MACHINE LEARNING IN QUANTUM COMPUTERS VIA GENERAL BOLTZMANN MACHINES: GENERATIVE AND DISCRIMINATIVE TRAINING THROUGH ANNEALING

A PREPRINT

Siddhartha Srivastava
Department of Aerospace Engineering
University of Michigan Ann Arbor
sidsriva@umich.edu

Veera Sundararaghavan
Department of Aerospace Engineering
University of Michigan Ann Arbor
veeras@umich.edu

February 4, 2020

ABSTRACT

We present a Hybrid-Quantum-classical method for learning Boltzmann machines (BM) for generative and discriminative tasks. Boltzmann machines are undirected graphs that form the building block of many learning architectures such as Restricted Boltzmann machines (RBM’s) and Deep Boltzmann machines (DBM’s). They have a network of visible and hidden nodes where the former are used as the reading sites while the latter are used to manipulate the probability of the visible states. BM’s are versatile machines that can be used for both learning distributions as a generative task as well as for performing classification or function approximation as a discriminative task. We show that minimizing KL-divergence works best for training BM’s for applications of function approximation. In our approach, we use Quantum annealers for sampling Boltzmann states. These states are used to approximate gradients in stochastic gradient descent scheme. The approach is used to demonstrate logic circuits in the discriminative sense and a specialized two-phase distribution using generative BM.

1 Introduction

Boltzmann machines (BM) are graph models that capture dependencies between variables by associating a probability to each combination of variables based on an Ising–type energy expression. Quantum annealers are physical realizations of Boltzmann machines, in which each sample (containing multiple qubit states with binary read-out values) corresponds to
an energy with the probability increasing with decreasing energy. An example of nomenclature of Boltzmann machines is shown in Fig. 1. The model observes the data by using “visible units” (v) and constructs abstract representation of the data by using “hidden units” (h). The challenge in this approach is to learn weights (e.g., interconnection J) that are the best for hidden representation of data.

While all three models in Fig. 1 are Boltzmann machines, the particular nomenclature adopted is because of slowness of learning in general Boltzmann machines, which is NP-hard. Hinton (2002) \cite{2} proposed Restricted Boltzmann machine (RBM), which has no connections between hidden units and can be trained efficiently on a classical computer. Boltzmann machines have received a lot of attention as building blocks of multi-layer learning architectures for speech and image recognition \cite{3,4}. The idea is that features from one RBM can serve as input to another RBM. By stacking RBMs in this way, one can learn features from features in the hope of arriving at a high level representation. It is known that approximate inference in deep Boltzmann machines can handle uncertainty in better way and deal with ambiguous data \cite{1}. However, the learning process of deep Boltzmann machines on classical computers is quite slow due to the complexity involved in computing the likelihood of an undirected model or its gradient. Among classical methods, sampling techniques are typically employed but can be time-consuming due to the slow mixing of Gibbs sampling. This issue has been addressed in literature by methods that need fewer sampling steps, for instance, Contrapositive divergence (CD), Persistent Contrapositive divergence (PCD) and Fast Persistent Contrapositive divergence (FPCD) learning methods \cite{2,5}.

Quantum computing has the potential to significantly speed up training of Boltzmann machines. The stochastic gradient based training of the objective (such as log-likelihood functions) can be accelerated by sampling from quantum states. In noisy intermediate scale quantum computers (NISQ) the approach of amplitude amplification and estimation have been developed that results in a quadratic speedup with respect to sample acceptance probabilities \cite{6}. However, NISQ devices are limited small number of quantum bits, that restricts the problem to Boltzmann machines with small number of nodes. More recent work thus, has been along the lines of quantum/classical methods where classical training methods are enhanced with quantum state samples. The use of quantum annealers are promising for quantum/classical training since a large number of qubits are available and the training takes advantage of measurements on the physical
realization of the Boltzmann machine [7, 8]. Most importantly, the approach allows training of general Boltzmann machines as long as it can be embedded on the device. Our goal in this paper is to demonstrate the use of quantum annealers for discriminative and generative tasks involving Boltzmann machines.

Figure 2: An illustration of the hybrid Quantum-Classical computation technique: Quantum Annealer is used as a Boltzmann sampler while the gradient optimization is carried out using classical computation

2 Mathematical description

A Boltzmann Machine is a probabilistic graphical model defined on a complete graph of binary variables. We can partition the graph into “visible” nodes taking up values observed during training denoted by vector, \( v \), and “hidden” nodes where values must be inferred taking up values denoted by vector, \( h \). The probability of observing a state in the Boltzmann Machine is governed by its energy function:

\[
E(v, h) = - \sum_{i=1}^{m} \alpha^v_v v_i - \sum_{j=1}^{n} \alpha^h_h h_j - \sum_{i=1}^{m} \sum_{j=1}^{n} \gamma^{hv}_{ij} v_i h_j - \sum_{i=1}^{m} \sum_{j=i+1}^{m} \gamma^{vh}_{ij} v_i v_j - \sum_{i=1}^{n} \sum_{j=1}^{n} \gamma^{hh}_{ij} h_i h_j \tag{1}
\]

Where \( \alpha^v \) and \( \alpha^h \) denote self-interaction at nodes and \( \gamma^{hv} \), \( \gamma^{vh} \) and \( \gamma^{hh} \) denotes the node–node interaction terms. The superscript signifies the Together, these constitute the set of parameters \( \theta \) that need to be learnt for achieving a desired probability of visible states \( v \). The Boltzmann Machine is equivalent to a model from statistical physics known as the Ising model. The Ising model is a mathematical description of a physical phenomenon, which suggests a procedure for sampling states from a quantum annealer.

