

Article

Entropy, Free Energy, and Work of Restricted Boltzmann Machines

Sangchul Oh ^{1,*} , Abdelkader Baggag ²  and Hyunchul Nha ³

¹ Qatar Environment and Energy Research Institute, Hamad Bin Khalifa University, Qatar Foundation, 5825 Doha, Qatar

² Qatar Computing Research Institute, Hamad Bin Khalifa University, Qatar Foundation, 5825 Doha, Qatar; abaggag@hbku.edu.qa

³ Department of Physics, Texas A&M University at Qatar, Education City, 23874 Doha, Qatar; hyunchul.nha@qatar.tamu.edu

* Correspondence: soh@hbku.edu.qa

Received: 8 April 2020; Accepted: 8 May 2020; Published: 11 May 2020



Abstract: A restricted Boltzmann machine is a generative probabilistic graphic network. A probability of finding the network in a certain configuration is given by the Boltzmann distribution. Given training data, its learning is done by optimizing the parameters of the energy function of the network. In this paper, we analyze the training process of the restricted Boltzmann machine in the context of statistical physics. As an illustration, for small size bar-and-stripe patterns, we calculate thermodynamic quantities such as entropy, free energy, and internal energy as a function of the training epoch. We demonstrate the growth of the correlation between the visible and hidden layers via the subadditivity of entropies as the training proceeds. Using the Monte-Carlo simulation of trajectories of the visible and hidden vectors in the configuration space, we also calculate the distribution of the work done on the restricted Boltzmann machine by switching the parameters of the energy function. We discuss the Jarzynski equality which connects the path average of the exponential function of the work and the difference in free energies before and after training.

Keywords: restricted Boltzmann machines; entropy; subadditivity of entropy; Jarzynski equality; machine learning

1. Introduction

A restricted Boltzmann machine (RBM) [1] is a generative probabilistic neural network. RBMs and general Boltzmann machines are described by a probability distribution with parameters, i.e., the Boltzmann distribution. An RBM is an undirected Markov random field and is considered a basic building block of deep neural networks. RBMs have been applied widely, for example, in dimensionality reduction, classification, feature learning, pattern recognition, topic modeling, and so on [2–4].

As its name implies, the RBM is closely connected to physics and they share some important concepts such as entropy, free energy, and so forth [5]. Recently, RBMs have gained renewed attention in physics since Carleo and Troyer [6] showed that a quantum many-body state could be efficiently represented by the RBM. Gabré et al. and Tramel et al. [7] employed the Thouless–Anderson–Palmer mean-field approximation, used for a spin glass problem, to replace the Gibbs sampling of contrastive-divergence training. Amin et al. [8] proposed a quantum Boltzmann machine based on the quantum Boltzmann distribution of a quantum Hamiltonian. More interestingly, there is a deep connection between the

Boltzmann machine and tensor networks of quantum many-body systems [9–13]. Xia and Kais combined the restricted Boltzmann machine and quantum computing algorithms to calculate the electronic energy of small molecules [14].

While the working principles of RBMs have been well established, it may still be needed to understand the RBM better for further applications. In this paper, we investigate the RBM from the perspective of statistical physics. As an illustration, for bar-and-stripe pattern data, the thermodynamic quantities such as the entropy, the internal energy, the free energy, and the work are calculated as a function of the epoch. Since the RBM is a bipartite system composed of visible and hidden layers, it may be interesting, and informative, to see how the correlation between the two layers grows as the training goes on. We show that the total entropy of the RBM is always less than the sum of the entropies of visible and hidden layers, except at the initial time when the training begins. This is the so-called subadditivity of entropy, indicating that the visible layer becomes correlated with the hidden layer as the training proceeds. The training of the RBM is to adjust the parameters of the energy function, which can be considered as the work done on the RBM, from a thermodynamic point of view. Using the Monte-Carlo simulation of the trajectories of the visible and hidden vectors in the configuration space, we calculate the work of a single trajectory and the statistics of the work over the ensemble of trajectories. We also examine the Jarzynski equality that connects the ensemble of the work done on the RBM and the difference in free energies before and after the training of the RBM.

The paper is organized as follows. In Section 2, a detailed analysis of the RBM from the statistical physics point of view is described. In Section 3, we present the summary of the result together with discussions.

2. Statistical Physics of Restricted Boltzmann Machines

2.1. Restricted Boltzmann Machines

Let us start with a brief introduction of the RBM [1–3]. As shown in Figure 1, the RBM is composed of two layers; the visible layer and the hidden layer. Possible configurations of the visible and hidden layers are represented by the random binary vectors, $\mathbf{v} = (v_1, \dots, v_N) \in \{0, 1\}^N$ and $\mathbf{h} = (h_1, \dots, h_M) \in \{0, 1\}^M$, respectively. The interaction between the visible and hidden layers is given by the so-called weight matrix $w \in \mathbb{R}^N \times \mathbb{R}^M$, where the weight w_{ij} is the connection strength between a visible unit v_i and a hidden unit h_j . The biases $b_i \in \mathbb{R}$. and $c_j \in \mathbb{R}$ are applied to visible unit i and hidden unit j , respectively. Given random vectors \mathbf{v} and \mathbf{h} , the energy function of the RBM is written as an Ising-type Hamiltonian.

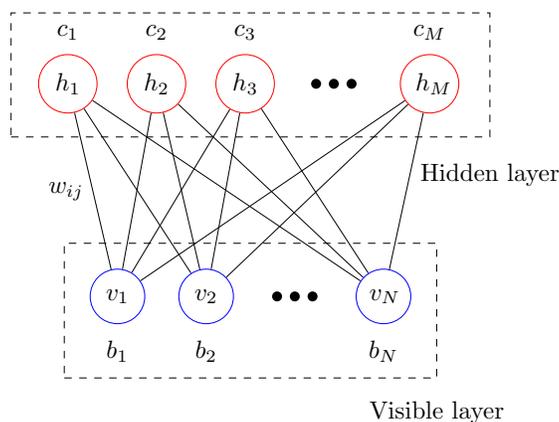


Figure 1. Graph structure of a restricted Boltzmann machine with the visible layer and the hidden layer.

