

Quantum Challenge 2022

Youngoh Son, Gyungmin Cho

October 4, 2022

1 Variational Quantum Eigensolver

1.1 Task 1

For spinless fermion system we could think emptied state and occupied state as "down" ($|0\rangle$) and "up" ($|1\rangle$) state of spin system. This means we could match annihilation/creation operator to spin-raising and spin-lowering operators which we could write as

$$a_i^\dagger = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} = \frac{(\sigma_i^x - i\sigma_i^y)}{2} = \sigma_i^-$$

$$a_i = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} = \frac{(\sigma_i^x + i\sigma_i^y)}{2} = \sigma_i^+$$

And as suggested in the question I will let $i < a < j < b$. Then the operator could be represented as below.

$$\begin{aligned} a_b^\dagger a_j a_a^\dagger a_i - a_i^\dagger a_a a_j^\dagger a_b &= \left(\prod_{l=1}^{i-1} (\sigma_l^z)^4 \right) (\sigma_i^z)^3 \sigma_i^+ \left(\prod_{l=i+1}^{a-1} (\sigma_l^z)^3 \right) (\sigma_a^z)^2 \sigma_a^- \left(\prod_{l=a+1}^{j-1} (\sigma_l^z)^2 \right) \sigma_j^- \sigma_j^+ \left(\prod_{l=j+1}^{b-1} \sigma_l^z \right) \sigma_b^- \\ &\quad - \left(\prod_{l=1}^{i-1} (\sigma_l^z)^4 \right) \sigma_i^- (\sigma_i^z)^3 \left(\prod_{l=i+1}^{a-1} (\sigma_l^z)^3 \right) \sigma_a^+ (\sigma_a^z)^2 \left(\prod_{l=a+1}^{j-1} (\sigma_l^z)^2 \right) \sigma_j^- \sigma_j^+ \left(\prod_{l=j+1}^{b-1} \sigma_l^z \right) \sigma_b^+ \\ &= \sigma_i^+ \sigma_{i+1}^z \cdots \sigma_{a-1}^z \sigma_a^- \sigma_j^+ \sigma_{j+1}^z \cdots \sigma_{b-1}^z \sigma_b^- - \sigma_i^- \sigma_{i+1}^z \cdots \sigma_{a-1}^z \sigma_a^+ \sigma_j^- \sigma_{j+1}^z \cdots \sigma_{b-1}^z \sigma_b^+ \end{aligned}$$

$$\begin{aligned} a_b^\dagger a_j a_a^\dagger a_i - a_i^\dagger a_a a_j^\dagger a_b &= \left(\prod_{l=1}^{i-1} (\sigma_l^z)^4 \right) (\sigma_i^z)^3 \sigma_i^+ \left(\prod_{l=i+1}^{a-1} (\sigma_l^z)^3 \right) (\sigma_a^z)^2 \sigma_a^- \left(\prod_{l=a+1}^{j-1} (\sigma_l^z)^2 \right) \sigma_j^- \sigma_j^+ \left(\prod_{l=j+1}^{b-1} \sigma_l^z \right) \sigma_b^- \\ &\quad - \left(\prod_{l=1}^{i-1} (\sigma_l^z)^4 \right) \sigma_i^- (\sigma_i^z)^3 \left(\prod_{l=i+1}^{a-1} (\sigma_l^z)^3 \right) \sigma_a^+ (\sigma_a^z)^2 \left(\prod_{l=a+1}^{j-1} (\sigma_l^z)^2 \right) \sigma_j^- \sigma_j^+ \left(\prod_{l=j+1}^{b-1} \sigma_l^z \right) \sigma_b^+ \\ &= \sigma_i^+ \sigma_{i+1}^z \cdots \sigma_{a-1}^z \sigma_a^- \sigma_j^+ \sigma_{j+1}^z \cdots \sigma_{b-1}^z \sigma_b^- - \sigma_i^- \sigma_{i+1}^z \cdots \sigma_{a-1}^z \sigma_a^+ \sigma_j^- \sigma_{j+1}^z \cdots \sigma_{b-1}^z \sigma_b^+ \end{aligned}$$

Using this formula, we could get the operators of example shown in the problem

(1)

$$\begin{aligned} t_{ijab}(\sigma_1^+ \sigma_2^- \sigma_3^+ \sigma_4^- - \sigma_1^- \sigma_2^+ \sigma_3^- \sigma_4^+) &= it_{ijab}(\sigma_4^x \sigma_3^x \sigma_2^x \sigma_1^y + \sigma_4^x \sigma_3^y \sigma_2^x \sigma_1^x \\ &\quad + \sigma_4^x \sigma_3^y \sigma_2^y \sigma_1^y + \sigma_4^y \sigma_3^x \sigma_2^y \sigma_1^y \\ &\quad - \sigma_4^y \sigma_3^x \sigma_2^x \sigma_1^x - \sigma_4^x \sigma_3^x \sigma_2^y \sigma_1^x \\ &\quad - \sigma_4^y \sigma_3^x \sigma_2^y \sigma_1^y - \sigma_4^y \sigma_3^y \sigma_2^y \sigma_1^x) \end{aligned}$$

