky-ReLU RBM and Bernoulli-Gaussian RBM

It is known that the reconstruction error is not a proper approximation of the likelihood (Hinton, 2012). One commonly adopted way to compare generative models is to sample from the model, and visualize the images to check the quality. However, Theis et al. (2016) show the better visualization does not imply better likelihood. Also, the single layer model cannot adequately model the complicated natural images (the result for Bernoulli-Gaussian RBM has been shown in Ranzato & Hinton (2010)), which makes the visualization comparison difficult (Appendix F has few visualization results).

Fortunately, our accurate estimate of the partition function for leaky RBM can produce a reliable *quantitative* estimate of the representation power of leaky RBM. We compare the Bernoulli-Gaussian RBM[3], which has Bernoulli hidden units and Gaussian visible units. We trained both models with CD-20[4] and momentum. For both model, we all used 500 hidden units. We initialized W by sampling from Unif$(0, 0.01)$, $a = 0$, $b = 0$ and $\sigma = 1$. The momentum parameter was 0.9 and the batch size was set to 100. We tuned the learning rate between 10^{-1} and 10^{-6}. We studied two benchmark data sets, including CIFAR10 and SVHN. The data was normalized to have zero mean and standard deviation of 1 for each pixel. The results of the log-likelihood are reported in Table 3.

From Table 3, leaky RBM outperforms Bernoulli-Gaussian RBM significantly. The unsatisfactory performance of Bernoulli-Gaussian RBM may be in part due to the optimization procedure. If we tune the decay schedule of the learning-rate for each dataset in an ad-hoc way, we observe the performance of Bernoulli-Gaussian RBM can be improved by ~ 300 nats for both datasets. Also, increasing CD-steps brings slight improvement. The other possibility is the bad mixing during the CD iterations. The advanced algorithms Tieleman (2008); Tieleman & Hinton (2009) may help. Although Nair & Hinton (2010) demonstrate the power of ReLU in terms of reconstruction error and classification accuracy, it does not imply its superior generative capability. **Our study confirms leaky RBM could have much better generative performance compared to Bernoulli-Gaussian RBM.**

5 Better Mixing by Annealing Leakiness

In this section, we show the idea of annealing between leakiness benefit the mixing in Gibbs sampling in other settings. A common procedure for comparison of *sampling* methods for RBM is through visualization. Here, we are interested in more quantitative metrics and the practical benefits of improved sampling. For this, we consider *optimization performance* as the evaluation metric.

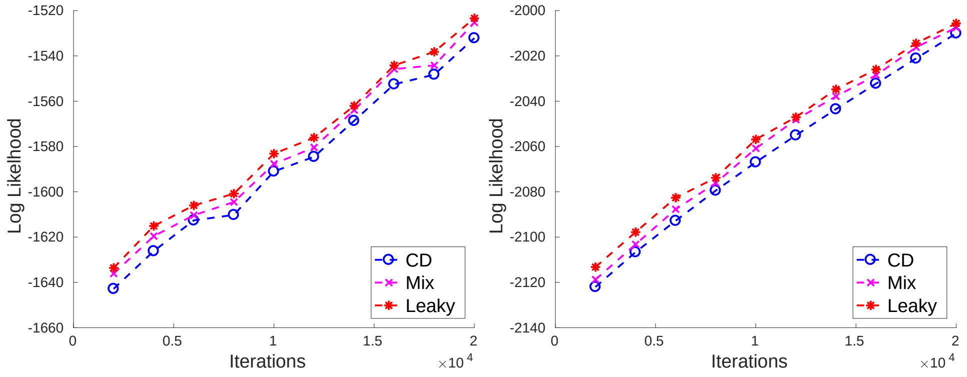
The gradient of the log-likelihood function $\mathcal{L}(\theta|v_{data})$ of general RBM models is

$$\frac{\partial \mathcal{L}(\theta|v_{data})}{\partial \theta} = \mathbb{E}_{h|v_{data}} \left[\frac{\partial E(v, h)}{\partial \theta} \right] - \mathbb{E}_{v,h} \left[\frac{\partial E(v, h)}{\partial \theta} \right]. \tag{9}$$

Since the second expectation in (9) is usually intractable, different approximation algorithms are used (Fischer & Igel, 2012).