$$E(\mathbf{v}, \mathbf{h}; \theta) = - \sum_{i=1}^N \sum_{j=1}^M w_{ij} v_i h_j - \sum_{i=1}^N b_i v_i - \sum_{i=1}^M c_i h_i, \tag{1}$$

where the set of model parameters is denoted by $\theta \equiv \{w_{ij}, b_i, c_j\}$. The joint probability of finding \mathbf{v} and \mathbf{h} of the RBM in a particular state is given by the Boltzmann distribution

$$p(\mathbf{v}, \mathbf{h}; \theta) = \frac{e^{-E(\mathbf{v}, \mathbf{h}; \theta)}}{Z(\theta)}, \tag{2}$$

where the partition function, $Z(\theta) \equiv \sum_{\mathbf{v}, \mathbf{h}} e^{-E(\mathbf{v}, \mathbf{h}; \theta)}$, is the sum over all possible configurations. The marginal probabilities $p(\mathbf{v}; \theta)$ and $p(\mathbf{h}; \theta)$ for visible and hidden layers are obtained by summing up the hidden or visible variables, respectively,

$$p(\mathbf{v}; \theta) = \sum_{\mathbf{h}} p(\mathbf{v}, \mathbf{h}; \theta) = \frac{1}{Z(\theta)} \sum_{\mathbf{h}} e^{-E(\mathbf{v}, \mathbf{h}; \theta)}, \tag{3}$$

$$p(\mathbf{h}; \theta) = \sum_{\mathbf{v}} p(\mathbf{v}, \mathbf{h}; \theta) = \frac{1}{Z(\theta)} \sum_{\mathbf{v}} e^{-E(\mathbf{v}, \mathbf{h}; \theta)}. \tag{4}$$

The training of the RBM is to adjust the model parameter θ such that the marginal probability of the visible layer $p(\mathbf{v}; \theta)$ becomes as close as possible to the unknown probability $p_{\text{data}}(\mathbf{v})$ that generate the training data. Given identically and independently sampled training data $\mathcal{D} \in \{\mathbf{v}^{(1)}, \dots, \mathbf{v}^{(D)}\}$, the optimal model parameters θ can be obtained by maximizing the likelihood function of the parameters, $\mathcal{L}(\theta|\mathcal{D}) = \prod_{i=1}^D p(\mathbf{v}^{(i)}; \theta)$, or equivalently by maximizing the log-likelihood function $\ln \mathcal{L}(\theta|\mathcal{D}) = \sum_{i=1}^D \ln p(\mathbf{v}^{(i)}; \theta)$. Maximizing the likelihood function is equivalent to minimizing the Kullback–Leibler divergence or the relative entropy of $p(\mathbf{v}; \theta)$ from $q(\mathbf{v})$ [15,16]

$$D_{\text{KL}}(q || p) = \sum_{\mathbf{v}} q(\mathbf{v}) \ln \frac{q(\mathbf{v})}{p(\mathbf{v}; \theta)}, \tag{5}$$

where $q(\mathbf{v})$ is an unknown probability that generates the training data, $q(\mathbf{v}) = p_{\text{data}}(\mathbf{v})$. Another method of monitoring the progress of training is the cross-entropy cost between the input visible vector $\mathbf{v}^{(i)}$ and a reconstructed visible vector $\bar{\mathbf{v}}^{(i)}$ of the RBM,

$$C = -\frac{1}{D} \sum_{i \in \mathcal{D}} \left[\mathbf{v}^{(i)} \ln \bar{\mathbf{v}}^{(i)} + (1 - \mathbf{v}^{(i)}) \ln(1 - \bar{\mathbf{v}}^{(i)}) \right]. \tag{6}$$

The stochastic gradient ascent method for the log-likelihood function is used to train the RBM. Estimating the log-likelihood function requires the Monte-Carlo sampling for the model probability distribution. Well-known sampling methods are the contrastive-divergence, denoted by CD- k , and the persistent contrastive divergence PCD- k . For details of the RBM algorithm, please see References [2–4]. Here, we employ the CD- k method.

2.2. Free Energy, Entropy, and Internal Energy

From a physics point of view, the RBM is a finite classical system composed of two subsystems, similar to an Ising spin system. The training of the RBM is considered the driving of the system from an initial equilibrium state to the target equilibrium state by switching the model parameters. It may be interesting to see how thermodynamic quantities such as free energy, entropy, internal energy, and work change as the training progresses.

It is straightforward to write down various thermodynamic quantities for the total system. The free energy F is given by the logarithm of the partition function Z ,

$$F(\theta) = -\ln Z(\theta). \tag{7}$$

The internal energy U is given by the expectation value of the energy function $E(\mathbf{v}, \mathbf{h}; \theta)$

$$U(\theta) = \sum_{\mathbf{v}, \mathbf{h}} E(\mathbf{v}, \mathbf{h}; \theta) p(\mathbf{v}, \mathbf{h}; \theta). \tag{8}$$

The entropy S of the total system comprising the hidden and visible layers is given by

$$S(\theta) = -\sum_{\mathbf{v}, \mathbf{h}} p(\mathbf{v}, \mathbf{h}; \theta) \ln p(\mathbf{v}, \mathbf{h}; \theta). \tag{9}$$

Here, the convention of $0 \ln 0 = 0$ is employed if $p(\mathbf{v}, \mathbf{h}) = 0$ [17]. The free energy (7) is related to the difference between the internal energy (9) and the entropy (10)

$$F = U - TS, \tag{10}$$

where T is set to 1.

Generally, it is very challenging to calculate the thermodynamic quantities, even numerically. The number of possible configurations of N visible units and M hidden units grow exponentially as 2^{N+M} . Here, for a feasible benchmark test, the 2×2 bar-and-stripe data are considered [18,19]. Figure 2 shows the 6 possible 2×2 bar-and-stripe patterns out of 16 possible configurations, which will be used as the training data in this work. We take the sizes of the visible and the hidden layers as $N = 4$ and $M = 6$, respectively. One may take a larger size of hidden layers, i.e., $M = 8$ or 10 , but it does not make an appreciable difference in our results. $M = 6$ is not a choice of magic number but was used as an example since we were rather limited in our capacity of numerical computation. In order to understand better how the RBM is trained, the thermodynamic quantities are calculated numerically for this small benchmark system.

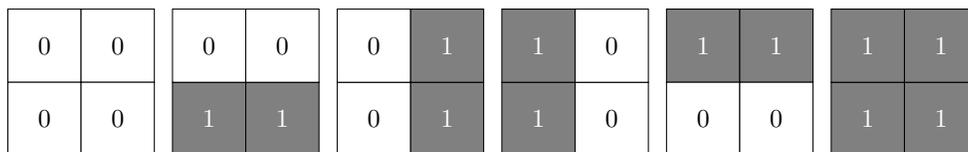


Figure 2. Six samples of 2×2 bar-and-stripe patterns used as the training data in this work. Each configuration is represented by a visible vector $\mathbf{v} \in \{0,1\}^{2 \times 2}$ or by a decimal number; $(0,0,0,0) = 0$, $(0,0,1,1) = 3$, $(0,1,0,1) = 5$, $(1,0,1,0) = 10$, $(1,1,0,0) = 12$, $(1,1,1,1) = 15$ in row-major ordering.

