

Framework for evaluating quantum-assisted training of Restricted Boltzmann Machines

Pedro Ripper, Joaquim Dias Garcia

PSR

Rio de Janeiro 22250-040, Brazil

{pedroripper, joaquim}@psr-inc.com

Gustavo Castro do Amaral

The Netherlands Organization for Applied Scientific Research, TNO

Stieltjesweg 1, Delft 2628 CK, The Netherlands

gustavo.gustavoamaral@gmail.com

Guilherme Penello Temporão

Department of Electrical Engineering, Pontifical Catholic University of Rio de Janeiro

Rio de Janeiro 22453-900, Brazil

temporao@puc-rio.br

ABSTRACT

Restricted Boltzmann Machines (RBM) are neural networks that can be used to learn underlying data distributions. RBMs fall within the class of Energy Based Models, where the dependence between their variables can be represented by an energy function $E(\mathbf{v}, \mathbf{h})$. In order to train this model, different methods have been proposed, with the common goal of approximating the variable distributions, such as Contrastive Divergence and Persistent Contrastive Divergence. In this work, we are investigating different ways that Quantum Computing (QC) or Quantum-Inspired technologies can aid the training process. To accomplish that, a framework that implements different quantum-assisted training techniques is being developed, leveraging an interface with contrasting QC platforms.

KEYWORDS. Quantum Computing, Quantum Machine Learning, QUBO, Julia

SS-QPO - Quantum Methods and their Applications in Operational Research

1. Introduction

Optimization problems in the QUBO (Quadratic Unconstrained Binary Optimization) format are gaining great importance over the last few years, as a consequence of the development of Quantum Computers and other Quantum-Inspired technologies that are well suited to sample solutions for this type of problem. That being said, different techniques such as the Variational Quantum Eigensolver (VQE), Quantum Approximate Optimization Algorithm (QAOA) and Quantum Annealing are being applied in different contexts where the task at hand can be mapped to a QUBO.

These use cases include Finance [Orus et al., 2019], Engineering [Bernal et al., 2022] and Machine Learning (ML) [Ajagekar and You, 2020]. Regarding solutions for ML, quantum computers can be used for two different roles. First, one can use Quantum Machine Learning (QML) methods which, for nowadays, still require the support of classical machines [Cerezo et al., 2022], for the size of the problem being tackled takes a heavy toll on the quantum computer execution.

A different path that one might take is the use of quantum computers to help the training process of classical ML models [Ajagekar and You, 2020]. A relevant example for this case

is the use of quantum devices to sample states from energy-based models, such as Boltzmann Machines [Hinton, 2002], and Hopfield Networks [Hopfield, 1982], where the energy function of these neural networks can be easily mapped to a QUBO.

This work will focus on analysing the benefits of using quantum computers to help the training process of a Restricted Boltzmann Machine (RBM), via a Julia package that executes different quantum-assisted RBM-learning techniques. Different aspects will be observed to evaluate whether this method can actually improve the learning of an RBM, such as the training time and how well the trained model performs, compared to one that was trained solely by a classical computer.

2. Restricted Boltzmann Machine (RBM)

Restricted Boltzmann Machines are neural networks that can be used to learn underlying data distributions. RBMs can be represented as bipartite graphs, as portrayed in Figure 1, with n visible neurons and m hidden ones. Each visible node v_i is connected to all hidden nodes h_j , having an edge with weight w_{ij} . Additionally, each visible node has a bias parameter a_i , while each hidden neuron has a bias parameter b_j .

