January 15-19: Quantum Computing, Quantum Machine Learning and Quantum Information Theories

Morten Hjorth-Jensen[1,2]

[1]Department of Physics and Center for Computing in Science Education, University of Oslo, Norway
[2]Department of Physics and Astronomy and Facility for Rare Isotope Beams, Michigan State University, East Lansing, Michigan, USA

Dec 9, 2023

Overview of week first week, Basic Notions of Quantum Mechanics

Basics in Linear Algebra, The Hilbert Space, Operators on Hilbert Spaces, States and qubits.

1. Mathematical notation, Hilbert spaces and operators

2. Description of Quantum Systems and one-qubit systems

3. States in Hilbert Space, pure and mixed states

4. Operators and gates

Reading recommendation: Scherer, Mathematics of Quantum Computations, chapter 2

Practicalities

1. Although the course is defined as a self-study course, we can have weekly lectures with small weekly exercise assignments

2. We plan to work on two projects which will define the content of the course, the format can be agreed upon by the participants but the following topics can be included

   - Quantum computing and simulation of quantum mechanical model systenms

- Continuation of the first topic to more realistic systems or applications of quantum machine learning algorithms

3. Final oral examination to be agreed upon

4. All info at the GitHub address https://github.com/CompPhysics/QuantumComputingMachineLearning

Defining basis states and quantum mechanical operators

We start by defining a state vector $\boldsymbol{x}$ (meant to represent various quantum mechanical degrees of freedom) with n components as

$$\boldsymbol{x} = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \dots \\ \dots \\ x_{n-1} \end{bmatrix}.$$

Throughout these notes we will use the so-called Dirac **bra-ket** formalism and we will replace the above standard boldfaced notation for a vector with

$$\boldsymbol{x} = |x\rangle = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \dots \\ \dots \\ x_{n-1} \end{bmatrix},$$

and

$$\boldsymbol{x}^\dagger = \langle x| = \begin{bmatrix} x_0^* & x_1^* & x_2^* & \dots & \dots & x_{n-1}^* \end{bmatrix},$$

With a given vector $|x\rangle$, we define the inner product as

$$\langle x|x\rangle = \sum_{i=0}^{n-1} x_i^* x_i = x_0^2 + x_1^2 + \cdots + x_{n-1}^2.$$

For two arbitrary vectors $|x\rangle$ and $|y\rangle$ with the same lentgh, we have the general expression

$$\langle y|x\rangle = \sum_{i=0}^{n-1} y_i^* x_i = y_0^* x_0 + y_1^* x_1 + \cdots + y_{n-1}^* x_{n-1}.$$

Note well that the inner product $\langle x|x\rangle$ is always a real number while for a two different vectors $\langle y|x\rangle$ is in general not equal to $\langle x|y\rangle$, as can be seen from the following example

We note in bypassing that $|x\rangle^\dagger = \langle x|$, $\langle x|^\dagger = |x\rangle$ and $(|x\rangle^\dagger)^\dagger = |x\rangle$.

Examples. Let us assume that $|x\rangle$ is given by

$$|x\rangle = \begin{bmatrix} 1 - \imath \\ 2 + \imath \end{bmatrix}.$$

The inner product gives us

$$\langle x|x \rangle = (1 + \imath)(1 - \imath) + (2 - \imath)(2 + \imath) = 7,$$

a real number. We can use the norm/inner product to normalize the vector $|x\rangle$ and obtain

$$|x\rangle = \frac{1}{\sqrt{7}} \begin{bmatrix} 1 - \imath \\ 2 + \imath \end{bmatrix}.$$

As another example, consider the two vectors

$$|x\rangle = \begin{bmatrix} -1 \\ 2\imath \\ 1 \end{bmatrix},$$

and

$$|y\rangle = \begin{bmatrix} 1 \\ 0\imath \\ \imath \end{bmatrix}.$$

We see that the inner products $\langle x|y \rangle = -1 + \imath$, which is not the same as $\langle y|x \rangle = -1 - \imath$. This leads to the important rule

$$\langle x|y \rangle^* = \langle y|x \rangle.$$

Outer products

In addition to inner products between vectors/states, the outer product plays a central role in all of quantum mechanics. It is defined as

$$|x\rangle\langle y| = \begin{bmatrix} x_0 y_0^* & x_0 y_1^* & x_0 y_2^* & \cdots & \cdots & x_0 y_{n-2}^* & x_0 y_{n-1}^* \\ x_1 y_0^* & x_1 y_1^* & x_1 y_2^* & \cdots & \cdots & x_1 y_{n-2}^* & x_1 y_{n-1}^* \\ x_2 y_0^* & x_2 y_1^* & x_2 y_2^* & \cdots & \cdots & x_2 y_{n-2}^* & x_2 y_{n-1}^* \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{n-2} y_0^* & x_{n-2} y_1^* & x_{n-2} y_2^* & \cdots & \cdots & x_{n-2} y_{n-2}^* & x_{n-2} y_{n-1}^* \\ x_{n-1} y_0^* & x_{n-1} y_1^* & x_{n-1} y_2^* & \cdots & \cdots & x_{n-1} y_{n-2}^* & x_{n-1} y_{n-1}^* \end{bmatrix}$$