(2)

$$\begin{aligned} t_{ijab}(\sigma_1^+ \sigma_2^- \sigma_3^+ \sigma_4^- - \sigma_1^- \sigma_2^+ \sigma_3^- \sigma_4^+) &= it_{ijab}(\sigma_5^x \sigma_4^x \sigma_2^x \sigma_1^y + \sigma_5^x \sigma_4^y \sigma_2^x \sigma_1^x \\ &\quad + \sigma_5^x \sigma_4^y \sigma_2^y \sigma_1^y + \sigma_5^y \sigma_4^x \sigma_2^y \sigma_1^y \\ &\quad - \sigma_5^y \sigma_4^x \sigma_2^x \sigma_1^x - \sigma_5^x \sigma_4^x \sigma_2^y \sigma_1^x \\ &\quad - \sigma_5^y \sigma_4^x \sigma_2^y \sigma_1^y - \sigma_5^y \sigma_4^y \sigma_2^y \sigma_1^x) \end{aligned}$$

Implementing with computer is on Problem1.ipynb

1.2 Task 3

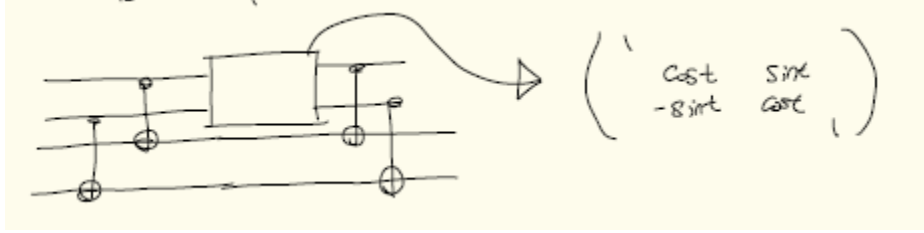
From here, I will use X,Y,Z notation for pauli matrices. Let's arrange the formula above.

$$\begin{aligned} & YYXY + XXXY + XYYY + XYXX - YXYX - YXXX - YYYX - XXYX \\ &= (XX + YY) + (XY(XX + YY) - YX(XX + YY) - (XX + YY)YX) \\ &= (XX + YY)(XY - YX) + (XY - YX)(XX + YY) \end{aligned}$$

Then the operator can be arranged like below

$$\begin{aligned} \Rightarrow e^{t(a_4^\dagger a_3 a_2^\dagger a_1 - a_1^\dagger a_2 a_3^\dagger a_4)} &= e^{\frac{it}{2}(\frac{XX+YY}{2})(\frac{XY-YX}{2})} e^{\frac{it}{2}(\frac{XY-YX}{2})(\frac{XX+YY}{2})} \\ &= e^{-t(|0101\rangle\langle 0101| - |1010\rangle\langle 0101|)} \\ &= CNOT_{13} CNOT_{24} e^{-t(|0001\rangle\langle 0010| - |0010\rangle\langle 0001|)} CNOT_{13} CNOT_{24} \end{aligned}$$

So we could construct a circuit like the diagram



More details are written in the Problem1.ipynb

1.3 Task 2,4,5,6

The works on Task 2,4,5,6 are all in the Problem1.ipynb

2 Generative Adversarial Network Problem

2.1 2.1.1

GAN(Generative Adversarial Networks)의 목표는 훈련에 사용된 원본 데이터와 유사한 데이터를 생성하는 것이다. 이를 위해 생성기와 판별기라는 두 개의 신경망을 함께 사용하면서 학습을 진행하게 된다. 이때 생성기의 역할은 실제 훈련 데이터 세트를 모방하는 가짜 데이터를 생성하고 판별자는 실제 데이터와 가짜 데이터를 구별하는 역할을 한다. 훈련 과정에서 두 네트워크는 상호작용하며 학습시키면서 결국 생성기는 판별자에 최대한 걸리지 않는 훈련 데이터 세트와 매우 유사한 새 데이터를 생성할 수 있게 된다.

생성기는 데이터 세트는 알려지지 않은 데이터 분포 P_{data} 중 일부를 입력 받아 P_{data} 데이터 분포를 파악할 수 있도록 학습한다. 결과적으로 P_{data} 와 완전히 같은 데이터 분포를 생성하는 것이 이상적이다.