Figure 3 shows how the weight w_{ij} , the bias b_i on the visible unit i and the bias c_j on the hidden unit j change as the training goes on. The weights w_{ij} are clustered into three classes. The evolution of the bias b_i on the visible layer is somewhat different from that of the bias c_j on the hidden layer. The change in c_i is larger than that in b_i . Figure 4 shows the change in the marginal probabilities $p(\mathbf{v})$ of the visible layer and $p(\mathbf{h})$ of the hidden layer before and after training. Note that the marginal probability $p(\mathbf{v})$ after training is not distributed exclusively over six possible outcomes corresponding to the training data set in Figure 2.

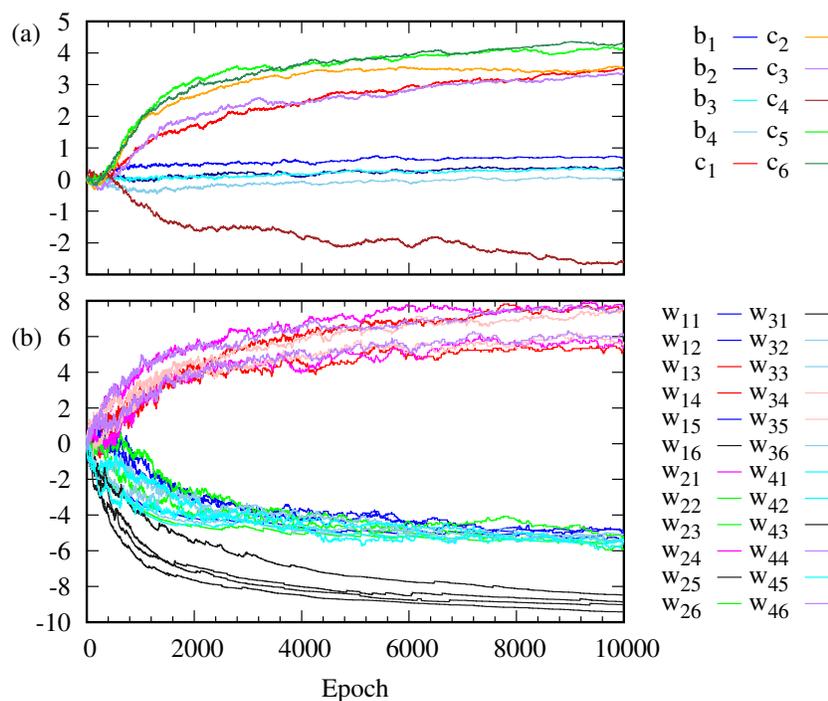


Figure 3. (a) Bias b_i on the visible unit i and bias c_j on the hidden unit j are plotted as a function of the epoch. (b) Weight w_{ij} connecting the visible unit i and the hidden unit j are plotted as a function of the epoch.

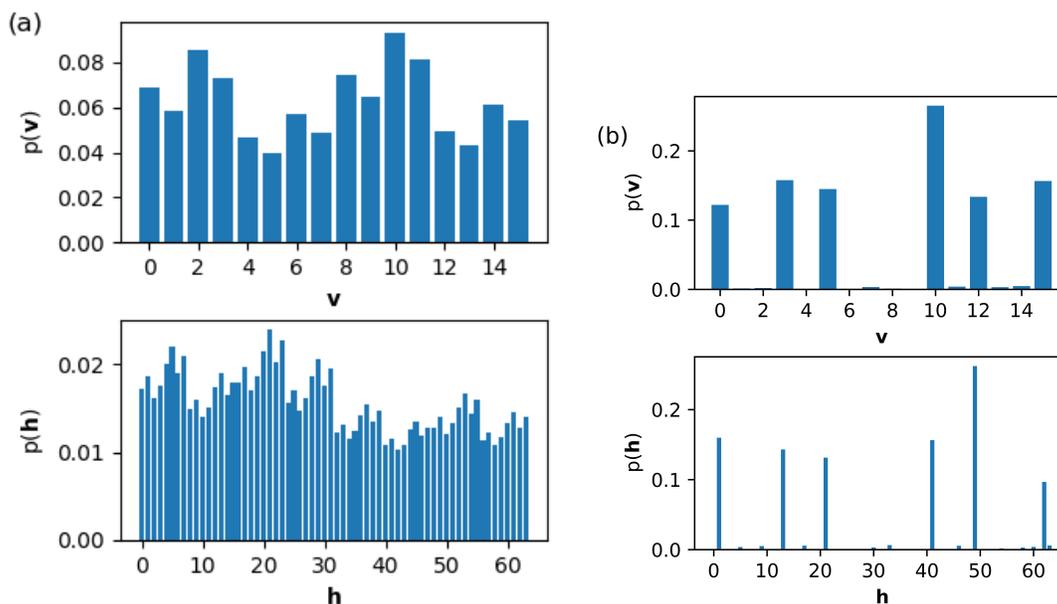


Figure 4. Marginal probabilities $p(\mathbf{v})$ of visible layer and $p(\mathbf{h})$ of hidden layer are plotted (a) before training and (b) after training. The binary vector \mathbf{v} or \mathbf{h} in the x-axis is represented by the decimal number as noted in the caption of Figure 2. The visible and the hidden layers have a total number of configurations given by $2^4 = 16$ and $2^6 = 64$, respectively. The learning rate is 0.15, the training epoch 20000, and $k = 100$ in CD- k .

Typically, the progress of learning of the RBM is monitored by the loss function. Here, the Kullback–Leibler divergence, Equation (5), and the reconstructed cross entropy, Equation (6), are used. Figure 5 plots the reconstructed cross entropy C , the Kullback–Leibler divergence D_{KL} , the entropy S , the free energy F , and the internal energy U as a function of the epoch. As shown in Figure 5a, it is interesting to see that even after a large number of epochs $\sim 10,000$, the cost function C continues approaching zero while the entropy S and the Kullback–Leibler divergence D_{KL} become steady. On the other hand, the free energy F continues decreasing together with the internal energy U , as depicted in Figure 5b. The Kullback–Leibler divergence is a well-known indicator of the performance of RBMs. Then, our result implies that the entropy may be another good indicator to monitor the progress of the RBM while other thermodynamic quantities may be not.