The distribution of equilibrated states can be modeled, at least approximately, as a Boltzmann distribution:

\[
p(v, h) = \frac{1}{Z} e^{-E(v, h)/k_B T} = \frac{1}{Z} e^{-\beta E(v, h)} \tag{2}
\]
Here, $Z$ denotes the partition function and is estimated as $Z = \sum_{v,h} e^{-\beta E(v,h)}$. The probability of a particular visible state $v$,

$$p(v) = \sum_h p(v,h) = \frac{1}{Z} \sum_h e^{-\beta E(v,h)}$$

(3)

### 2.1 Distribution matching using BMs

In the last section, we saw that each visible state occurs with a probability determined by the Boltzmann distribution. This implies that in a sufficiently long simulation of a BM each visible state is sampled with a known probability distribution which is refer to as the model probability distribution, $p$ (often written as $p_\theta$ to signify dependence on the parameters of energy). It is desirable that this distribution matches that of the data, denoted as the true probability, $q$. The process of learning distributions via BM essentially requires one to estimate the parameters $\theta$ such that $p_\theta$ and $q$ are close to each other. The Kullback-Leibler divergence $D_{KL}(q||p)$ is defined as,

$$D_{KL}(q||p) = -\sum_{v \in \{v^1, ..., v^D\}} q(v) \ln \frac{p(v)}{q(v)}$$

where the set $\{v^1, ..., v^D\}$ represents the $D$ data points. The KL divergence is always non-negative with $D_{KL}(q||p) = 0$ if and only if $q = p$ almost everywhere. For this property, this function is chosen to be the cost function. It is minimized using gradient based optimization techniques. The gradient is estimated as (See Appendix [A] for calculation):

$$\frac{\partial D_{KL}(q||p(\theta))}{\partial \theta} = -\beta \sum_{d=1}^D q(v^d) \left( \sum_h p(h|v^d) \frac{\partial E(v^d,h)}{\partial \theta} \right) + \beta \sum_{v',h'} p(v',h') \frac{\partial E(v',h')}{\partial \theta}$$

(4)

We first specialize this gradient for the Ising model (Eq [1]). We drop the $\beta$ constant as it can be subsumed in the learning rate:

$$\frac{\partial D_{KL}(q||p(\theta))}{\partial \alpha_i^v} = -\sum_{d=1}^D q(v^d) \nu_i^d + \sum_{v',h'} p(v',h')\nu_i'$$

(5a)

$$\frac{\partial D_{KL}(q||p(\theta))}{\partial \alpha_i^h} = -\sum_{d=1}^D q(v^d) \left( \sum_{h_i} p(h_i|v^d)h_i \right) + \sum_{v',h'} p(v',h')h_i'$$

(5b)

$$\frac{\partial D_{KL}(q||p(\theta))}{\partial \gamma_{ij}^{vh}} = -\sum_{d=1}^D q(v^d) \left( \sum_{h_j} p(h_j|v^d)\nu_j^d h_j \right) + \sum_{v',h'} p(v',h')\nu_j'$$

(5c)

$$\frac{\partial D_{KL}(q||p(\theta))}{\partial \gamma_{ij}^{vv}} = -\sum_{d=1}^D q(v^d) \nu_i^d \nu_j^d + \sum_{v',h'} p(v',h')\nu_i'\nu_j'$$

(5d)

$$\frac{\partial D_{KL}(q||p(\theta))}{\partial \gamma_{ij}^{hh}} = -\sum_{d=1}^D q(v^d) \left( \sum_h p(h|v^d)h_i h_j \right) + \sum_{v',h'} p(v',h')h_i'h_j'$$

(5e)
2.1.1 Approximating the gradient

The second terms in Eq(5a-5e) are summed over all possible states for a given parameter and therefore estimating this term exactly entails the tedious task of calculating $2^{m+n}$ probabilities. However, we can approximate the probabilities by sampling the states using quantum annealers. The assumption here is that the annealer samples the states with the same Boltzmann distribution. The states which are not sampled are assigned zero probability. It is evident through experimentation that with increasing number of samples the sample probability approached the model probability.

The first terms in Eq(5a,5d) are independent of $\theta$ and are only evaluated once. For the first terms in Eq (5b, 5c and 5e), required more work. If the visible data appears in the sampled states then the conditional probabilities can be directly calculated by considering only the samples with the specified visible data. In the event that there is no sample corresponding to some specific visible data, more samples can be generated. Since it is difficult to ensure sampling of a specific visible data, we just simulate the hidden subgraph of the Boltzmann machine i.e. we ignore the visible nodes and the connections between the visible node. The self-interaction term is augmented to be $b_j \rightarrow \sum_{i=1}^{m} (b_j + c_{ij}v_i^d)$. It is observed from Eq[6] that this procedure ensures that the energy gaps between the states of hidden subgraph remains unchanged. Since the Boltzmann distribution is unchanged by translation of energy, the samples of hidden subgraph are directly used to estimate the conditional probability in the second term of Eq (5b, 5c and 5e)

$$E(h|v^d) = -\sum_{i=1}^{m} a_i v_i^d - \sum_{i=1}^{m} \sum_{j=i+1}^{m} d_{ij} v_i^d v_j^d - \sum_{i=1}^{m} \sum_{j=1}^{n} (b_j + c_{ij} v_i^d) h_j - \sum_{i=1}^{n} \sum_{j=i+1}^{n} e_{ij} h_i h_j$$

