Quantum Cohort Project Business Application

For each weekly project, your team is asked to complete the below business application exercise. To complement the technical tasks, please consider the four questions below. You are free to format your response to these four questions as you wish (with the final question done as a short recorded video), and to include the content (or links to the content) on your forked repository.

Step 1: Explain the technical problem you solved in this exercise

There exist many data sets that are multi-dimensional, and, at scale, these data sets grow exponentially. This exponential growth in the data also means exponential growth in computing resources and time. In many cases, given the limits of classical computing, analyzing or simulating these scaled data sets is not possible. Fortunately, machine learning algorithms can help compress this scale in a way that is manageable. In this exercise, we used input data on molecules to train machine learning algorithms which we then used to calculate and graph potential energy curves. We used Restricted Boltzmann Machines (RBM) to construct a binary quadratic model with a significantly small size number of parameters which reproduces the probability distribution observed in the sample input data.

Summary

Actions

- For the hydrogen molecule, we had a large multidimensional set of 10-parameter data for 54 r-equidistant separations of the two hydrogen atoms with each r-sample consisting of 10,000 rows and 2-columns. We wanted to understand the relationship between the potential energy stored in a molecule (E_bond) and the separation (r) between the two atoms.
- We studied the ground state of the Ridberg atom using a sample of size (20000, 100).
- We used this input data to train a machine learning algorithm within an RBM.

Results

- The output, graphs of E_bond vs. r, helped us understand the optimal bond length, or the point at which the attractive and repulsive forces are in equilibrium.
- Reduced/compressed dataset size of the RBM v-, h- biases, and weights per individual r-point was of dimension (2+10)^2=144.

- We found about 50-fold speed up for obtaining the 54-RBMs models (one per r-input parameter). The independent training for 190 epoches per single r-point takes about 50 sec and for all 54 r-points this is a total of 46 min reaching ~1% accuracy. While the same 54-RBMs models could be obtained within 48 sec with less than 0.9% accuracy.
- The speed up technique was used to determine minimalistic (n_h, n_{data}) models that resulted in binding energy accuracy of less than 0.0001 for the Ridberg atoms dataset within 1000 training epochs at most, and input data size less than 5000 out of the 20000 samples available.

Details

Task 1: Potential binding energy of hydrogen molecule as function of the r-distance between the two H-atoms.

The first task of the challenge consists in the training of a Restricted Boltzmann Machines onto a dataset coming from the sampling of a simulation of a H_2 molecule on a quantum computer. The objective is to build the curve of the energy of the system expressed as a function of the distance separating the two hydrogen atoms.

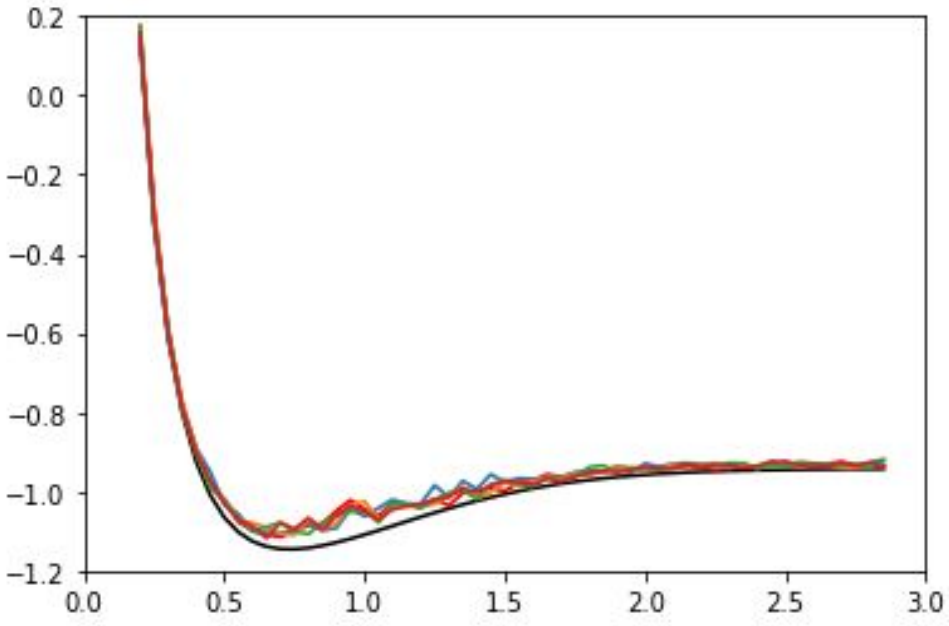
The first idea that might come across our mind is to blindly train the RBM based on the data for each distance in order to retrieve the energy distribution.