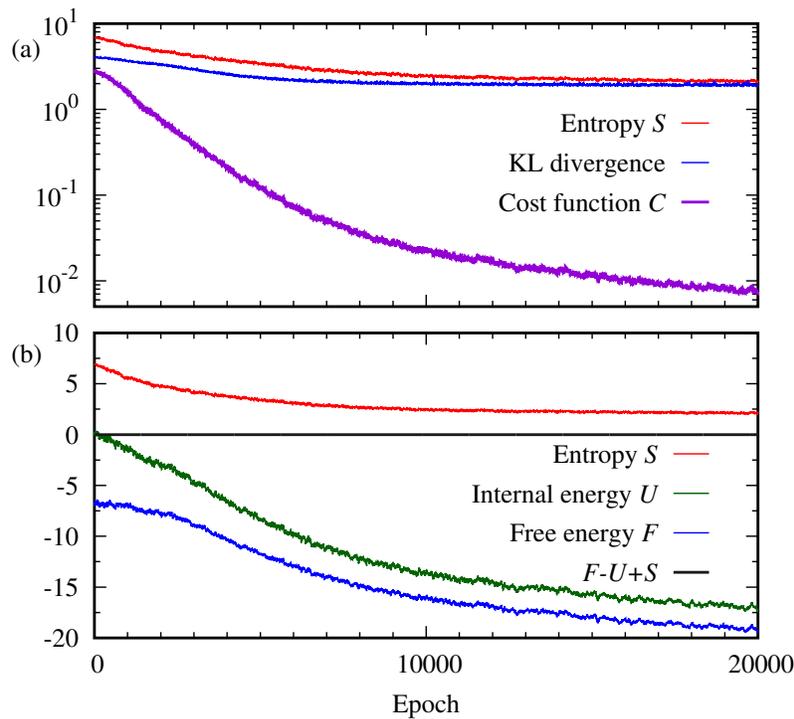


Figure 5. For 2×2 bar-and-stripe data, (a) cost function C , entropy S , and the Kullback–Leibler divergence $D_{KL}(q || p)$ are plotted as a function of the epoch. (b) Free energy F , entropy S , and internal energy U of the RBM are calculated as a function of the epoch.

In addition to the thermodynamic quantities of the total system of the RBM, Equations (7)–(9), it is interesting to see how the two subsystems of the RBM evolve. Since the RBM has no intra-layer connection, the correlation between the visible layer and the hidden layer may increase as the training proceeds. The correlation between the visible layer and the hidden layer can be measured by the difference between the total entropy and the sum of the entropies of the two subsystems. The entropies of the visible and hidden layers are given by

$$S_V = - \sum_{\mathbf{v}} p(\mathbf{v}; \theta) \ln p(\mathbf{v}; \theta), \tag{11}$$

$$S_H = - \sum_{\mathbf{h}} p(\mathbf{h}; \theta) \ln p(\mathbf{h}; \theta). \tag{12}$$

The entropy S_V of the visible layer is closely related to the Kullback–Leibler divergence of $p(\mathbf{v}; \theta)$ to an unknown probability $q(\mathbf{v})$ which produces the data. Equation (5) is expanded as

$$D_{KL}(q || p) = \sum_{\mathbf{v}} q(\mathbf{v}) \ln q(\mathbf{v}) - \sum_{\mathbf{v}} q(\mathbf{v}) \ln p(\mathbf{v}; \theta). \tag{13}$$

The second term $-\sum_{\mathbf{v}} q(\mathbf{v}) \ln p(\mathbf{v}; \theta)$ depends on the parameter θ . As the training proceeds, $p(\mathbf{v}; \theta)$ becomes close to $q(\mathbf{v})$ so the behavior of the second term is very similar to that of the entropy S_V of the visible layer. If the training is perfect, we have $q(\mathbf{v}) = p(\mathbf{v}; \theta)$ that leads to $D_{KL}(q || p) = 0$ while S_V remains nonzero.

The difference between the total entropy and the sum of the entropies of subsystems is written as

$$S - (S_V + S_H) = \sum_{\mathbf{v}, \mathbf{h}} p(\mathbf{v}, \mathbf{h}) \ln \left[\frac{p(\mathbf{v}) p(\mathbf{h})}{p(\mathbf{v}, \mathbf{h})} \right]. \tag{14}$$

Equation (14) tells us that if the visible random vector \mathbf{v} and the hidden random vector \mathbf{h} are independent, i.e., $p(\mathbf{v}, \mathbf{h}; \theta) = p(\mathbf{v}; \theta)p(\mathbf{h}; \theta)$, then the entropy S of the total system is the sum of the entropies of subsystems. In general, the entropy S of the total system is always less than or equal to the sum of the entropy of the visible layer, S_V , and the entropy of the hidden layer, S_H , [20],

$$S \leq S_V + S_H. \tag{15}$$

This is called the subadditivity of entropy, one of the basic properties of the Shannon entropy, which is also valid for the von Neumann entropy [17,21]. This property can be proved using the log inequality, $-\ln x \geq -x + 1$. In another way, Equation (15) may be proved by using the log-sum inequality, which states that for the two sets of nonnegative numbers, a_1, \dots, a_n and b_1, \dots, b_n ,

$$\sum_i a_i \log \frac{a_i}{b_i} \geq \left(\sum_i a_i \right) \log \frac{(\sum_i a_i)}{(\sum_i b_i)}. \tag{16}$$

In other words, Equation (14) can be regarded as the negative of the relative entropy or Kullback–Leibler divergence of the joint probability $p(\mathbf{v}, \mathbf{h})$ to the product probability $p(\mathbf{v}) \cdot p(\mathbf{h})$,

$$I(p(\mathbf{v}, \mathbf{h}) || p(\mathbf{v})p(\mathbf{h})) = \sum_{\mathbf{v}, \mathbf{h}} p(\mathbf{v}, \mathbf{h}) \log \left[\frac{p(\mathbf{v}, \mathbf{h})}{p(\mathbf{v})p(\mathbf{h})} \right]. \tag{17}$$

For the 2×2 bar-and-stripe pattern, the entropies of visible and hidden layers, S_V, S_H are calculated numerically. Figure 6 plots the entropies, S_V, S_H, S , and the Kullback–Leibler divergence $D_{KL}(q || p)$ as a function of the epoch. Figure 6a shows that the Kullback–Leibler divergence, $D_{KL}(q || p)$ becomes saturated, though above zero, as the training proceeds. Similarly, the entropy S_V of the visible layer is saturated. This implies that the entropy of the visible layer, as well as the total entropy shown in Figure 5, can be a better indicator of learning than the reconstructed cross entropy C , Equation (6). The same can also be said about the entropy of the hidden layer, S_H . If some information measures such as entropy and the Kullback–Leibler divergence become steady, one may presume the training has been done.

The difference between the total entropy and the sum of the entropies of the two subsystems, $S - (S_V + S_H)$, becomes less than 0, as shown in Figure 6b. Thus, it demonstrates the subadditivity of entropy, i.e., the correlation between the visible and the hidden layer as the training proceeds. As it is saturated just as the total entropy and the entropies of the visible and hidden layers after a large number

of epochs, the correlation between the visible layer and the hidden layer can also be a good quantifier of the RBM progress.