2.1.2 Learning Algorithm

The following momentum-based update rule is used in this work:

$$\theta^{(t+1)} = \theta^{(t)} + \eta \frac{\partial}{\partial \theta^{(t)}} D_{KL}(q||p) - \lambda \theta^{(t)} + \nu \Delta \theta^{(t-1)}$$

An empirical choice of learning parameters is given in table 1.

| Range        | Empirically determined | $10^{-5} - 10^{-2}$ | $0 - 0.9$ |
|--------------|-----------------------|----------------------|-----------|
| $\eta$       |                       |                      |           |
| $\lambda$    |                       |                      |           |
| $\nu$        |                       |                      |           |

Table 1: Learning parameters range

For this problem to be well posed it is also required that the parameters are bounded. In all our example we will take $|H| \leq H_{max}$ and $|J| \leq J_{max}$ with $H_{max} = J_{max} = 1$ unless indicated otherwise. This choice of bound is motivated by the simplicity of the form and the acceptable range of D’Wave processor. The learning procedure is summarized in Algorithm 1.
**Algorithm 1 Learning algorithm (Distribution Matching)**

1: Find embedding of the BM graph and the hidden subgraph on Quantum annealer
2: Initialize $\theta$
3: $\Delta \theta = M$ (Large number)
4: Count=1
5: while $|\Delta \theta| > \Delta \theta_{min}$ AND Count < Maximum count do
6: Embed graph with parameter $\theta$ in quantum annealer and sample states
7: for $v_i \in \{v^1, \ldots, v^D\}$ do
8: if $v_i \notin$ Sampled data then
9: Embed hidden subgraph with self and pairwise interaction parameters of energy form Eq (6)
10: Anneal and sample states
11: Evaluate gradient using Eqns (5a, 5e)
12: Calculate $\theta_{new}$ using Eq(7)
13: $\Delta \theta = \theta_{new} - \theta$
14: $\theta = \theta_{new}$
15: Count=Count+1
16: end

2.2 Numerical Example

In this example, We consider the data presented in Fig3a. In this toy example, we consider 0 and 1 as two phases on a line discretized with 10 points. Each of these points is treated as a visible node which is assigned a phase (0/1). We want to consider only the phases with at-most one boundary between the left phase (assigned value 0) and the right phase (assigned value 1). Thus there are 11 possibilities and all are considered equally likely. We learned a fully-connected Boltzmann machine with 8 hidden nodes (Parameters presented in Appendix). The BM assigns high and similar probabilities to each phase as desired. The variation of probability of each state with respect to the $\beta$ is presented in Fig 3b. It is observed that the result does not have the expected distribution for very high or very value of $\beta$. This is due to the finite annealing temperature of D’Wave processor that decides the training $\beta$ of the BM. It can be observed in Fig 3c that in this case the range of appropriate $\beta$ is $[1.5, 3]$. In the next section, we will discuss the consequence of finite annealing temperature on the learning process.

2.3 Effect of Annealing temperature

To further understand the difference between a BM developed for $\beta \to \infty$ and for a finite $\beta$ we look at the example presented in Fig4. In this example two XOR gates are presented i.e the states $\{[0, 0, 0], [0, 1, 1], [1, 0, 1], [1, 1, 0]\}$ are chosen as visible data with each having equal probability. Here, the first two states are regarded as the input node (denoted by $v^I$) and the third one as the output node (denoted by $v^O$). Additionally, there is one more hidden node in
Figure 3: (a) Data points representing phases with at most one boundary. (b) The model probability of each data point with yellow region defining the spread between the minimum and the maximum probabilities (c) KL Divergence of the model at different $\beta$

the network. The ground states (Energy minimising states of the Ising model) of the left BM (Fig 4a) are designed to represent the visible data. We will refer to this BM as ground state BM. The right BM (Fig 4b) is learned using the method outlined in the previous section. It is evident from the probability distribution of states for ground state BM (Fig 4c) that all states are equally probable and this probability increases with the increase in $\beta$. On the other hand the probability distribution of some states for trained BM start to decrease for high BM. Infact, it can be shown that (see Appendix B for the proof of following statements):

Statement 1: As $\beta$ increases the ground state probability increases. It strictly increases in the case when not all states are ground states.

Statement 2: For every model parameter $\theta$, assuming the absolute value of energy of every state is bounded with $E_{max}$ and not all states are ground states, there exists a $\beta_c$ such that for all $\beta \geq \beta_c$ the probability of each excited state decreases with $\beta$. Here, we refer to excited state as any admissible state which is not a ground state.