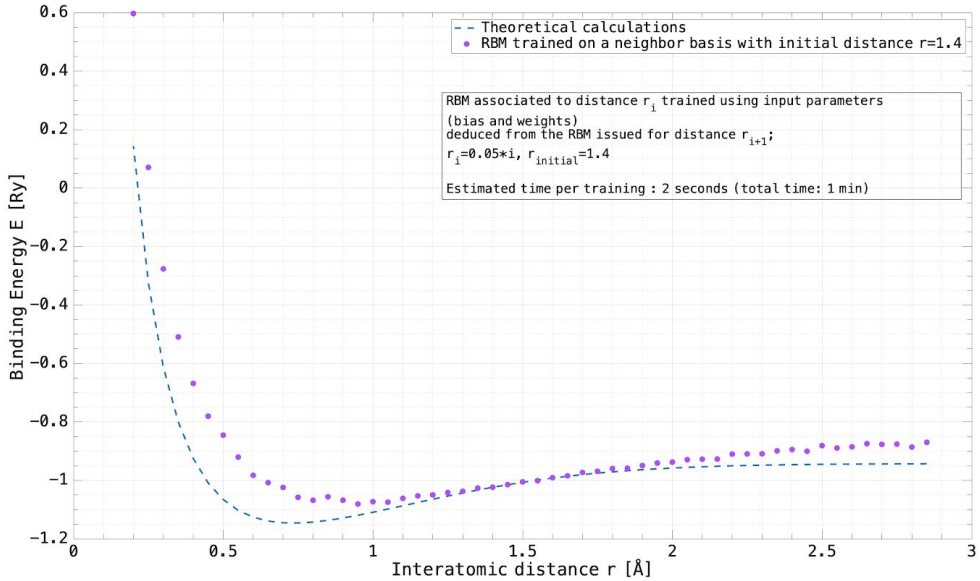
However, proceeding this way will yield a very long calculation (as can be seen a little bit below in the notebook. Is there a way to take advantage of the knowledge of the system to reduce the complexity of the task? In fact, the H_2 system is rather easy to handle directly, and one can use our knowledge of the energy behavior of the interatomic distance to reduce the number of RBM models to be trained from scratch. More precisely, one shall take into account that there are regions for which energy does not change significantly with an incremental increase of the distance ($r \in [1.5,2.5]$). Therefore, the data obtained for each of those values should more or less allow us to retrieve the same conclusions regarding the energy of the system. One can hence think about using one single RBM, trained for a distance value that accounts well for the behavior of its surrounding region for which the associated energy value is close, and train RBMs for this latter with bias issued from the witness point chosen. This method treats the variation of energy around the considered reference point as a small perturbation of the system, which lets us assess that parameters of the RBM trained for a close distance do account small corrections on the ones found for the reference point. Proceeding this way step by step allows an efficient reconstruction of the energy