[3] Our GPU implementation with gnumpy and cudamat can reproduce the results of http://www.cs.toronto.edu/ tang/code/GaussianRBM.m

[4] CD-n means that contrastive divergence was run for n steps



(a) SVHN

(b) CIFAR10

Figure 4: Training leaky RBM with different sampling algorithms.

In this section, we compare two gradient approximation procedures. The first one is the conventional contrastive divergence (CD) (Hinton, 2002). The second method is using Algorithm 2 (Leaky) with the same number of mixing steps as CD. The experiment setup is the same as that of Section 4.

The results are shown in Figure 4. The proposed sampling procedure is slightly better than typical CD steps. The reason is we only anneals the leakiness for 20 steps. To get accurate estimation requires thousands of steps as shown in Section 4 when we estimate the partition function. Therefore, the estimated gradient is still inaccurate. However, it still outperforms the conventional CD algorithm.

The drawback of Algorithm 2 is that sampling v from $\mathcal{N}\left((I - WW^\top)^{-1}Wb, (I - WW^\top)^{-1}\right)$ requires computing mean, covariance and the Cholesky decomposition of the covariance matrix in every iteration, which are computationally expensive. We study a mixture algorithm by combining CD and the idea of annealing leakiness. The mixture algorithm replaces the sampling from $\mathcal{N}\left((I - WW^\top)^{-1}Wb, (I - WW^\top)^{-1}\right)$ with sampling from the empirical data distribution. The resulted mix algorithm is almost the same as CD algorithm while it *anneals the leakiness* over the iterations as Algorithm 2. The results of the mix algorithm is also shown in Figure 4.

The mix algorithm is slightly worse than the original leaky algorithm, but it also outperforms the conventional CD algorithm without additional computation cost. The comparison in terms of CPU time is shown in Appendix F. Annealing the leakiness helps the mix algorithm explore different modes of the distribution, thereby improves the training. The idea could also be combined with more advanced algorithms (Tieleman, 2008; Tieleman & Hinton, 2009)[5].

6 Conclusion

In this paper, we study the properties of the exponential family distribution produced by leaky RBM. This study relates the leaky RBM model and truncated Gaussian distribution and reveals an underlying positive definite constraint of training leaky RBM. We further proposed a meta sampling algorithm, which anneals between leakiness during the Gibbs sampling procedure. We first demonstrate the proposed sampling algorithm is significantly more effective and efficient in estimating the partition function than the conventional AIS algorithm. Second, we show that the proposed sampling algorithm has comparatively better mixing properties (compared to CD). A few direction are worth further study; in particular we are investigating on speeding up the naive projection step; either using the barrier function as shown in Hsieh et al. (2011) or by eliminating the need for projection by artificially bounding the domain via additional constraints.

References

Y. Bengio. Learning deep architectures for ai. *Found. Trends Mach. Learn.*, 2009.

[5] We studied the PCD extension of the proposed sampling algorithm. However, the performance is not as stable as CD.

Y. Burda, R. B. Grosse, and R. Salakhutdinov. Accurate and conservative estimates of mrf log-likelihood using reverse annealing. In *AISTATS*, 2015.

D. E. Carlson, P. Stinson, A. Pakman, and L. Paninski. Partition functions from rao-blackwellized tempered sampling. In *ICML*, 2016.

A. Fischer and C. Igel. An introduction to restricted boltzmann machines. In *CIARP*, 2012.

Y. Freund and D. Haussler. Unsupervised learning of distributions on binary vectors using two layer networks. Technical report, 1994.

I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *ICML*. 2014.