As a result, the probability of each of the visible data increases in this case and at $\beta \rightarrow \infty$, it exactly matches the data. And it can be seen in Fig 4e that the KL Divergence decreases to zero in this limit. It should be noted that this behavior is not true in general. In fact all ground states are equally probable in this limit, therefore arbitrary probability distribution of data cannot be exactly matched with a grounds state BM. On the other hand, in the trained BM, the KL Divergence (Fig 4f) is minimum for a finite value of $\beta$ and it diverges as $\beta \rightarrow \infty$. This BM performs best near this minima as evident from the probability distribution. In fact, it can be shown that if $D_{KL} \rightarrow 0$ for a finite value of $\beta$ i.e.
when $p \sim q$, we have

$$\frac{d^2 D_{KL}}{d\beta^2} \rightarrow \sum_{v \in \{v^1, \ldots, v^D\}} p(v)(-\operatorname{Var}(E|v) + \operatorname{Var}(E)) \quad \text{(see Appendix A)}$$

$$= \operatorname{Var}(\mathbb{E}_v(E|v)) \quad \text{(Law of total variance)}$$

This implies that $\frac{d^2 D_{KL}}{d\beta^2} \geq 0$ in the above case with equality only in the extremely rare case of same expected value of Energy for each visible data. Therefore, a well-trained BM (i.e. $q \sim p$ for the training $\beta$) often shows a local minima at the training $\beta$.

A quantum annealer samples Boltzmann states with an unknown $\beta$. In previous studies, it has been seen that the value of $\beta$ is high but also dependent on the size and scale of the problem (e.g. number of qubits used). The next feature that we will investigate in this study is the consequence of training the Boltzmann machine at an unknown $\beta$. We first look at the $D_{KL}$ of the trained BM for the data corresponding to the AND-gate (Fig5a). Subsequent to training the Boltzmann machine, solutions are sampled and the $D_{KL}$ for each sample is measured. It is observed that the distribution of $D_{KL}$ of sample lies in the neighborhood of the minimum $D_{KL}$ of the model with respect to $\beta$. This supports the intuitive argument that a trained BM performs best in the neighborhood of the sampling or training $\beta$. However, other than these experimental results, we are not aware of any theoretical guarantees for such a phenomenon.

Next, we look at a similar example using OR-gate data. Three BM’s with 2, 5 and 10 hidden nodes are considered. It is observed that the $D_{KL}$ (Fig5b) is minimum at different values of $\beta$ which suggests that the training of these models is carried out at different $\beta$.

Motivated by this finding, we simulated complete graphs of different sizes with random parameters. We estimated the $\beta$ values which corresponded to the closest match with analytical distribution. Specifically, we estimated

$$\beta^* = \arg \min_{\beta} \frac{1}{2} \sqrt{\sum_i (\sqrt{p_i^s} - \sqrt{p_i^a(\beta)})^2}$$

where $p^s$ is the sample probability, $p^a(\beta)$ is the analytical probability (dependent on $\beta$) and the cost function corresponds the the Hellinger distance between two distributions. The training $\beta$ calculated from this analysis is presented in Fig 6. It shows that the local solver (C4) trains the BM at an almost constant value of $\beta = 3$ while the training $\beta$ of remote solver decreases with Graph size. It should be noted that such an analysis is not possible for very large graph as analytical calculation of probabilities take $2^N$ computations where $N$ is the number of nodes.

### 2.4 Initial parameter dependence on convergence

The energy surface of Ising model is non-linear and hence gradient-based methods are prone to getting stuck in local minima. This is evident in the Fig7 where the $D_{KL}$ at each learning step is presented for different initial conditions. It can observed that some solution go as low as $10^{-5}$ while others do not. Following learning parameters are used in the simulation: Learning rate $= 10^{-1}$, Learning Steps ($\eta$) $= 10^3$, Weight decay rate ($\lambda$) $= 10^{-5}$, Momentum rate ($\nu$) $= 0.6$. 

A preprint - February 4, 2020
Figure 4: (a) A ground state BM for XOR-gate data, (b) A trained BM for XOR-gate data (c) Model probability distribution variation with $\beta$ for the ground state BM, (d) Model probability distribution variation with $\beta$ for the trained BM, (e) KL-Divergence variation with $\beta$ for the ground state BM and (f) KL-Divergence variation with $\beta$ for the trained BM
3 Function Approximation

So far, we have talked about using BM’s for imitating the distribution of a given data. It can also be used as a function. In simple terms, we want to encode the tuple \((x, f(x))\) as the state of the visible nodes. Here, \(f(x)\) denotes the function that is to be approximated. In the context of the BM, the visible data can be thought of as the binary expansion of \(x\) and \(f(x)\). It is desired that if the visible nodes corresponding to the input variable take a particular value \(x'\), then the visible nodes corresponding to the output variable take a value of \(f(x')\) with high probability. Thus a BM for approximating function \(f(x)\) has high conditional probability, \(p(f(x)|x)\). This conditional probability can be estimated using Eq(8) where the visible data \(v\) is written as concatenation of input data, \(v^I\) and output data, \(v^O\). An example of a BM for approximating AND gate is presented in Fig[8]. It is observed that \(p(v^O|v^I)\) → 1 as \(\beta \rightarrow \infty\) (Low temp solution) and
Figure 7: $D_{KL}(q||p)$ at each learning step for 5 randomly initiated parameters for training data set corresponding to AND-gate

$p(v^O|v^I) 	o 0.5$ as $\beta \to 0$ (High temp solution).

\[ p(v^O|v^I) = \frac{p([v^I, v^O])}{\sum_v p([v^I, v^O])} \]  \hspace{1cm} (8)