이렇게 판별기 D와 생성기 G는 함께 작동하며 아래와 같은 식을 최대화 시켜야 한다. 판별기는 유사데이터 판별 확률을 올리고, 생성자는 판별 확률을 낮추고자 학습이 진행된다.

$$\min_G \max_D V(D, G) = \mathbb{E}_{\mathbf{x} \sim p_{data}} [\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z} \sim p_z} [\log(1 - D(G(\mathbf{z})))]$$

- \mathbf{x} : real data sample
- \mathbf{z} : latent vector
- $D(\mathbf{x})$: probability of the discriminator classifying real data as real
- $G(\mathbf{z})$: fake data
- $D(G(\mathbf{z}))$: probability of discriminator classifying fake data as real

실제 학습과정에서는 아래와 같은 손실함수를 최소화 시키는 방향으로 판별기와 생성기는 학습을 진행하여 최대화 시키고 싶은 확률을 증가시킬 수 있다.

$$L_D = -[y \cdot \log(D(x)) + (1 - y) \cdot \log(1 - D(G(z)))]$$

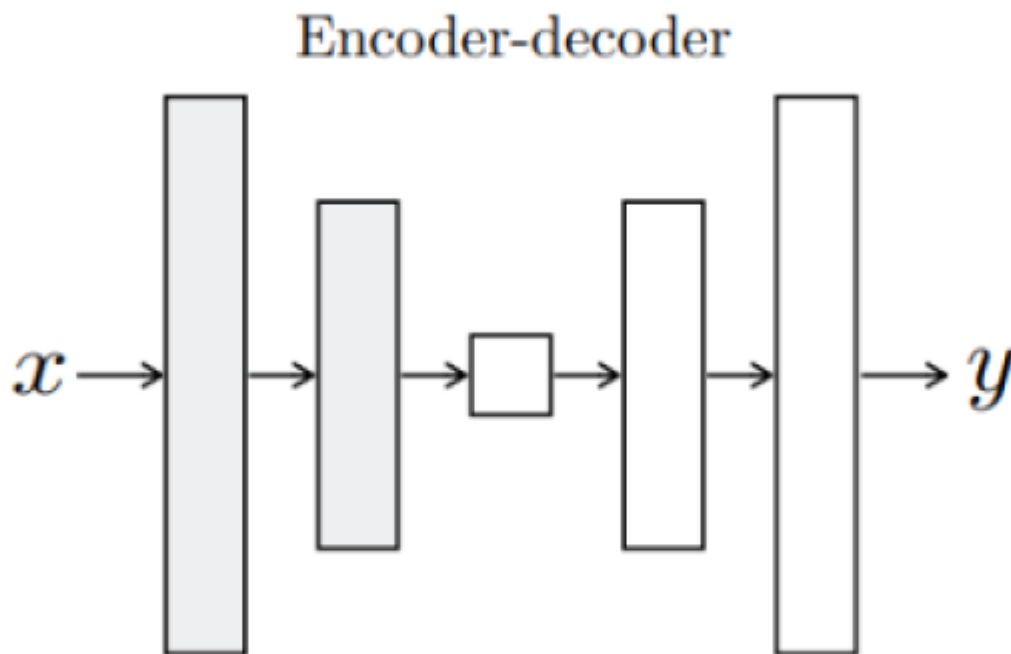
$$L_G = [(1 - y) \cdot \log(1 - D(G(z)))]$$

2.2 2.1.2

The code is written in Problem2_2.ipynb

2.3 2.2.1, 2.2.2

번 문제에서는 pix2pix의 UNet 구조 대신 encoder, decoder 방식을 사용하여 양자회로에서
 수 있는 스케일로 down sizing을 진행후 끝났을 때 이를 Quantization과 함께 upper sizing
 행하고자 한다. 기존의 256 by 256 크기의 구조에서 UNet의 downsizing을 진행하였을 때 5
 기의 벡터가 생성되지만, 양자컴퓨터의 규모적 한계에 의해 16크기의 벡터로 인코딩을 진행
 이를 통해 quantum한 성질을 추가할 수 있도록 스케일을 줄인 뒤 이를 다시 고전적인 방
 로 upper sizing을 진행한다.



위의 그림을 보면 인코더 구조에서 가장 정중앙에서 존재하는 벡터를 인풋으로 받고 양자회
 적절한 Ansatz를 설정한 이후 고전적 CNN layer들과 Ansatz의 파라미터를 학습하는 구조이
 러한 방식을 오직 고전적인 방식으로만 사용했을 때와 하이브리드한 구조를 사용했을 때를
 problem2_2.ipynb와 problem2_2classicML.ipynb을 통해 진행하였다. Unet처럼 decoding 과
 서 encoding data를 가지고 오지 않아서 기존의 pix2pix에 비해 고전, 하이브리드 모두 성능
 족하였지만 하이브리드 구조가 고전인 구조에 비해 확실히 적은 에러를 가지고 있다. 이는
 더 노트북에서 확인 가능하다. 이를 좀더 발전 시켜 pix2pix처럼 encoding data를 concat하
 식을 채택하여 Qpix2pix를 사용하면 훨씬 더 좋은 결과가 있을 것으로 예상된다.