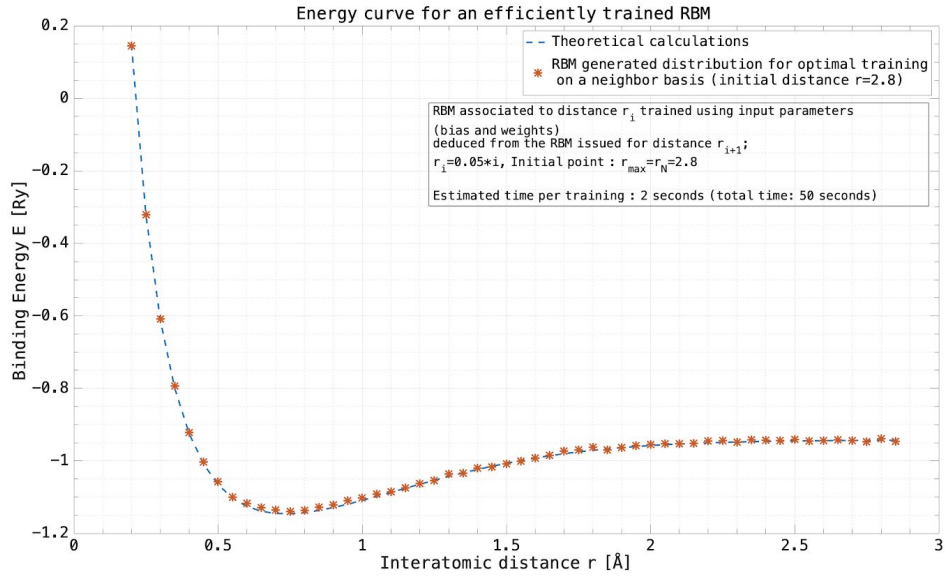
curve, the generated model being adjusted efficiently in a small amount being more performant. The critical point is hence to pick a distance or a set of distances whose associated energy values account well for the global energy distribution.

We uncovered an efficient computational approach that takes advantage of RBM model results at nearby points which significantly reduces the amount of training needed to find the model parameters for all 54 r-separation points.

Instead of starting the training for every r-point with a random initial configuration, we had the insight to use the last iteration at r_n as the starting configuration for the next r_{n+1} point. This works because the RBM parameters describing a system have changed a little and thus they will be also very close to the best solution for nearby r-values. The time saved was 50-fold because the result obtained for one data point was used wisely for the nearby points resulting in getting all 54 RBMs within the time for one r-point evaluation.