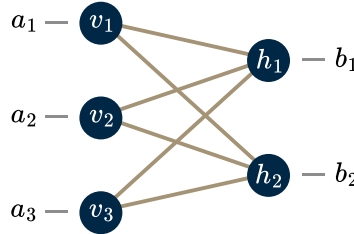


Figure 1: RBM graph representation

As mentioned, RBMs are Energy-Based Models [Dawid and LeCun, 2023], where the probability of a given $\mathbf{v}, \mathbf{h}$ configuration, displayed in Equation 1, depends on an energy function $E(\mathbf{v}, \mathbf{h})$, portrayed in Equation 2, and a normalizing factor Z , also known as the partition function.

$$p(\mathbf{v}, \mathbf{h}) = \frac{1}{Z} \exp(-E(\mathbf{v}, \mathbf{h})) \quad (1)$$

$$E(\mathbf{v}, \mathbf{h}) = -\sum_{i=1}^n a_i v_i - \sum_{j=1}^m b_j h_j - \sum_{i,j}^{m,n} w_{ij} v_i h_j \quad (2)$$

The partition function Z is estimated by summing over all possible configurations of visible and hidden nodes, which can be regarded as a computation with exponential cost, as the number of variables increases. Its formula is presented below.

$$Z = \sum_{\mathbf{v}, \mathbf{h}} \exp(-E(\mathbf{v}, \mathbf{h})) \quad (3)$$

Observing the energy function in Equation 2, one can notice that, in fact, its formula corresponds to the objective function of a QUBO problem. Considering that the values for w_{ij} can be mapped to the quadratic terms, using $w_{ij} = -2q_{ij}$, and a_i, b_j corresponding to the linear terms ($-q_{ii}$), it is possible to structure a QUBO model as follows in Equation 4, where $\mathbf{x} \in \mathbb{R}^{n+m}$ represents the concatenation of the $\mathbf{v}, \mathbf{h}$ vectors.

$$(\text{QUBO}) : \min_{\mathbf{x} \in \mathbb{B}^{m+n}} 2 \sum_{i=1}^{m+n} \sum_{j=i+1}^{m+n} q_{i,j} x_i x_j + \sum_{i=1}^{m+n} q_{i,i} x_i \quad (4)$$

2.1. RBM Training

The goal of training an RBM is to maximize the log likelihood that the model assigns to a training data $\mathbf{v}$, i.e. $p(\mathbf{v})$ Tieleman [2008]. This can be performed via gradient descent, where the gradient of the log probability, with respect to a hyperparameter, is described in Equation 5, considering the weight value w_{ij}

$$\frac{\partial \log p(\mathbf{v}_{data})}{\partial w_{ij}} = \frac{\partial E(\mathbf{v}_{data}, \mathbf{h}_{data})}{\partial w_{ij}} - \frac{\partial Z}{\partial w_{ij}} = \langle v_i h_j \rangle_{data} - \langle v_i h_j \rangle_{model} \quad (5)$$

In this formula, the expectation value with a *data* subscript is related to the training data, where h_j can be easily calculated by conditional probability, as presented in Equation 6 Tieleman [2008]. Conversely, the expectation with a *model* subscript corresponds to visible and hidden variables values sampled from the current RBM's energy function.

$$p(h_j = 1 | \mathbf{v}) = \sigma(b_j + \sum_i w_{ij} v_i), \text{ where } \sigma(x) = 1 / (1 + \exp(-x)) \quad (6)$$

Although finding the value for $\langle v_i h_j \rangle_{data}$ can be regarded as a trivial task, sampling values from the *model* variables presents a computational challenge. Different methods have been presented to approximate sampled values, such as the Contrastive Divergence (CD) and Persistent Contrastive Divergence (PCD) algorithms [Hinton, 2002; Tieleman, 2008].

3. Framework for evaluating quantum-assisted RBM Training

The training process of RBMs aided by quantum computers has been presented by different studies, such as [Ajagekar and You, 2020] and [Moro and Prati, 2023], with a focus on Quantum Annealing sampling compared against the CD algorithm. This work aims to broaden the research scope, comparing quantum sampling techniques performed by different platforms besides quantum annealers. Additionally, more advanced classical approaches, such as the PCD algorithm, will be used to benchmark traditional learning techniques with quantum sampling.