R. B. Grosse, C. J. Maddison, and R. Salakhutdinov. Annealing between distributions by averaging moments. In *NIPS*, 2013.

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

G. E. Hinton. A practical guide to training restricted boltzmann machines. In *Neural Networks: Tricks of the Trade (2nd ed.)*. 2012.

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

C.-J. Hsieh, M. A. Sustik, I. S. Dhillon, and P. Ravikumar. Sparse inverse covariance matrix estimation using quadratic approximation. In *NIPS*, 2011.

D. P. Kingma and M. Welling. Auto-encoding variational bayes. *CoRR*, 2013.

H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In *ICML*, 2009.

Q. Liu, J. Peng, A. Ihler, and J. Fisher III. Estimating the partition function by discriminance sampling. In *UAI*, 2015.

A. L. Maas, A. Y. Hannun, and A. Y. Ng. Rectifier nonlinearities improve neural network acoustic models. In *ICML Workshop on Deep Learning for Audio, Speech, and Language Processing*, 2013.

V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *ICML*, 2010.

A. Pakman and L. Paninski. Exact hamiltonian monte carlo for truncated multivariate gaussians. *Journal of Computational and Graphical Statistics*, 2014.

N. Parikh and S. Boyd. Proximal algorithms. *Found. Trends Optim.*, 2014.

M. Ranzato and G. E. Hinton. Modeling pixel means and covariances using factorized third-order boltzmann machines. In *CVPR*, 2010.

S. Ravanbakhsh, B. Póczos, J. G. Schneider, D. Schuurmans, and R. Greiner. Stochastic neural networks with monotonic activation functions. In *AISTATS*, 2016.

R. Salakhutdinov and G. Hinton. Deep Boltzmann machines. In *AISTATS*, 2009.

R. Salakhutdinov and I. Murray. On the quantitative analysis of Deep Belief Networks. In *ICML*, 2008.

P. Smolensky. Parallel distributed processing: Explorations in the microstructure of cognition, vol. 1. 1986.

Q. Su, X. Liao, C. Chen, and L. Carin. Nonlinear statistical learning with truncated gaussian graphical models. In *ICML*, 2016.

L. Theis, A. van den Oord, and M. Bethge. A note on the evaluation of generative models. In *ICLR*, 2016.

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

T. Tieleman and G.E. Hinton. Using Fast Weights to Improve Persistent Contrastive Divergence. In *ICML*, 2009.

M. Welling, M. Rosen-Zvi, and G. E. Hinton. Exponential family harmoniums with an application to information retrieval. In *NIPS*, 2004.

E. Yang, P. Ravikumar, G. I. Allen, and Z. Liu. Graphical models via generalized linear models. In *NIPS*, 2012.

A Derivation of leaky (ReLU) RBM

A.1 Conditional Distributions

For leaky RBM, the activation function of hidden units is defined as $f(\eta_j) = \max(c\eta_j, \eta_j)$, where $c \in (0, 1)$ and $\eta_j = \sum_{i=1}^{I} W_{ij} v_i + b_j$. The inverse function of f is $f^{-1}(h_j) = \min(h_j, h_j/c)$. Therefore, the anti-derivatives are

$$F(\eta_j) = \begin{cases} \frac{1}{2}\eta_j^2, & \text{if } \eta_j > 0 \\ \frac{c}{2}\eta_j^2, & \text{else,} \end{cases} \tag{10}$$

and

$$F^*(h_j) = \begin{cases} \frac{1}{2}h_j^2, & \text{if } \eta_j > 0 \\ \frac{1}{2c}h_j^2, & \text{else.} \end{cases} \tag{11}$$

The activation function of Gaussian visible units can be treated as the linear unit $\tilde{f}(\nu_i) = \nu_i$, where $\nu_i = \sum_{j=1}^{J} W_{ij} h_j$. Following the similar steps for deriving F and F^*, we get the anti-derivatives $\tilde{F}(\nu_i) = \frac{1}{2}\nu_i^2$ and $\tilde{F}^*(v_i) = \frac{1}{2}v_i^2$.