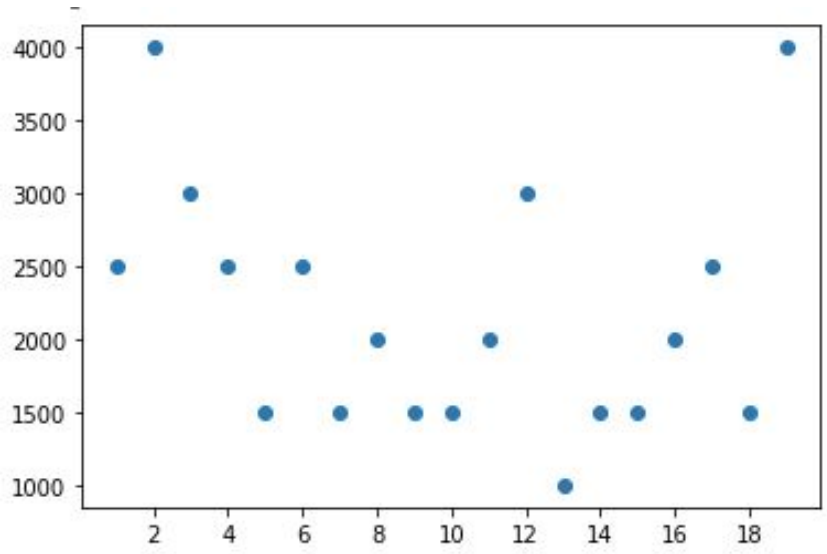


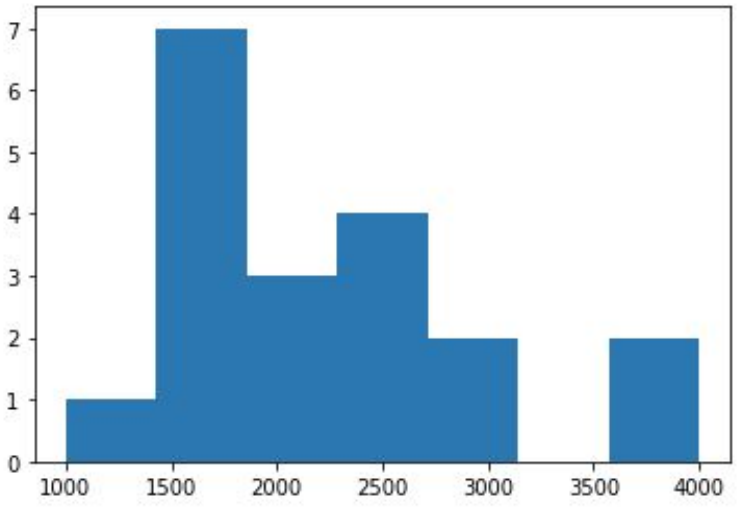
Top: It takes less than a minute per data r-point to do 190 epochs and to reach a few percent accuracy (~1%). Shown are results for 40, 80, 120, 160, and 190 epochs. The black curve corresponds to the exact binding energy values. Total run time is about 50 minutes (num_samples = 500). **Bottom**: In contrast, reusing the weights, and the v- and h- bias vectors by starting from r=2.8 inward one obtains RBMs for all points in less than 50 sec with less than 0.5% accuracy with only a few epochs per each point. **Middle:** The RBM obtained for r=1.4 describes the energy curve nearby but deviates for the points that are far from r=1.4 but maintains the overall trend.

Computational note: in the H_2 example we have 54 points - we can choose to do each RBM training/optimization 54 independent times, thus total time = 54 x $\mathbf{T_{avg}}$ or we can use the RBM_i to produce the next $RBM_{(i+1)}$ in a similar way one can use to interpolate between r-values as well. Note, that $RBM(r_i)$ is not useful to generate data about r_j that is too far from r_i , that is if $|(j-i)|>>1$ then the error is significant.

Task 2: the Rydberg atom - it is remarkable that n_h=1 can reach e-4 accuracy within 1k epochs. Charting the space (n_h, n_{data}) is going to take a lot of computer time unless one uses a trick such as the nearby point RBMs to speed up the exploration by only restarting the RBMs when moving to a new n_h value.

The graphs below show **Top**: the points (n_h, n_{data}) where accuracy less than 0.0001 in the binding energy (E_b = -4.1203519096) was reached for the smallest n_{data} at a given n_h while **Bottom:** is the 7-bins histogram on how often n_{data} resulted in less than 0.0001 accuracy.





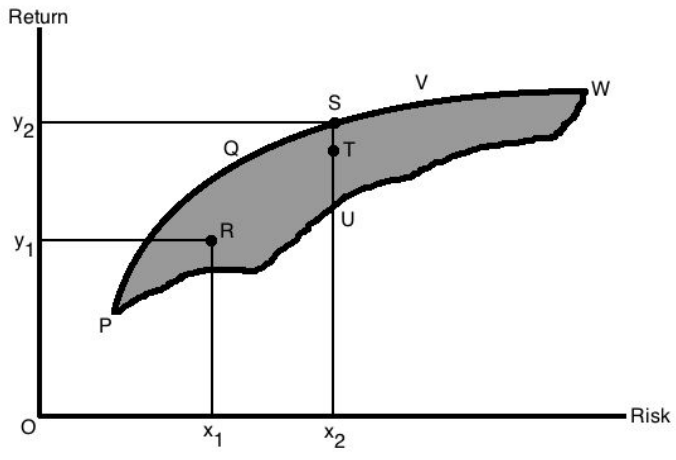
Step 2: Explain or provide examples of the types of real-world problems this solution can solve

There are many real-world problems that this solution can solve. For example:

- Chemistry/Pharmaceutical: Molecular simulations and/or methods based on weighted combination of possible alternative models (examples: H_2, BeH_2, LiH).
- Finance: Portfolio optimization and rebalancing, as well as weighting of multi-strategy methods and evolutionary adjustments;
- State defects removal and noise mitigation for quantum computers and adiabatic annealers, e.g. D-Wave (based on the MNIST pictures reconstruction example).

We will focus here on portfolio optimization problems in finance.

In finance, many investment firms have portfolios of investments with the goal of maximizing returns and minimizing risk. See the chart below where PQVW is called the efficient frontier which represents the highest returns for a given level of risk.