Figure 8: (a) Boltzmann machine to approximate AND Gate with 3 visible nodes and 1 hidden node. The numerical values in blue represent the self-interaction parameters and in red represent the node-node interaction parameters. (b) The conditional probability of the BM evaluated at different values of $\beta$. 
The energy defined in Eq (1) can be slightly modified to distinguish the interaction terms of the input and output visible nodes as follows:

\[
E(v^I, v^O, h) = - \sum_{i=1}^{m_I} \alpha_i v^I_i - \sum_{i=1}^{m_O} \alpha_i v^O_i - \sum_{j=1}^{n} \alpha_j h_j - \sum_{i=1}^{m_I} \sum_{j=1}^{n} \gamma_i^I h_j v^I_i - \sum_{i=1}^{m_O} \sum_{j=1}^{n} \gamma_i^O h_j v^O_i - \sum_{i=1}^{m_I} \sum_{j=1}^{n} \gamma_{ij} v^I_i v^O_i h_j - \sum_{i=1}^{m_O} \sum_{j=1}^{n} \gamma_{ij} v^O_i v^O_i h_j
\]

Here the superscripts $v^I$ and $v^O$ are introduced to signify that the respective parameter affects input and the output node, respectively.

### 3.1 Learning Method for function approximators

The cost function in this case will be taken as the Negative Conditional Log-likelihood $L$ defined as,

\[
L(\theta) = - \sum_{[v^I, v^O] \in \{v^1, ..., v^D\}} \ln p(v^O|v^I, \theta)
\]

where, as before, the set $\{v^1, ..., v^D\}$ represents the $D$ data points. The gradient of the cost function is estimated as:

\[
\frac{\partial L(\theta)}{\partial \theta} = \beta \sum_{[\tilde{v}^I, \tilde{v}^O] \in \{v^1, ..., v^D\}} \left( \sum_{v^O, h} p(v^O, h|\tilde{v}^I) \frac{\partial E(\tilde{v}^I, v^O, h)}{\partial \theta} - \sum_{h'} p(h'|\tilde{v}^I, \tilde{v}^O) \frac{\partial E(\tilde{v}^I, \tilde{v}^O, h')}{\partial \theta} \right)
\]

As before we use the momentum-based gradient scheme defined in Eq(7) to learn the BM. In this case the sampling will be done on (1) the subgraph of hidden nodes (2) the subgraph of hidden and output nodes. The parameters of these graphs can be estimated in the same way as in the case of distribution matching.
Algorithm 2 Learning algorithm (Function approximator)

1: Find embedding of the BM hidden subgraph and the hidden+output subgraph on Quantum annealer
2: Initialize $\theta$
3: $\Delta \theta = M$ (Large number)
4: Count=1
5: while $|\Delta \theta| > \Delta \theta_{\text{min}}$ AND Count < Maximum count do
6: Embed graph with parameter $\theta$ in quantum annealer and sample states
7: for $v^d \in \{v^1,...,v^D\}$ do
8: Embed hidden subgraph with modified parameters
9: Anneal, sample states and approximate $p(h'|\tilde{v}^l, \tilde{v}^O)$.
10: Embed hidden+output subgraph with modified parameters
11: Anneal, sample states and approximate $p(v^O,h|\tilde{v}^l)$.
12: Evaluate gradient using Eq (10)
13: Calculate $\theta_{\text{new}}$ using Eq (7)
14: $\Delta \theta = \theta_{\text{new}} - \theta$
15: $\theta = \theta_{\text{new}}$
16: Count=Count+1
17: end

3.2 Example of an Adder Gate

As an example we present a BM for 2-bit adder shown in Table 2. A comparison of BM learned as a Function approximator and a Distribution is presented in Fig. 9 with values of the trained parameter presented in Appendix D. It is observed that the KL Divergence of distribution BM (Fig 9d) has a minimum at around $\beta \sim 5$ which asserts that the training $\beta$ for this BM is close to 5. As expected, the function approximator does not show any minimizing effect in the KL divergence. However, the conditional probabilities in the case of Function approximator (Fig 9a) are higher in comparison of Distribution BM (Fig 9b). At $\beta 5$, the minimum conditional probability in the case of Function approximator is close to 0.3 while that of distribution BM is 0.15 This is the desired outcome from this procedure.

4 Conclusion

We have developed and analysed two training methodologies for training Boltzmann machine. A summary of these method is as follows:

1. Generative training for distribution matching is carried out by minimizing KL divergence. We present the example of learning a distribution of 2-phase 1D material with at-most one boundary.
2. Discriminative training for function learning is carried out by minimizing conditional log-likelihood. We present the example of a 2-bit adder circuit.

Gradient based methods are used to minimize these costs. Quantum annealer is employed for direct sampling for the approximation of the gradients. In both methods the number of sampling steps depend linearly on the number of data points. We also analysed the effect of training BM at a finite annealing temperature. The resulting trained BM is not a Ground state BM. Moreover, the annealing temperature is shown to depend on the Graph-size (with possibly other factors). As a next step, we will analyse the result of combined Generative-Discriminative training of classifiers.

References

[1] Ruslan Salakhutdinov and Geoffrey Hinton. Deep boltzmann machines. In *Artificial intelligence and statistics*, pages 448–455, 2009.

[2] Geoffrey E Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002.