As a means to perform this task, a package for an agnostic training of RBMs, that can be performed with either quantum or classical algorithms, is being developed. The authors will make use of `QUBO.jl` [Xavier et al., 2023], a Julia ecosystem that provides an interface with different QUBO samplers, in order to create a comprehensive quantum assisted learning tool for RBMs. Having access to contrasting devices, this package will set a framework to not only compare quantum-assisted training against classical algorithms, but also to benchmark this novel technique over distinct QC platforms. A simplified diagram of the training process with quantum sampling is presented in Figure 2, where Θ is one of the hyperparameters.



Figure 2: Simplified diagram of the quantum-assisted training process of an RBM

At this moment, the package has already been used to benchmark the learning process of a database containing health indicators¹, for a varying learning rates ϵ , from 0.001 to 0.0001. The selected platform to produce these preliminary results for quantum-assisted training was a simulated quantum annealing technique. As presented in Figure 3, the classical technique still yielded better results. However, as the tool is applied on more complex datasets, it is expected that quantum-assisted training becomes more competitive.

¹The database is available on Kaggle

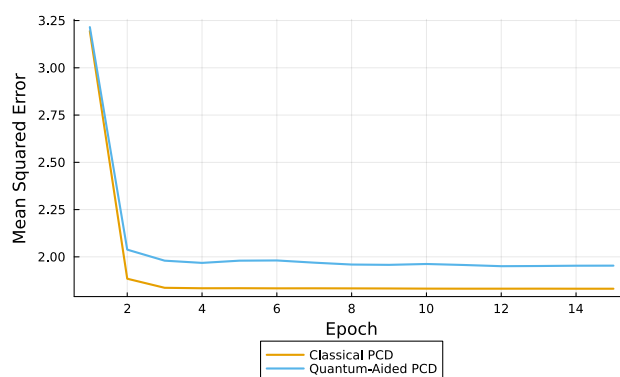


Figure 3: Classical and Quantum-assisted (simulated) training mean Squared Error decay over 15 epochs for a varying learning rate $\epsilon \in [0.001, 0.0001]$

References

- Ajagekar, A. and You, F. (2020). Quantum computing assisted deep learning for fault detection and diagnosis in industrial process systems. *Computers and Chemical Engineering*, 143: 107119. ISSN 0098-1354. URL <http://dx.doi.org/10.1016/j.compchemeng.2020.107119>.
- Bernal, D. E., Ajagekar, A., Harwood, S. M., Stober, S. T., Trennev, D., and You, F. (2022). Perspectives of quantum computing for chemical engineering. *AIChE J.*, 68(6).
- Cerezo, M., Verdon, G., Huang, H.-Y., Cincio, L., and Coles, P. J. (2022). Challenges and opportunities in quantum machine learning. *Nature Computational Science*, 2(9):567–576. ISSN 2662-8457. URL <http://dx.doi.org/10.1038/s43588-022-00311-3>.
- Dawid, A. and LeCun, Y. (2023). Introduction to latent variable energy-based models: A path towards autonomous machine intelligence.
- Hinton, G. E. (2002). Training products of experts by minimizing contrastive divergence. URL <http://dx.doi.org/10.1162/089976602760128018>.
- Hopfield, J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences of the United States of America*, 79: 2554–8.
- Moro, L. and Prati, E. (2023). Anomaly detection speed-up by quantum restricted boltzmann machines. *Communications Physics*, 6(1). ISSN 2399-3650. URL <http://dx.doi.org/10.1038/s42005-023-01390-y>.
- Orus, R., Mugel, S., and Lizaso, E. (2019). Quantum computing for finance: Overview and prospects. *Reviews in Physics*, 4:100028. ISSN 2405-4283. URL <https://www.sciencedirect.com/science/article/pii/S2405428318300571>.
- Tieleman, T. (2008). Training restricted boltzmann machines using approximations to the likelihood gradient. In *International Conference on Machine Learning*. URL <https://api.semanticscholar.org/CorpusID:7330145>.
- Xavier, P. M., Ripper, P., Andrade, T., Garcia, J. D., Maculan, N., and Neira, D. E. B. (2023). Qubo.jl: A julia ecosystem for quadratic unconstrained binary optimization.