Each asset in your portfolio will have a return and a weight where the weight is the proportion of the asset in the portfolio. When your assets scale, you have the same curse of dimensionality problem as you would in our

chemistry example. So, based on past and current data, we can essentially change the weights of the assets in order to maximize returns and minimize risk.

Portfolio Managers Risk Analysis

Let us consider a team of portfolio managers responsible for actively managing a portfolio of investments in the equities market. Their stock picking decisions are partially guided by unforeseen daily random events (e.g., Covid-19) while its majority is guided with some latent variables such as their underlying investment philosophy and trading strategy. Modelling such latent variables would allow for reconstruction/enumeration of many variants of portfolios subject to the same latent variables in order to further study the trading team's risk characteristics.

This enumeration of potential portfolios would complement historical constructed portfolios managed by the portfolio management team by increasing the sample size for better statistical significance analysis. It would require many years of data to gather enough constructed portfolios by a given PM team in order to draw a proper risk measure with a rigid confidence interval. But a generative model can shorten the length of the time by enumerating similar portfolios that never got the chance to be traded yet.

This problem/solution can be elegantly modeled and solved using *restricted boltzmann machine* (RBM) where the visible layer represent exclusion of a stock ticker in a portfolio (length of visible layer is equal to the possible equities available in the PM's trading universe) while the hidden layer is representing the latent variables dictating the underlying trading strategy and requires proper parameter optimization.

Given the potential size of the input layer and available training data for a given PM team (which is very low), the training process could settle on a local optima as it suffers from the curse of dimensionality and large sparsity. However, since RBM can be modeled using *Ising Model* and solved using a quantum annealer, we argue that using available systems such as D-Wave would further enhance the quality of models. In other words, we could potentially receive positive feedback from Adiabatic Quantum Computer for solving this problem.

Step 3: Identify at least one potential customer for this solution - ie: a business who has this problem and would consider paying to have this problem solved

We would target **portfolio managers (PM)** at the following type financial firms, because these firms have already indicated an interest in quantum (key quantum hires) :

- Hedge funds (Citadel, Two Sigma, DE Shaw)

- Investment banks (Goldman Sachs, Morgan Stanley, JP Morgan)

We have been considering Fidelity and similar investment companies but since we are not aware of their quantum programs we are not targeting them at this stage.

These firms employ large numbers of PMs and properly modelling their individual risk is imperative to the success of these businesses.

Step 4: Prepare a 90 second video explaining the value proposition of your innovation to this potential customer in non-technical language

Value proposition for Portfolio Managers

As a portfolio manager, we understand that your goal is to optimize your portfolios by maximizing the returns of your assets while minimizing risk.

Problem:
At scale, with hundreds of assets, there exists a dimensionality problem that makes it nearly impossible for managers to optimize on a frequent, more than daily, basis.

Solution:
Imagine that you want to build an optimal portfolio for the next year. Ideally, you would want to gather data from 1000's of different strategies and investment decisions over the last three years. Based on what you would learn from this data you could generate an ideal portfolio that minimizes risk while maximizing return. Now, on classical machines this is impossible due to the sheer size of the data. However our firm leverages the power of quantum computing which makes it possible to analyze this quantity of data.

1. **Input:** First, we gather the input data which are three years of historical data on 1000's of firms which includes all of their investment decisions and results.
2. **Mapping:** From this data, we first create an efficient mapping of the data onto layers of information which are understandable by a quantum computer (qubits).
3. **Function:** Then we create a function which allows us to systematically compare strategies and understand the most optimal strategies (technically - cost function embedded into a hamiltonian)

4. **Output:** The output from the quantum computer is an optimal investment strategy, one which has a high return while minimizing risk (sits on the efficient frontier).

So the result is an optimal portfolio that could never have been generated before, and one that you will have more confidence in because of the sheer amount of data that was used!

Links to external sources:

GitHub repo:
https://github.com/VGGatGitHub/CohortProject_2020_week1/tree/master/Project_1_RBM_and_Tomography/solutions_pavy

Video recording:
https://drive.google.com/file/d/10Yxh66ZMqGwhI_Hq_nTN5_FBqhlVmqds/view?usp=sharing