[3] Navdeep Jaitly and Geoffrey Hinton. Learning a better representation of speech soundwaves using restricted boltzmann machines. In *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5884–5887. IEEE, 2011.

[4] SM Ali Eslami, Nicolas Heess, Christopher KI Williams, and John Winn. The shape boltzmann machine: a strong model of object shape. *International Journal of Computer Vision*, 107(2):155–176, 2014.

[5] Asja Fischer and Christian Igel. An introduction to restricted boltzmann machines. In *Iberoamerican Congress on Pattern Recognition*, pages 14–36. Springer, 2012.
Figure 9: Boltzmann machine representing a 2-bit adder circuit: (a) Conditional Probability of a BM trained as a function approximator (b) Conditional Probability of a BM trained as a distribution (c) KL Divergence of a BM trained as a function approximator (d) KL Divergence of a BM trained as a distribution

[6] Nathan Wiebe, Ashish Kapoor, and Krysta M Svore. Quantum deep learning. arXiv preprint arXiv:1412.3489, 2014.

[7] Amir Khoshaman, Walter Vinci, Brandon Denis, Evgeny Andriyash, and Mohammad H Amin. Quantum variational autoencoder. Quantum Science and Technology, 4(1):014001, 2018.

[8] Walter Vinci, Lorenzo Buffoni, Hossein Sadeghi, Amir Khoshaman, Evgeny Andriyash, and Mohammad H Amin. A path towards quantum advantage in training deep generative models with quantum annealers. arXiv preprint arXiv:1912.02119, 2019.
A Estimation of gradients

A.1 Gradient of Log-likelihood for a single data

\[ \frac{\partial \ln p_{\theta}(v)}{\partial \theta} = \frac{\partial}{\partial \theta} \left( \ln \sum_{v} e^{-E(v,h)} \right) - \frac{\partial}{\partial \theta} \left( \ln \sum_{v'} e^{-E(v',h')} \right) \]

\[ = -\beta \sum_{h} e^{-E(v,h)} \frac{\partial E(v,h)}{\partial \theta} + \beta \sum_{v',h'} e^{-E(v',h')} \frac{\partial E(v',h')}{\partial \theta} \]

\[ = -\beta \sum_{h} \frac{1}{Z} \sum_{h'} e^{-E(v,h)} \frac{\partial E(v,h)}{\partial \theta} + \beta \sum_{v',h'} \frac{e^{-E(v',h')}}{Z} \frac{\partial E(v',h')}{\partial \theta} \]

\[ = -\beta \sum_{h} p(v,h) \frac{\partial E(v,h)}{\partial \theta} + \beta \sum_{v'} p(v',h') \frac{\partial E(v',h')}{\partial \theta} \]

A.2 Derivative of \( D_{KL} \) w.r.t. \( \beta \)

\[ \frac{dD_{KL}}{d\beta} = - \sum_{v \in \{v^1,...,v^D\}} q(v) \frac{d}{d\beta} \ln p(v) = - \sum_{v \in \{v^1,...,v^D\}} q(v) \frac{d}{d\beta} \sum_{h} e^{-E(v,h)} \frac{\partial E(v,h)}{\partial \theta} \]

\[ = - \sum_{v \in \{v^1,...,v^D\}} q(v) p(v) \left( - \sum_{h} \frac{E(v,h)}{\sum_{h'} e^{-E(v,h')}} \right) + \left( \sum_{h} e^{-E(v,h)} \right) \left( \sum_{v',h'} \frac{E(v',h')}{\sum_{h''} e^{-E(v',h'')}} \right) \]

\[ = \sum_{v \in \{v^1,...,v^D\}} q(v) \left( \sum_{h} E(v,h)p(h|v) - \sum_{v',h'} E(v',h')p(v',h') \right) \]

\[ = -E_{v,h}(E) + \sum_{v \in \{v^1,...,v^D\}} q(v) \sum_{h} E(v,h)p(h|v) \]

\[ \frac{d^2D_{KL}}{d\beta^2} = \sum_{v \in \{v^1,...,v^D\}} q(v) \left( \sum_{h} \frac{E(v,h)}{\sum_{h'} e^{-E(v,h')}} + \frac{\partial}{\partial \beta} \frac{\partial E(v,h)}{\partial \theta} \right) \]

\[ = \sum_{v \in \{v^1,...,v^D\}} q(v) \left( \sum_{h} \frac{e^{-E(v,h)}}{\sum_{h'} e^{-E(v,h')}} \right) - \sum_{v',h'} E(v',h') \frac{d}{d\beta} \frac{e^{-E(v',h')}}{\sum_{h''} e^{-E(v',h'')}} \]

\[ = \text{Term I} - \text{Term II} \]
Here, we evaluate term \( I \) as:

\[
\frac{d}{d\beta} \sum_{h''} e^{-\beta E(v,h)} = -E(v, h) e^{-\beta E(v,h)} + \frac{e^{-\beta E(v,h)} \sum_{h'} E(v, h') e^{-\beta E(v,h')}}{(\sum_{h''} e^{-\beta E(v,h'')})^2}
\]

\[
= -E(v, h)p(h|v) + p(h|v) \sum_{h'} E(v, h') p(h'|v)
\]