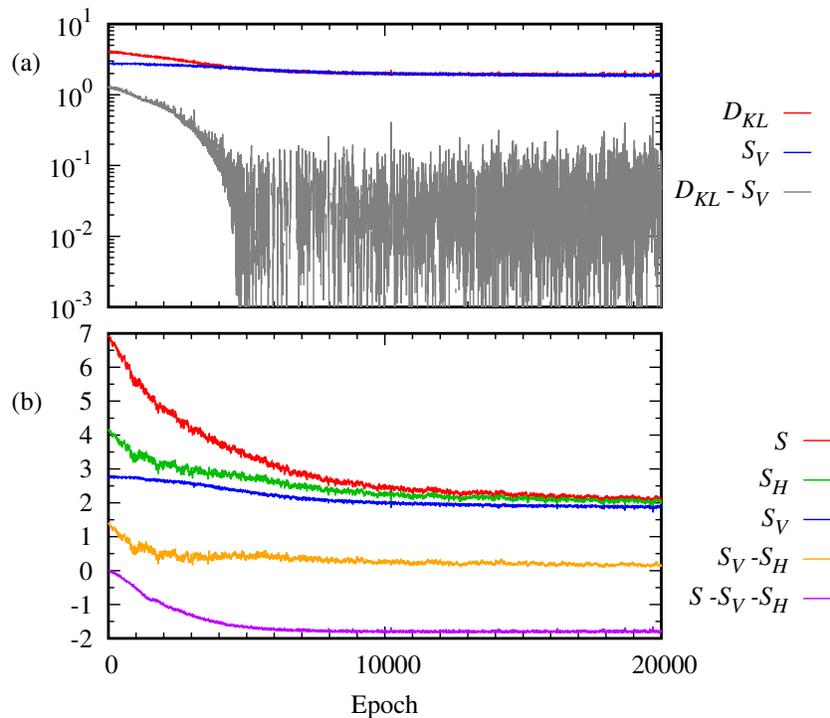


Figure 6. (a) Kullback–Leibler divergence $D_{KL}(q || p)$, entropy S_V , and their difference are plotted as a function of the epoch. (b) Entropy S of the total system, entropy S_V of the visible layer, entropy S_H of the hidden layer, and the difference $S - S_H - S_V$ are plotted as a function of the epoch.

2.3. Work, Free Energy, and Jarzynski Equality

The training of the RBM may be viewed as driving a finite classical spin system from an initial equilibrium state to a final equilibrium state by changing the system parameters θ slowly. If the parameters θ are switched infinitely slowly, the classical system remains in a quasi-static equilibrium. In this case, the total work done on the systems is equal to the Helmholtz free energy difference between the before-training and the after-training, $W_\infty = F_1 - F_0$. For switching θ at a finite rate, the system may not evolve immediately to an equilibrium state, the work done on the system depends on a specific path of the system in the configuration space. Jarzynski [22,23] proved that for any switching rate, the free energy difference ΔF is related to the average of the exponential function of the amount of work W over the paths

$$\langle e^{-W} \rangle_{\text{path}} = e^{-\Delta F}. \tag{18}$$

The RBM is trained by changing the parameters θ through a sequence $\{\theta_0, \theta_1, \dots, \theta_\tau\}$, as shown in Figure 3. To calculate the work done during the training, we perform the Monte-Carlo simulation of the trajectory of a state (\mathbf{v}, \mathbf{h}) of the RBM in configuration space. From the initial configuration, $(\mathbf{v}_0, \mathbf{h}_0)$ which is sampled from the initial Boltzmann distribution, Equation (2), the trajectory $(\mathbf{v}_0, \mathbf{h}_0) \rightarrow (\mathbf{v}_1, \mathbf{h}_1) \rightarrow \dots \rightarrow (\mathbf{v}_\tau, \mathbf{h}_\tau)$ is obtained using the Metropolis–Hastings algorithm of the Markov chain Monte-Carlo

method [24,25]. Assuming the evolution is Markovian, the probability of taking a specific trajectory is the product of the transition probabilities at each step,

$$p(\mathbf{v}_0, \mathbf{h}_0 \xrightarrow{\theta_1} \mathbf{v}_1, \mathbf{h}_1) p(\mathbf{v}_1, \mathbf{h}_1 \xrightarrow{\theta_2} \mathbf{v}_2, \mathbf{h}_2) \dots p(\mathbf{v}_{\tau-1}, \mathbf{h}_{\tau-1} \xrightarrow{\theta_\tau} \mathbf{v}_\tau, \mathbf{h}_\tau). \tag{19}$$

The transition $(\mathbf{v}, \mathbf{h}) \rightarrow (\mathbf{v}', \mathbf{h}')$ can be implemented by the Metropolis–Hastings algorithm based on the detailed balance condition for the fixed parameter θ ,

$$\frac{p(\mathbf{v}, \mathbf{h} \xrightarrow{\theta} \mathbf{v}', \mathbf{h}')}{p(\mathbf{v}, \mathbf{h} \xleftarrow{\theta} \mathbf{v}', \mathbf{h}')} = \frac{e^{-E(\mathbf{v}', \mathbf{h}'; \theta)}}{e^{-E(\mathbf{v}, \mathbf{h}; \theta)}}. \tag{20}$$

The work done on the RBM at epoch i may be given by

$$\delta W_i = E(\mathbf{v}_i, \mathbf{h}_i; \theta_{i+1}) - E(\mathbf{v}_i, \mathbf{h}_i; \theta_i). \tag{21}$$

The total work $W = \sum \delta W_i$ performed on the system is written as [26]

$$W = \sum_{i=0}^{\tau-1} [E(\mathbf{v}_i, \mathbf{h}_i; \theta_{i+1}) - E(\mathbf{v}_i, \mathbf{h}_i; \theta_i)]. \tag{22}$$

Given the sequence of the model parameter $\{\theta_0, \theta_1, \dots, \theta_\tau\}$, the Markov evolution of the visible and hidden vectors $(\mathbf{v}, \mathbf{h}) \in \{0, 1\}^{N+M}$ may be considered the discrete random walk. Random walkers move to the points with low energy in configuration space. Figure 7 shows the heat map of energy function $E(\mathbf{v}, \mathbf{h}; \theta)$ of the RBM for the 2×2 bar-and-stripe pattern after training. One can see the energy function has deep levels at the visible vectors corresponding to the bar-and-stripe patterns of the training data set in Figure 2, representing a high probability of generating the trained patterns. Furthermore, note that the energy function has many local minima. Figure 8 plots a few Monte-Carlo trajectories of the visible vector \mathbf{v} as a function of the epoch. Before training, the visible vector \mathbf{v} is distributed over all possible configurations, represented by the number $(0, \dots, 15)$. As the training progresses, the visible vector \mathbf{v} becomes trapped into one of the six possible outcomes $(0, 3, 5, 10, 12, 15)$.

