PHSX815_Project4:

Estimating the ground or first excited state energy of a quantum harmonic oscillator using the variational Monte Carlo (VMC) Method

Johnpaul Mbagwu

(Dated: May 8, 2023)

Abstract

This Python code implements a Monte Carlo method for estimating the one-dimensional harmonic oscillator's ground state or first excited state energy using the variational quantum Monte Carlo (VQMC) method. The code takes command line arguments for controlling various parameters, including the seed for the random number generator, the number of Monte Carlo (MC) samples, the energy level (0 for the ground state and 1 for the first excited state), the number of iterations for minimizing the energy, and the seed for the parameter alpha used in the trial wave function. The trial wave functions and the corresponding local energy functions are defined in the code. The code uses the Metropolis algorithm to perform the Monte Carlo integration, and the gradient descent method to minimize the energy. The code also generates two plots, one showing the energy as a function of iteration and the other showing the evolution of the parameter alpha.

1. Introduction

The code provided implements the Variational Monte Carlo (VMC) method to estimate the ground state and first excited state energies of the Harmonic Oscillator (HO) using trial functions and local energy functions. VMC is a quantum Monte Carlo method that approximates the ground state wave function and its energy by minimizing the expectation value of the Hamiltonian with respect to a trial wave function.

The code generates a probability distribution based on the trial function, samples it with a Monte Carlo approach, and then estimates the energy using the local energy function. The optimization of the trial function parameter is performed by minimizing the energy through gradient descent. The code also allows the user to set various parameters such as the number of Monte Carlo samples, the number of iterations for minimization, and the energy level to calculate.

The optimization of the trial wave function is performed by minimizing the energy with respect to a parameter alpha using a gradient descent method. The code uses the NumPy and Matplotlib libraries for numerical calculations and visualization, respectively. The input parameters can be specified in the command line arguments, including the seed, number of Monte Carlo samples, number of iterations for minimization, and the energy level (0 for the ground state and 1 for the first excited state). The output of the code includes the optimized parameters, the estimated energy, and the variance of the energy. Additionally, the code generates plots of the estimated energy and the optimized parameter versus iteration.

Khan and Gao (2007) presented a variational Monte Carlo (VMC) method that works the same way for either the ground or the excited states of a quantum system.

2. Hypotheses behind the estimation of the ground or first excited state energy of a quantum harmonic oscillator using the variational Monte Carlo (VMC) method

The variational Monte Carlo (VMC) method is used to estimate the ground or first excited state energy of a quantum harmonic oscillator. This is done by minimizing the energy expectation value with respect to a trial wave function that depends on a parameter (alpha). The minimization is carried out using a stochastic gradient descent algorithm, where at each iteration, a Monte Carlo integration is performed to estimate the energy and its derivative with respect to alpha. The parameter alpha is then updated using the derivative and a learning rate (gamma).

Two trial wave functions are defined in the code, ftrial0 and ftrial1, which correspond to the ground and first excited state, respectively. The local energy function, Elocal, is also defined for each trial wave function and is used in the Monte Carlo integration to estimate the energy.

The VMC method is based on the variational principle, which states that the energy expectation value of any trial wave function is an upper bound to the ground state energy of the system. Therefore, by minimizing the energy expectation value with respect to the trial wave function parameter, we can obtain an estimate of the ground state energy.

The VMC method is particularly useful when the wave function of the system is not known analytically, as is the case with many quantum mechanical systems. It is also useful when the wave function is known, but too complex to be solved analytically.

3. Code and Experimental Simulation

The code provided is a Python script that performs a Monte Carlo simulation to estimate the energy of the ground state or first excited state of a harmonic oscillator. The simulation uses the Metropolis algorithm to generate random samples from a trial probability distribution and calculates the expectation value of the energy using those samples. The code also performs a minimization of a parameter (either α or β) to find the value that minimizes the energy.

Here is a brief overview of the main parts of the code:

1. The trial functions 'ftrial0' and 'ftrial1' define two different trial wave functions that are used to generate random samples in the Monte Carlo simulation.

2. The local energy functions Elocal0 and Elocal1 define the local energy of the harmonic oscillator for each of the two trial functions.

3. The command line arguments are parsed to set the values of various parameters such as the number of Monte Carlo samples and the number of iterations for the minimization.

4. The PD function defines the probability distribution that is used to generate the random samples in the Monte Carlo simulation. This is simply the square of the trial wave function.

5. The Monte Carlo function performs the Monte Carlo integration by generating random samples from the probability distribution and calculating the local energy for each sample.

6. The for loop in the main program performs a minimization of the parameter (either α or β) by repeatedly performing the Monte Carlo simulation and calculating the energy and its gradient with respect to the parameter. The parameter is updated using the gradient descent method.