And term \( II \) as:

\[
\frac{d}{d\beta} \sum_{w', h''} e^{-\beta E(w',h'')} = -E(v', h') e^{-\beta E(v',h')} + \frac{e^{-\beta E(v',h')} \sum_{w'', h''} E(v'', h'') e^{-\beta E(v'',h'')}}{Z^2}
\]

\[
= -E(v', h') p(v', h') + p(v', h') \sum_{w'', h''} E(v'', h'') p(v'', h'')
\]

Combining the two terms:

\[
\frac{d^2 D_{KL}}{d\beta^2} = \sum_{v \in \{v^1, \ldots, v^D\}} q(v) \left( \sum_h -E^2(v, h)p(h|v) + \left( \sum_{h'} E(v, h') p(h'|v) \right)^2 \right) + \sum_{v', h'} E^2(v', h') p(v', h') - \left( \sum_{w'', h''} E(v'', h'') p(v'', h'') \right)^2 
\]

\[
= \sum_{v \in \{v^1, \ldots, v^D\}} q(v) \left( -\text{Var}(E|v) + \text{Var}(E) \right)
\]

\[\text{B Proof of Propositions}\]

\text{Proposition 1:}\ As \( \beta \) increases the ground state probability increases. It strictly increases in the case when not all states are ground states.

\text{Proof:}\ Let \( S_g \) be a ground state and \( E_{\text{min}} = E(S_G) \). It suffices to show that:

\[
\frac{d \ln p(S_g)}{d\beta} \geq 0
\]

We estimate the derivative as follows:

\[
\frac{d \ln p(S_g)}{d\beta} = \frac{d \ln e^{-\beta E_{\text{min}}}}{d\beta} = \frac{d (-\beta E_{\text{min}} - \ln Z)}{d\beta}
\]

\[
= -E_{\text{min}} + \sum_S E(S) \frac{e^{-\beta E(S)}}{Z}
\]

\[
= -E_{\text{min}} + \sum_S E(S) p(S)
\]

\[
= -E_{\text{min}} + \mathbb{E}(E) \geq 0
\]
Here $\mathbb{E}(E)$ denotes the expected energy of the system. It can be easily seen that the inequality is strict if and only if there not all states have the same energy.

**Proposition 2:** For every model parameter $\theta$, assuming the absolute value of energy of every state is bounded with $E_{\max}$, then given any two states $S_1$ and $S_2$ with $E(S_1) < E(S_2)$ there exists $\beta_{S_1} > \beta_{S_2}$ such that for all $\beta > \beta_{S_i}$, the probability of state $S_i$, denoted by $p(S_i)$ decreases.

**Proof:** Let $S$ be any state. Following the calculation done in proof of proposition 1, we get:

$$\frac{d \ln p(S)}{d\beta} = -E(S) + \mathbb{E}(E)$$

Let $S_g$ be a ground state with $E_{\min} = E(S_G)$ and $S_{e_1}$ be a first excited state (State with energy strictly greater than the ground state but less than every excited state) with $E_1 = E(S_{e_1})$. This means for any excited state, $S_{exc}$, we have:

$$\frac{d \ln p(S_{exc})}{d\beta} = -E(S_{exc}) + \mathbb{E}(E) \leq -E(S_1) + \mathbb{E}(E) = \frac{d \ln p(S_1)}{d\beta}$$

We will first show that $\frac{d \ln p(S_1)}{d\beta} \leq 0$ for all $\beta \geq \beta_c$ for some $\beta_c > 0$.

$$\frac{d \ln p(S_{e_1})}{d\beta} = -E_1 + \mathbb{E}(E) = -E_1 + \sum_{S \in S} E(S)p(S)$$

$$= -E_1 + \sum_{S_g \in \mathcal{G}} E(S_g)p(S_g) + \sum_{S \in S - \mathcal{G}} E(S)p(S)$$

$$\leq -E_1 + E_{\min}\sum_{S_g \in \mathcal{G}} p(S_g) + E_{\max}\sum_{S_g \in \mathcal{G}} p(S_g)$$

Denoting with $P \equiv P(\theta, \beta)$ the total probability of ground states, estimated as $P = \sum_{S_g \in \mathcal{G}} p(S_g) = |\mathcal{G}|e^{-\beta E_{\min}}Z$, where $|\mathcal{G}|$ denotes the multiplicity of ground states, we get that

$$\frac{d \ln p(S_1)}{d\beta} \leq -E_1 + E_{\min}P + E_{\max}(1 - P)$$

$$= (E_{\max} - E_1) - P(E_{\max} - E_{\min})$$

We want to choose $\beta_c$ such that $P(\theta, \beta_c) \geq \frac{E_{\max} - E_1}{E_{\max} - E_{\min}}$. Noting that not all states have energy $E_{\min}$, we get, $(E_{\max} - E_1) < P(E_{\max} - E_{\min})$. From Proposition 1, we know that $P$ is strictly increasing with $\sup_{\beta} P = 1$. We get that there is some $\beta_c$ such that $P(\theta, \beta_c)$ has the desired value. Next, to prove the claim we note that:

$$\frac{d \ln p(S_1)}{d\beta} = -E(S_1) + \mathbb{E}(E) > -E(S_2) + \mathbb{E}(E) = \frac{d \ln p(S_2)}{d\beta}$$

This means $p(S_2)$ is decreasing whenever $p(S_1)$ is decreasing. 