From Ravanbakhsh et al. (2016), the conditional distribution is defined as

$$p(h_j|\eta_j) = \exp\left(-\eta_j h_j + F(\eta_j) + F^*(h_j)\right) \tag{12}$$

By plugging F and F^* into (12), we get the conditional distribution for leaky RBM

$$p(h_j|v) = \begin{cases} \mathcal{N}(\eta_j, 1)\text{with } g(h_j) = -\log(\sqrt{2\pi}), & \text{if } \eta_j > 0 \\ \mathcal{N}(c\eta_j, c)\text{with } g(h_j) = -\log(\sqrt{2c\pi}), & \text{if } \eta_j \leq 0. \end{cases} \tag{13}$$

Similarly, we have $p(v_i|\nu_i) = \mathcal{N}(\nu_i, 1)$ with $g(v_i) = -\log(\sqrt{2\pi})$.

A.2 Joint and Marginal Distributions

Given the conditional distributions $p(v|h)$ and $p(h|v)$, the joint distribution $p(v, h)$ from the general treatment for MRF model given by Yang et al. (2012) is

$$p(v, h) \propto \exp\left(v^\top W h - \sum_{i=1}^{I}(\tilde{F}^*(v_i) + g(v_i)) - \sum_{j=1}^{J}(F^*(h_j) + g(h_j))\right), \tag{14}$$

By plugging F^*, $\tilde{F}^*$ and g from Section A.1 into (14), we have

$$p(v, h) \propto \exp\left(v^\top W h - \frac{\|v\|^2}{2} - \sum_{\eta_j > 0}\left(\frac{h_j^2}{2} + \log\sqrt{2\pi}\right) - \sum_{\eta_j \leq 0}\left(\frac{h_j^2}{2c} + \log\sqrt{2c\pi}\right) + b^\top h\right),$$

Then the marginal distribution is

$$\begin{aligned}
p(v) &\propto \int_h p(v,h)dh \\
&\propto \int_h \exp\left(-\frac{\|v\|^2}{2}\right) \prod_{\eta_j>0} \exp\left(-\frac{h_j^2}{2} + \eta_j h_j - \log\sqrt{2\pi}\right) \prod_{\eta_j\le 0} \left(-\frac{h_j^2}{2c} + h_j\eta_j - \log\sqrt{2c\pi}\right) dh \\
&\propto \exp\left(-\frac{\|v\|^2}{2}\right) \prod_{\eta_j>0} \exp\left(\frac{\eta_j^2}{2}\right) \prod_{\eta_j\le 0} \left(\frac{c\eta_j^2}{2}\right) \\
&\propto \exp\left(-\frac{1}{2} v^\top \left(I - \sum_{\eta_j>0} W_j W_j^\top - c\sum_{\eta_j\le 0} W_j W_j^\top\right) v + \sum_{\eta_j>0} b_j W_j^\top v + c\sum_{\eta_j\le 0} b_j W_j^\top v\right).
\end{aligned}$$

B Proof of Theorem 1

Proof. Since $WW^\top - \sum_j \alpha_j W_j W_j = \sum_j (1-\alpha_j) W_j W_j^\top \succeq 0$, we have $WW^\top \succeq \sum_j \alpha_j W_j W_j$. Therefore, $I - \sum_j \alpha_j W_j W_j^\top \succeq I - WW^\top \succeq 0$. □