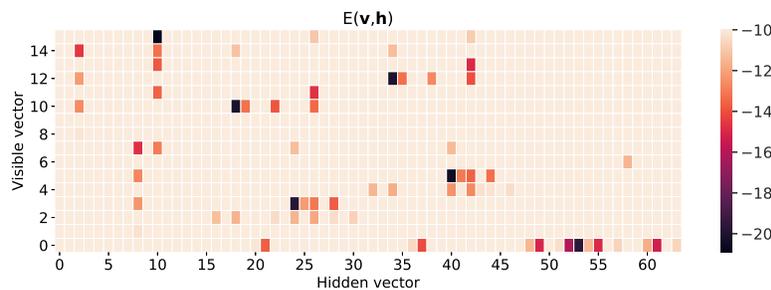


Figure 7. Heat map of energy function $E(\mathbf{v}, \mathbf{h}; \theta)$, representing the energy level of each configuration, after training of 2×2 bar-and-stripe patterns for 50000 epochs. The sizes of visible and hidden layers are $N = 4$ and $M = 6$, respectively. The learning rate is $r = 0.15$ and the value of k in CD- k is 100. The vertical and the horizontal axes represent each configuration of the visible and the hidden layers, respectively. The black tiles represent the lowest energy configurations among all configurations, thus the probability of finding that configuration is high.

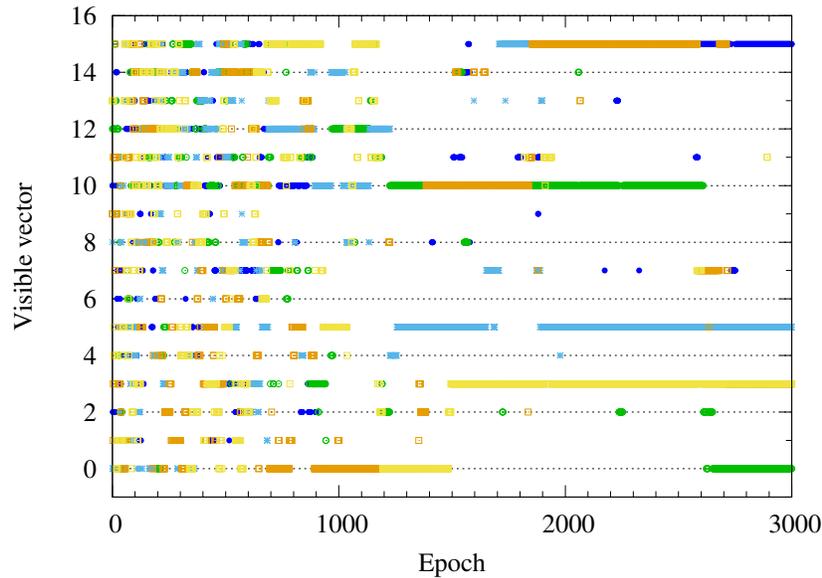


Figure 8. Markov chain Monte-Carlo trajectories of the visible vector v_i are plotted as a function of the epoch. The visible vector jumps frequently in the early state of training and becomes trapped into one of the target states as the training proceeds.

In order to examine the relation between work done on the RBM during the training and the free energy difference, the Monte-Carlo simulation is performed to calculate the average of the work over paths generated by the Metropolis–Hastings algorithm of the Markov chain Monte-Carlo method. Each path starts from an initial state sampled from the uniform distribution over the configuration space, as shown in Figure 4a. Since the work done on the system depends on the path, the distribution of the work is calculated by generating many trajectories. Figure 9 shows the distribution of the work over 50000 paths at 5000 training epochs. The Monte-Carlo average of the work is $\langle W \rangle \approx -5.481$, and its standard deviation is $\sigma_W \approx 3.358$. The distribution of the work generated by the Monte-Carlo simulation is well fitted to the Gaussian distribution, as depicted by the red curve in Figure 9. This agrees with the statement in Reference [23] that for the slow switching of the model parameters the probability distribution of work is approximated to the Gaussian.

We perform the Monte-Carlo calculation of the exponential average of work, $\langle e^{-W} \rangle_{\text{path}}$ to check the Jarzynski equality, Equation (18). The free energy difference can be estimated as

$$e^{-\Delta F} = \langle e^{-W} \rangle_{\text{path}} \approx \frac{1}{N_{\text{mc}}} \sum_{n=1}^{N_{\text{mc}}} e^{-W_n}, \tag{23}$$

where N_{mc} is the number of the Monte-Carlo samplings. At a small epoch number, the Monte-Carlo estimated value of the free energy difference is close to ΔF calculated from the partition function. However, this Monte-Carlo calculation gives rise to the poor estimation of the free energy difference if the epoch is greater than 5000. This numerical errors can be explained by the fact that the exponential average of the work is dominated by rare realization [27–31]. As shown in Figure 9, the distribution of work is given by the Gaussian distribution $\rho(W)$ with the mean $\langle W \rangle$ and the standard deviation σ_W . If the standard deviation σ_W becomes larger, the peak position of $\rho(W)e^{-W}$ moves to the long tail of the Gaussian distribution.

So the main contribution of the integration of $\langle e^{-W} \rangle$ comes from the rare realizations. Figure 10 shows that the standard deviation σ_W grows with the epoch, so the error of the Monte-Carlo estimation of the exponential average of the work grows quickly.

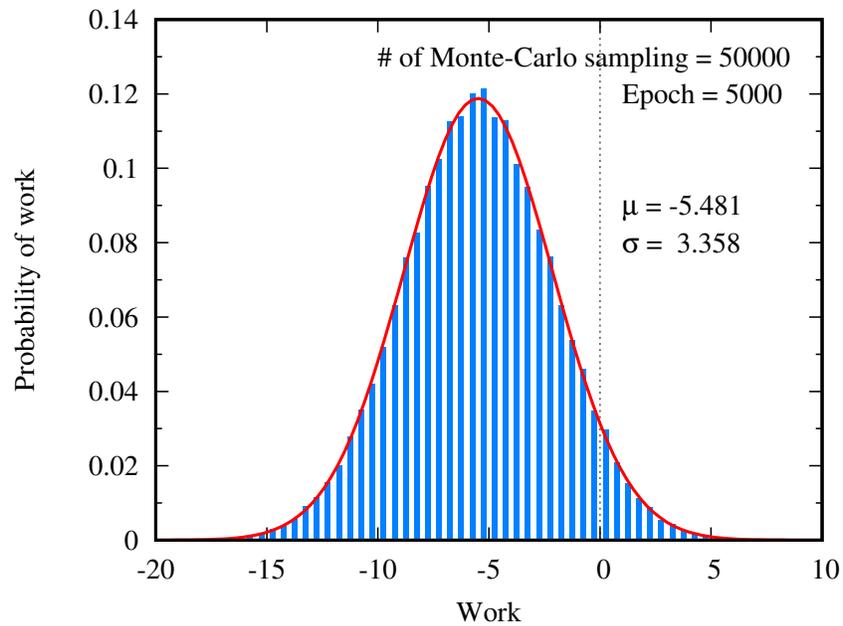


Figure 9. Gaussian distribution of work done by the restricted Boltzmann machine (RBM) during the training. The number of the Monte-Carlo sampling is 50000. The red curve is the plot of the Gaussian distribution using the mean and the standard deviation calculated by the Monte-Carlo simulation.