$\blacksquare$
Corollary 3: For every model parameter $\theta$, assuming the absolute value of energy of every state is bounded with $E_{max}$ and not all states are ground states, there exists a $\beta_c$ such that for all $\beta \geq \beta_c$ the probability of each excited state decreases with $\beta$. ■

C Change of basis

It is a common practice to define the Ising states as either $\{0, 1\}$ or $\{-1, +1\}$ states. We choose the former format while D-Wave works with the latter one. In this section we will present the details about conversion between these formats. For the purpose of discussion, we will denote the $\{0, 1\}$ Ising model with lower case variables, $\{s, \theta^I, \theta^{II}\}$ and the $\{-1, 1\}$ Ising with upper case variables, $\{S, \Theta^I, \Theta^{II}\}$. Here, $s/S$ denotes the state of a graph, $\theta^I/\Theta^I$ denotes the self-interaction parameter and $\theta^{II}/\Theta^{II}$ denote the Pairwise interaction parameter. The states of the system can be interchanged using the following equation:

$$S = 2s - 1$$ (12)

In this transformation of variables, the equivalent transformation in energy parameters is given by:

$$\Theta^I_{kl} = \frac{1}{4} \theta^I_{kl}$$

$$\Theta^I_k = \frac{1}{2} \theta^I_k + \frac{1}{4} \sum_{l \in \mathcal{N}(k)} \theta^{II}_{kl}$$ (13)

Here we denote the neighbouring vertices of node $k$ as $\mathcal{N}(k)$. This transformation shifts the energy of each state with a constant value and hence leaves the Boltzmann probability unchanged as required.

D Parameters for Boltzmann machine
### Table 3: Parameters for Boltzmann machine ($10^{-4}$ units) representing single jump from 0 to 1 trained at $\beta \sim 2$. The diagonal terms represent the field strength. Due to symmetric relation of interaction the lower diagonal is taken to be zero.

|   | $v_1$  | $v_2$  | $v_3$  | $v_4$  | $v_5$  | $v_6$  | $v_7$  | $v_8$  | $v_9$  | $v_{10}$ | $h_1$  | $h_2$  | $h_3$  | $h_4$  | $h_5$  | $h_6$  | $h_7$  | $h_8$  |
|---|--------|--------|--------|--------|--------|--------|--------|--------|--------|----------|--------|--------|--------|--------|--------|--------|--------|--------|
| $v_1$ | 7009   | 9446   | -4488  | -0392  | -0918  | 1762   | 1967   | 2256   | 2911   | 1935     | -3519  | 2152   | 5159   | -0329  | 1238   | -0146  | 2742   | 4321   |
| $v_2$ | 0      | 6927   | -9555  | -3119  | -1885  | 0984   | 1944   | 2216   | 2024   | 0946     | -4469  | 1052   | 7119   | 2408   | 1518   | 0369   | 3999   | 2524   |
| $v_3$ | 0      | 0      | 6546   | -7917  | -2526  | -0039  | -1300  | 0823   | 0540   | 1529     | -4151  | 3168   | 5857   | -0242  | 2070   | 2206   | 5279   | 3935   |
| $v_4$ | 0      | 0      | 0      | 2176   | -6351  | -2532  | -1320  | -0058  | 0887   | -0831    | 2895   | 3605   | 7262   | 0516   | 1932   | 0069   | 6981   | 5151   |
| $v_5$ | 0      | 0      | 0      | 0      | 4953   | -8388  | -4282  | -2942  | 0295   | -1597    | -1135  | 3144   | 4633   | 1975   | 0508   | 0668   | 4561   | 7515   |
| $v_6$ | 0      | 0      | 0      | 0      | 0      | 3723   | -9747  | -4126  | -0873  | -1271    | 0162   | 5440   | 3709   | 1110   | -2177  | -0613  | 3087   | 6950   |
| $v_7$ | 0      | 0      | 0      | 0      | 0      | 7447   | -9664  | -3826  | -1666  | 1104     | 5864   | 1671   | 1570   | -4787  | -0780  | 2767   | 3082   |        |
| $v_8$ | 0      | 0      | 0      | 0      | 0      | 0      | 8708   | -9827  | -6112  | -0021    | 5564   | 1466   | 1168   | -5490  | 0846   | 0739   | 2666   |        |
| $v_9$ | 0      | 0      | 0      | 0      | 0      | 0      | 0      | 0      | 0      | 0         | 0      | 0      | 0      | -5490  | 0846   | 0739   | 2666   |        |
| $v_{10}$ | 0     | 0      | 0      | 0      | 0      | 0      | 0      | 0      | 0      | 0         | 0      | 0      | 0      | 0      | 0      | 0      | 0      |        |

Due to symmetric relation of interaction the lower diagonal is taken to be zero.
Table 4: Parameters for Boltzmann machine ($10^{-4}$ units) representing Adder circuit trained at $\beta \sim 5$ as a function approximator. The diagonal terms represent the field strength. The bounds for of training were taken as $H_{max} = 2$ and $J_{max} = 1$. Due to symmetric relation of interaction the lower diagonal is taken to be zero.
Table 5: Parameters for Boltzmann machine (10^-4 units) representing Adder circuit trained at $\beta \sim 5$ as a distribution. The diagonal terms represent the field strength. The bounds for of training were taken as $H_{max} = 2$ and $J_{max} = 1$. Due to symmetric relation of interaction the lower diagonal is taken to be zero.