C Proof of Theorem 2

Proof. Let the SVD decomposition of W and $\tilde{W}$ as $W = USV^\top$ and $\tilde{W} = \tilde{U}\tilde{S}\tilde{V}^\top$. Then we have

$$\|W - \tilde{W}\|_F^2 = \|USV^\top - \tilde{U}\tilde{S}\tilde{V}^\top\|_F^2 \ge \sum_{i=1}^{I} (S_{ii} - \tilde{S}_{ii})^2, \tag{15}$$

and the constraint $I - \tilde{W}\tilde{W}^\top \succeq 0$ can be rewritten as $0 \le \tilde{S}_{ii} \le 1, \forall i$. The transformed problem has a Lasso-like formulation and we can solve it by $\tilde{S}_{ii} = \min(S_{ii}, 1)$ (Parikh & Boyd, 2014). Also, the lower bound $\sum_{i=1}^{I} (S_{ii} - \tilde{S}_{ii})^2$ in (15) becomes a tight bound when we set $\tilde{U} = U$ and $\tilde{V} = V$, which completes the proof. □

D Necessity of the Projection Step

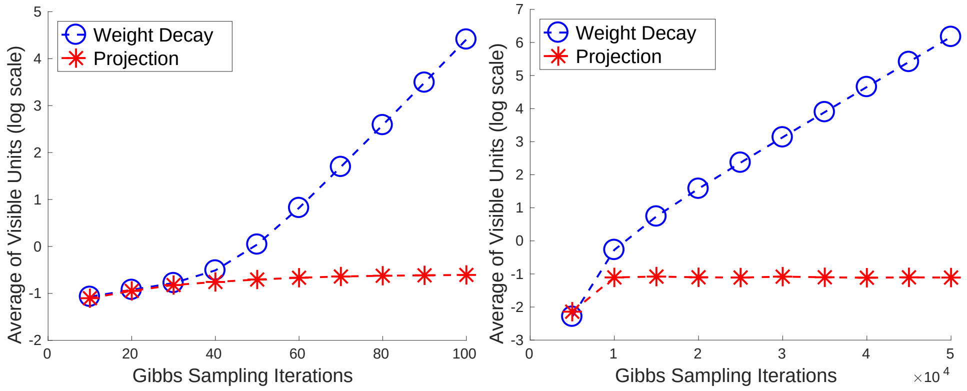
We conduct a short comparison to demonstrate the projection step is necessary for the leaky RBM on generative tasks. We train two leaky RBM as follows. The first model is trained by the same setting in Section 4. We use the convergence of log likelihood as the stopping criteria. The second model is trained by CD-1 with weight decay and without the projection step. We stop the training when the reconstruction error is less then 10^{-2}. After we train these two models, we run Gibbs sampling with 1000 independent chains for several steps and output the average value of the visible units. Note that the visible units are normalized to zero mean. The results on SVHN and CIFAR10 are shown in Figure 5.

From Figure 5, the model trained by weight decay without projection step is suffered by the problem of the diverged values. It confirms the study shown in Section 3.1. It also implies that we cannot train leaky RBM with larger CD steps when we do not do projection; otherwise, we would have the diverged gradients. Therefore, the projection is necessary for training leaky RBM for the generative purpose. However, we also oberseve that the projection step is not necessary for the classification and reconstruction tasks. he reason may be the independency of different evaluation criteria (Hinton, 2012; Theis et al., 2016) or other implicit reasons to be studied.

E Equivalence between Annealing the Energy and Leakiness

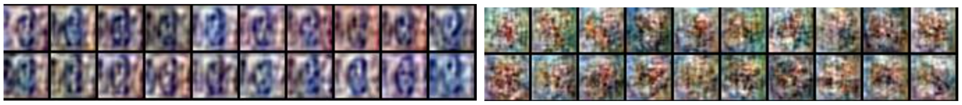
We analyze the performance gap between AIS-Leaky and AIS-Energy. One major difference is the initial distribution. The intermediate marginal distribution of AIS-Energy has the following form:

$$p_k(v) \propto \exp\left(-\frac{1}{2} v^\top \left(I - (1-\beta_k)\sum_{\eta_j>0} W_j W_j^\top - (1-\beta_k)c\sum_{\eta_j\le 0} W_j W_j^\top\right) v\right). \tag{16}$$



(a) SVHN (b) CIFAR10

Figure 5: Divergence results on two datasets.



(a) SVHN (b) CIFAR10

Figure 6: Sampled images from leaky RBM.