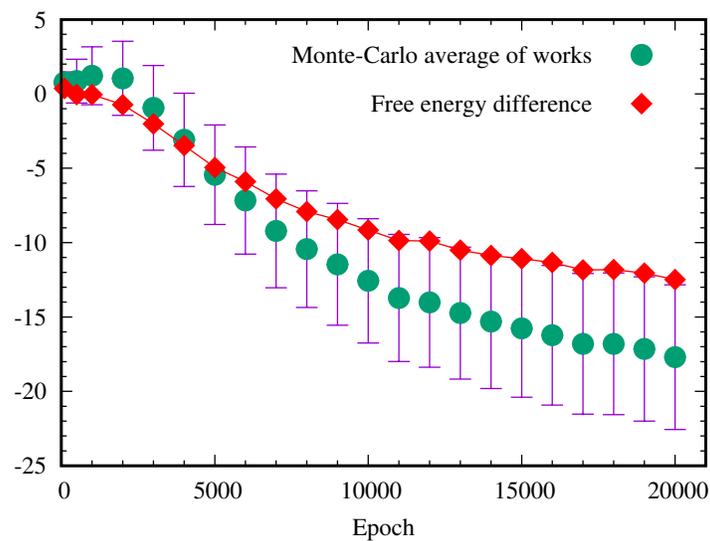


Figure 10. Average of work done with standard deviation and free energy difference $\Delta F = F(\text{epoch}) - F(\text{epoch} = 0)$ as a function of the epoch. The error bar of the work represents the standard deviation of the Gaussian distribution.

If $\sigma_W^2 \ll k_B T$, the free energy is related to the average of work and its variance as

$$\Delta F = \langle W \rangle_{\text{path}} - \frac{\sigma_W^2}{2k_B T}. \quad (24)$$

Here, the case is the opposite, the spread of the value of work is large, i.e., $\sigma_W^2 \gg k_B T$ ($= 1$), so the central limit theorem does not work and the above equation can not be applied [32]. Figure 10 shows how the average of work, $\langle W \rangle_{\text{path}}$, over the Markov chain Monte-Carlo paths changes as a function of the epoch. The standard deviation of the Gaussian distribution of the work also grows as a function of the training epoch. The free energy difference between before-training and after-training is called the reversible work $W_r = \Delta F$. The difference between the actual work and the reversible work is called the dissipative work, $W_d = W - W_r$ [26]. As depicted in Figure 10, the magnitude of the dissipative work grows with the training epoch.

3. Summary

In summary, we analyzed the training process of the RBM in the context of statistical physics. In addition to the typical loss function, i.e., the reconstructed cross entropy, the thermodynamic quantities such as free energy F , internal energy U , and entropy S were calculated as a function of the epoch. While the free energy and the internal energy decrease rather indefinitely with epochs, the total entropy and the entropies of the visible and the hidden layers become saturated together with the Kullback–Leibler divergence after a sufficient number of epochs. This result suggests that the entropy of the system may be a good indicator of the RBM progress along with the Kullback–Leibler divergence. It seems worth investigating the entropy for other larger data sets, for example, MNIST handwritten digits [33], in future works.

We have further demonstrated the subadditivity of the entropy, i.e., the entropy of the total system is less than the sum of the entropies of the two layers. This manifested the correlation between the visible and hidden layers growing with the training progress. Just as the entropies are well saturated together with the Kullback–Leibler divergence, so is the correlation that is determined by the total and the local entropies. In this sense, the correlation between the visible and the hidden layer may become another good indicator of the RBM performance.

We also investigated the work done on the RBM by switching the parameters of the energy function. The trajectories of the visible and hidden vectors in the configuration space were generated using the Markov chain Monte-Carlo simulation. The distribution of the work follows the Gaussian distribution and its standard deviation grows with the training epochs. We discussed the Jarzynski equality, which connects the free energy difference and the average of the exponential function of the work over the trajectories. We note that, in addition to the Jarzynski equality, the Crooks path-ensemble average method [34,35] with the forward and backward transformations could be also used to connect the free energy difference and the work. This is called the bidirectional estimator [36] in contrast to the unidirectional estimator such as the Jarzynski equality or the Hummer–Szabo method [37].

A more detailed analysis from a full thermodynamics or statistical physics point of view can bring us useful insights into the performance of the RBM. This course of study may enable us to come up with possible methods for a better performance of the RBM for many different applications in the long run. Therefore, it may be worthwhile to further pursue our study, e.g., a rigorous assessment of scaling behavior of thermodynamic quantities with respect to epochs as the sizes of the visible and hidden layers increase. We also expect that a similar analysis on a quantum Boltzmann machine can be valuable as well.