7. The plt functions are used to plot the energy and the value of the parameter as a function of the iteration number.

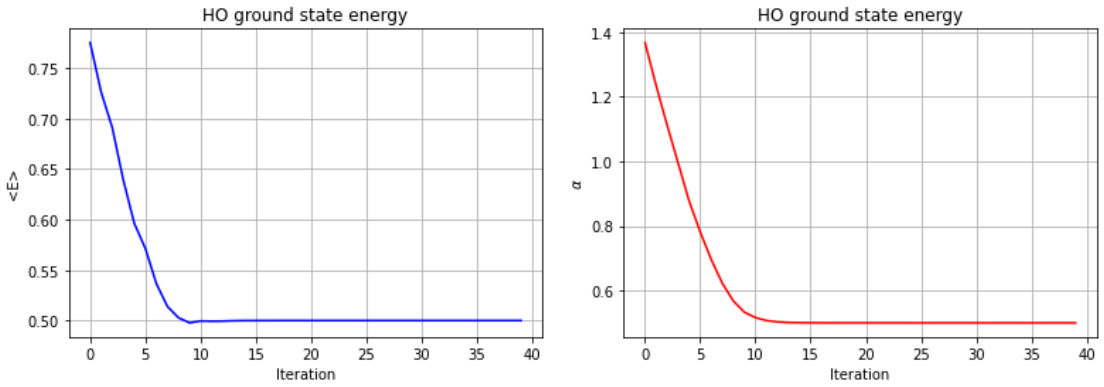

Figure 1: The plots show the variation of the energy and the parameter alpha with the number of iterations. The first plot shows how the energy approaches the minimum value, while the second plot shows the variation of the alpha parameter with the number of iterations. The plots help visualize the convergence of the algorithm and the optimization of the energy with respect to alpha.

4. Analysis

This is a program written in Python that performs variational Monte Carlo simulations to estimate the ground state and first excited state energies of the harmonic oscillator. It uses trial functions and local energy functions to perform the Monte Carlo integration. The program takes several command-line arguments that allow the user to customize the simulation, such as the number of Monte Carlo samples, the number of iterations for minimization, and the value of the parameter alpha used in the trial functions.

The program starts by importing the necessary libraries, such as numpy and matplotlib. It then defines two trial functions, ftrial0 and ftrial1, which are used to generate trial wave functions that will be used in the Monte Carlo integration. Two local energy functions, Elocal0 and Elocal1, are also defined, which are used to calculate the local energy for each trial wave function.

The program then checks for command-line arguments and sets default values for the various parameters used in the simulation, such as the number of Monte Carlo samples, the value of alpha, and the number of iterations for minimization. It then defines the probability distribution PD and the Monte Carlo integration function Monte Carlo, which uses the Metropolis algorithm to generate samples and estimate the energy of the system.

The program then initializes several arrays that will be used to store the energy, alpha, and variance estimates at each iteration of the minimization process. It then performs the minimization process using a loop, in which the Monte Carlo integration function is called to estimate the energy and variance of the system. The energy and variance estimates are then used to update the value of alpha using the gradient descent method. The loop iterates for the specified number of iterations, and the energy and alpha estimates are stored in the arrays defined earlier.

Finally, the program plots the energy and alpha estimate as a function of the iteration number using the matplotlib library. The resulting plots show the convergence of the energy and alpha estimates to their respective ground state or first excited state values.

5. Conclusion

This Python code defines and minimizes the energy of two trial functions for the ground state and first excited state of a quantum harmonic oscillator. It performs a Monte Carlo integration to estimate the expectation value of the local energy, and uses the stochastic gradient descent method to minimize the energy with respect to the variational parameter alpha. The program takes command-line arguments to allow the user to specify the seed for the random number generator, the number of Monte Carlo samples, the value of alpha to start with, and the number of iterations to perform for minimization. If no arguments are specified, the program uses default values.

After minimization, the program plots the energy and variational parameter values for each iteration, as well as the variance of the energy estimates. If the first excited state is minimized, the plot of alpha vs iteration is labelled as beta instead. The code reads command-line arguments using sys.argv, which allows the user to specify input parameters such as the seed, energy level, number of Monte Carlo samples, the initial seed for the parameter to estimate, and the number of iterations for minimization.

The Montecarlo function performs the Monte Carlo integration using the trial wave function and the local energy function. The PD function defines the probability distribution to be sampled in the Monte Carlo integration. The 'for' loop with 'Nwalk' iterations performs the minimization of the parameter (either α for the ground state or β for the first excited state) using the gradient descent method.

References

Imran Khan and Bo Gao. Variance minimization variational monte carlo method. *arXiv preprint physics/0701223*, 2007.