Here we eliminated the bias terms b for simplicity. Compared with Algorithm 2, (16) not only anneals the leakiness $(1-\beta_k)c\sum_{\eta_j\leq 0} W_jW_j^\top$ when $\eta_j \leq 0$, but also in the case $(1-\beta_k)\sum_{\eta_j>0} W_jW_j^\top$ when $\eta_j > 0$, which brings more bias to the estimation. In other words, AIS-Leaky is a *one-sided leakiness annealing* while AIS-Energy is a *two-sided leakiness annealing* method.

To address the higher bias problem of AIS-Energy, we replace the initial distribution with the one used in Algorithm 2. By elementary calculation, the marginal distribution becomes

$$p_k(v) \propto \exp\left(-\frac{1}{2}v^\top\left(I - \sum_{\eta_j>0} W_jW_j^\top - (\beta_k + (1-\beta_k)c)\sum_{\eta_j\leq 0} W_jW_j^\top\right)v\right), \tag{17}$$

which recovers the proposed Algorithm 2. From this analysis, we understand AIS-Leaky is a special case of conventional AIS-Energy with better initialization inspired by the study in Section 3. Also, by this connection between AIS-Energy and AIS-Leaky, we note that AIS-Leaky can be combined with other extensions of AIS (Grosse et al., 2013; Burda et al., 2015) as well.

F More Experimental Results For Sampling

We show the sampled images from leaky RBM train on CIFAR10 and SVHN datasets. We randomly initialize 20 chains and run Gibbs sampling for 1000 iterations. The sampled results are shown in Figure 6 The results shows that single layer RBM does not adequately model CIFAR10 and SVHN when compared to multilayer models. The similar results for single layer Bernoulli-Gaussian RBM from Ranzato & Hinton (2010) (in gray scale) is shown in Figure 7. Therefore, we instead focused on quantitative evaluation of the log-likelihood in Table 3.

The comparison in terms of CPU time of different sampling algorithms discussed in Section 5 is shown in Figure 8. Please note that the complexity of CD and Mix are the almost the same. Mix only need a few more constant time steps which can be ignored compared with sampling steps. Leaky is more time-consuming because of computing and decomposing the covariance matrix as we discussed in Section 5.

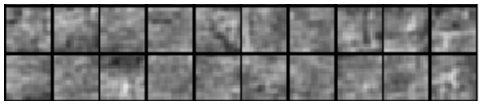
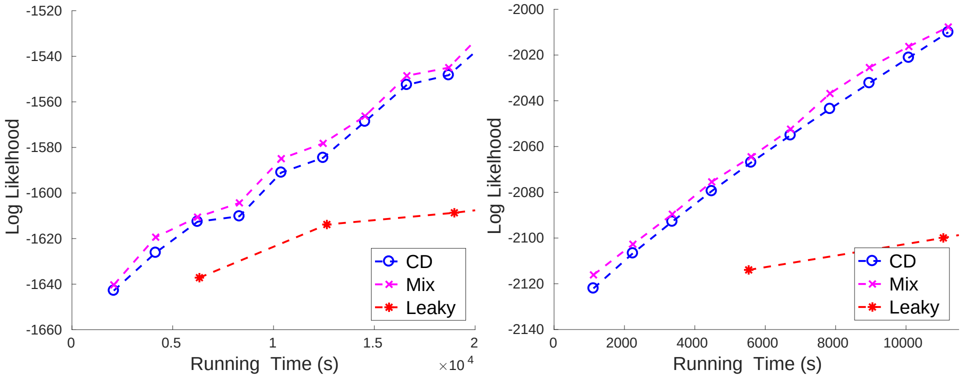


Figure 7: Sampled images in gray-scale from Bernoulli-Gaussian RBM trained on CIFAR10 (Ranzato & Hinton, 2010).



(a) SVHN

(b) CIFAR10

Figure 8: Training leaky RBM with different sampling algorithms.