3 Quantum Phase Estimation Challenge

3.1 Task 1

$$H = \frac{1}{2} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

$$|H - \lambda I| = 0$$

$$\lambda^2 - 1 = 0$$

$$\lambda = \pm 1$$

$$\text{for } \lambda = +1 : |\Psi_1\rangle = N_1 \begin{pmatrix} 1 + \sqrt{2} \\ 1 \end{pmatrix}$$

$$\text{for } \lambda = -1 : |\Psi_2\rangle = N_2 \begin{pmatrix} 1 - \sqrt{2} \\ 1 \end{pmatrix}$$

3.2 Task 2

For unitary matrices U , the following formula $UU^\dagger = U^\dagger U = I$ holds. So let λ and v be the eigenvalue and eigenvector of unitary matrices.

$$(Uv)^\dagger \cdot Uv = \lambda^* v^\dagger \lambda v$$

$$v^\dagger U^\dagger U v = |\lambda|^2 \|v\|^2$$

$$v^\dagger I v = |\lambda|^2 \|v\|^2$$

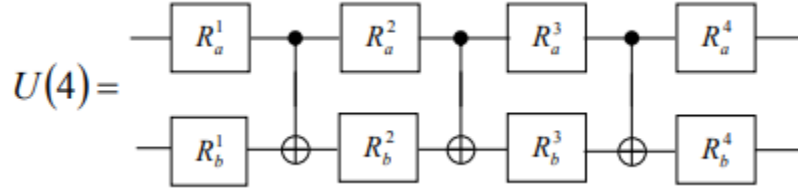
$$\|v\|^2 = |\lambda|^2 \|v\|^2$$

$$\therefore |\lambda|^2 = 1 \Rightarrow \lambda = e^{i\theta} \quad (\theta \in \mathbb{R})$$

So all eigenvalues of unitary matrices could be represented as $e^{i\theta}$

3.3 Task 3

From Quantum Computation and Quantum Information(MA Nielsen) we could say that all arbitrary U could be decomposed to single, two qubit gates. Also in this book they say arbitrary controlled single qubit gate is possible. So what we need to prove that arbitrary controlled two qubit unitary is possible. First, in this paper[UNIVERSAL QUANTUM CIRCUIT FOR n-QUBIT QUANTUM GATE: A PROGRAMMABLE QUANTUM GATE, PAULO BENÍCIO(2006)], it is possible to decompose arbitrary two qubit unitary with 3 cnot gates and 8 single qubit gates as picture below.



So arbitrary controlled two qubit unitary could be decompose into 3 toffoli gates and 8 controlled single qubit gates.

3.4 Task 4

Let

$$|a\rangle = \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_{N-1} \end{pmatrix}, \quad |\phi\rangle = \begin{pmatrix} \phi_0 \\ \phi_1 \\ \vdots \\ \phi_{N-1} \end{pmatrix},$$

$$F_N = \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & 1 & 1 & 1 & \dots & 1 \\ 1 & \omega & \omega^2 & \omega^3 & \dots & \omega^{N-1} \\ 1 & \omega^2 & \omega^4 & \omega^6 & \dots & \omega^{2(N-1)} \\ 1 & \omega^3 & \omega^6 & \omega^9 & \dots & \omega^{3(N-1)} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega^{N-1} & \omega^{2(N-1)} & \omega^{3(N-1)} & \dots & \omega^{(N-1)(N-1)} \end{bmatrix}$$

Then $|\phi\rangle = F_N|a\rangle$ holds. Also we could see that $(\omega^n)^* = \omega^{-n}$ also holds

3.5 Task 5

$$\langle F_i | F_j \rangle = \frac{1}{N} \sum_{n=0}^{N-1} (\omega^{in})^* \omega^{jn} = \frac{1}{N} \sum_{n=0}^{N-1} \omega^{(j-i)n}$$

$$\text{if } i \neq j: \quad \langle F_i | F_j \rangle = \frac{1 - \omega^{(j-i)N}}{1 - \omega^{j-i}} = 0 \quad (\because \omega^{(j-i)N} = 1)$$

$$\text{if } i = j: \quad \langle F_i | F_j \rangle = \frac{1}{N} \sum_{n=0}^{N-1} 1 = 1$$

3.6 Task 6

For N=2 the QFT matrix becomes

$$QFT = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & \omega \end{bmatrix}$$

If we substitute $\omega = -1$ the QFT is

$$QFT = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = H$$

3.7 Task 7,8,9

These problems are solved if Problem3.ipynb