Author Contributions: Conceptualization, S.O.; data curation, S.O.; formal analysis, S.O., A.B., and H.N.; investigation, S.O.; writing—original draft, S.O.; writing—review and editing, S.O., A.B., and H.N. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Smolensky, P. Information processing in dynamical systems: Foundations of harmony theory. In *Parallel Distributed Processing: Explorations in The Microstructure of Cognition*; Rumelhart, D., McClelland, J., Eds.; MIT Press: Cambridge, MA, USA, 1986; pp. 194–281.
2. Hinton, G.E. A Practical Guide to Training Restricted Boltzmann Machines. In *Neural Networks: Tricks of the Trade: Second Edition*; Montavon, G., Orr, G.B., Müller, K.R., Eds.; Springer Berlin Heidelberg: Berlin/Heidelberg, Germany, 2012; pp. 599–619. [[CrossRef](#)]
3. Fischer, A.; Igel, C. Training restricted Boltzmann machines: An introduction. *Pattern Recognit.* **2014**, *47*, 25–39. [[CrossRef](#)]
4. Melchior, J.; Fischer, A.; Wiskott, L. How to Center Deep Boltzmann Machines. *J. Mach. Learn. Res.* **2016**, *17*, 1–61.
5. Mehta, P.; Bukov, M.; Wang, C.H.; Day, A.G.R.; Richardson, C.; Fisher, C.K.; Schwab, D.J. A high-bias, low-variance introduction to Machine Learning for physicists. *Phys. Rep.* **2019**, *810*, 1–124. [[CrossRef](#)] [[PubMed](#)]
6. Carleo, G.; Troyer, M. Solving the quantum many-body problem with artificial neural networks. *Science* **2017**, *355*, 602–606. [[CrossRef](#)] [[PubMed](#)]
7. Tramel, E.W.; Gabrié, M.; Manoel, A.; Caltagirone, F.; Krzakala, F. Deterministic and Generalized Framework for Unsupervised Learning with Restricted Boltzmann Machines. *Phys. Rev. X* **2018**, *8*, 041006. [[CrossRef](#)]
8. Amin, M.H.; Andriyash, E.; Rolfe, J.; Kulchitsky, B.; Melko, R. Quantum Boltzmann Machine. *Phys. Rev. X* **2018**, *8*, 021050. [[CrossRef](#)]
9. Stoudenmire, E.; Schwab, D.J. Supervised Learning with Tensor Networks. In *Advances in Neural Information Processing Systems 29*; Lee, D.D., Sugiyama, M., Luxburg, U.V., Guyon, I., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2016; pp. 4799–4807.
10. Gao, X.; Duan, L.M. Efficient representation of quantum many-body states with deep neural networks. *Nat. Commun.* **2017**, *8*, 662. [[CrossRef](#)]
11. Chen, J.; Cheng, S.; Xie, H.; Wang, L.; Xiang, T. Equivalence of restricted Boltzmann machines and tensor network states. *Phys. Rev. B* **2018**, *97*, 085104. [[CrossRef](#)]
12. Das Sarma, S.; Deng, D.L.; Duan, L.M. Machine learning meets quantum physics. *Phys. Today* **2019**, *72*, 48–54. [[CrossRef](#)]
13. Huggins, W.; Patil, P.; Mitchell, B.; Whaley, K.B.; Stoudenmire, E.M. Towards quantum machine learning with tensor networks. *Quantum Sci. Technol.* **2019**, *4*, 024001. [[CrossRef](#)]
14. Xia, R.; Kais, S. Quantum machine learning for electronic structure calculations. *Nat. Commun.* **2018**, *9*, 4195. [[CrossRef](#)] [[PubMed](#)]
15. Kullback, S.; Leibler, R.A. On Information and Sufficiency. *Ann. Math. Statist.* **1951**, *22*, 79–86. [[CrossRef](#)]
16. Cover, T.M.; Thomas, J.A. *Elementary Information Theory*, 2 ed.; Wiley: New York, NY, USA, 2006.
17. Nielsen, M.A.; Chuang, I.L. *Quantum Computation and Quantum Information*; Cambridge University Press: New York, NY, USA, 2000.
18. Hinton, G.E.; Sejnowski, T.J., Learning and relearning in Boltzmann machines. In *Parallel Distributed Processing: Explorations in The Microstructure of Cognition*; Rumelhart, D.E., McClelland, J.L., Eds.; MIT Press: Cambridge, MA, USA, 1986; pp. 282–317.
19. MacKay, D.J.C. *Information Theory, Inference & Learning Algorithms*; Cambridge University Press: New York, NY, USA, 2002.

20. Reif, F. *Fundamentals of Statistical and Thermal Physics*; McGraw Hill: New York, NY, USA, 1965.
21. Araki, H.; Lieb, E.H. Entropy inequalities. *Commun. Math. Phys.* **1970**, *18*, 160–170. [[CrossRef](#)]
22. Jarzynski, C. Nonequilibrium Equality for Free Energy Differences. *Phys. Rev. Lett.* **1997**, *78*, 2690–2693. [[CrossRef](#)]
23. Jarzynski, C. Equalities and Inequalities: Irreversibility and the Second Law of Thermodynamics at the Nanoscale. *Annu. Rev. Condens. Matter Phys.* **2011**, *2*, 329–351. [[CrossRef](#)]
24. Metropolis, N.; Rosenbluth, A.W.; Rosenbluth, M.N.; Teller, A.H.; Teller, E. Equation of State Calculations by Fast Computing Machines. *J. Chem. Phys.* **1953**, *21*, 1087–1092. [[CrossRef](#)]
25. Hastings, W.K. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika* **1970**, *57*, 97–109. [[CrossRef](#)]
26. Crooks, G.E. Nonequilibrium Measurements of Free Energy Differences for Microscopically Reversible Markovian Systems. *J. Stat. Phys.* **1998**, *90*, 1481–1487. [[CrossRef](#)]
27. Jarzynski, C. Rare events and the convergence of exponentially averaged work values. *Phys. Rev. E* **2006**, *73*, 046105. [[CrossRef](#)]
28. Zuckerman, D.M.; Woolf, T.B. Theory of a Systematic Computational Error in Free Energy Differences. *Phys. Rev. Lett.* **2002**, *89*, 180602. [[CrossRef](#)]
29. Lechner, W.; Oberhofer, H.; Dellago, C.; Geissler, P.L. Equilibrium free energies from fast-switching trajectories with large time steps. *J. Chem. Phys.* **2006**, *124*, 044113. [[CrossRef](#)]
30. Lechner, W.; Dellago, C. On the efficiency of path sampling methods for the calculation of free energies from non-equilibrium simulations. *J. Stat. Mech. Theory Exp.* **2007**, *2007*, P04001. [[CrossRef](#)]
31. Yunger Halpern, N.; Jarzynski, C. Number of trials required to estimate a free-energy difference, using fluctuation relations. *Phys. Rev. E* **2016**, *93*, 052144. [[CrossRef](#)] [[PubMed](#)]
32. Hendrix, D.A.; Jarzynski, C. A fast growth method of computing free energy differences. *J. Chem. Phys.* **2001**, *114*, 5974–5981. [[CrossRef](#)]
33. LeCun, Y.; Cortes, C.; Burges, C. MNIST Handwritten Digit Database. Available online: <http://yann.lecun.com/exdb/mnist/> (accessed on 15 March 2020).
34. Crooks, G.E. Entropy production fluctuation theorem and the nonequilibrium work relation for free energy differences. *Phys. Rev. E* **1999**, *60*, 2721–2726. [[CrossRef](#)] [[PubMed](#)]
35. Crooks, G.E. Path-ensemble averages in systems driven far from equilibrium. *Phys. Rev. E* **2000**, *61*, 2361–2366. [[CrossRef](#)]
36. Minh, D.D.L.; Adib, A.B. Optimized Free Energies from Bidirectional Single-Molecule Force Spectroscopy. *Phys. Rev. Lett.* **2008**, *100*, 180602. [[CrossRef](#)]
37. Hummer, G.; Szabo, A. Free energy reconstruction from nonequilibrium single-molecule pulling experiments. *Proc. Natl. Acad. Sci. USA* **2001**, *98*, 3658–3661. [[CrossRef](#)]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).