Different operators and gates

In quantum computing, the so-called Pauli matrices, and other simple 2×2 matrices, play an important role, ranging from the setup of quantum gates to a rewrite of creation and annihilation operators and other quantum mechanical operators. Let us start with the familiar Pauli matrices and remind ourselves of some of their basic properties.

The Pauli matrices are defined as

$$\sigma_x = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix},$$

$$\sigma_y = \begin{bmatrix} 0 & -\imath \\ \imath & 0 \end{bmatrix},$$

and

$$\sigma_z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}.$$

It is easy to show that the matrices obey the properties (being involutory)

$$\sigma_x \sigma_x = \sigma_y \sigma_y = \sigma_z \sigma_z = I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},$$

that is their products with themselves result in the identity matrix I. Furthermore, the Pauli matrices are unitary matrices meaning that their inverses are equal to their hermitian conjugated matrices. The determinants of the Pauli matrices are all equal to -1, as can be easily verified.

The Pauli matrices obey also the following commutation rules

$$[\sigma_x, \sigma_y] = 2\imath \sigma_z.$$

Before we proceed with other matrices and how they can be used to operate on various quantum mechanical states, let us try to define various basis sets and their pertinent notations. We will often refer to these basis states as our computational basis.

Definition of Computational basis states

Assume we have a two-level system where the two states are represented by the state vectors $|\phi_0\rangle$ and $|\phi_1\rangle$, respectively. These states could represent selected or effective degrees of freedom for either a single particle (fermion or boson) or they could represent effective many-body degrees of freedon. In actual realizations of quantum computing we search often for candidate systems where we can use some low-lying states as computational basis states. But we are not limited to quantum computing. When doing many-body physics, due to the exploding degrees of freedom, we normally search after effective ways by which we can reduce the involved dimensionalities to a number of degrees of freedom we can handle by a given many-body method.

Examples: Hydrogen like states and the harmonic oscillator in one, two and three dimensions.   We will now relabel the above two states as two orthogonal and normalized basis (ONB) states

$$|\phi_0\rangle = |0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix},$$

and

$$|\phi_1\rangle = |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

It is straight forward to see that $\langle 1|0\rangle = 0$. With these two states we can define the define the identity operator $\boldsymbol{I}$ as the sum of the outer products of these two states, namely

$$\boldsymbol{I} = \sum_{i=0}^{i=1} |i\rangle\langle i| = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

We can further define the projection operators

$$\boldsymbol{P} = |0\rangle\langle 0| = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix},$$

and

$$\boldsymbol{Q} = |1\rangle\langle 1| = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}.$$

We note that $P^2 = P$, $Q^2 = Q$ (the operators are idempotent) and that their determinants are zero, meaning in turn that we cannot use these operators for unitary/orthogonal transformations. However, they play important roles in defining effective Hilbert spaces for many-body studies. Finally, before proceeding we note also that the two matrices commute and we have $\boldsymbol{PQ} = 0$ and $[\boldsymbol{P}, \boldsymbol{Q}] = 0$.

Superposition and more. Using the properties of ONBs we can expand a new state in terms of the above states. These states could also form a basis which is an eigenbasis of a selected Hamiltonian (more of this below).

We define now a new state which is a linear expansion in terms of these computational basis states

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle,$$

where the coefficients $\alpha = \langle 0|\psi\rangle$ and $\beta = \langle 1|\psi\rangle$ reresent the overlaps between the computational basis states and the state $|\psi\rangle$. In quantum speech, we say the state is in a superposition of the states $|0\rangle$ and $|1\rangle$.

Computing the inner product of $|\psi\rangle$ we obtain

$$\langle\psi|\psi\rangle = |\alpha|^2\langle 0|0\rangle + |\beta|^2\langle 1|1\rangle = |\alpha|^2 + |\beta|^2 = 1,$$

since the new basis, which is defined in terms of a a unitary/orthogonal transformation, preserves the orthogonality and norm of the original computational basis $|0\rangle$ and $|1\rangle$. To see this, consider the unitary transformation (show derivation of preserving orthogonality).

If we now act with the projection operators $\boldsymbol{P}$ and $\boldsymbol{Q}$ on the state $|\psi\rangle$ we get

$$\boldsymbol{P}|\psi\rangle = |0\rangle\langle 0|(\alpha|0\rangle + \beta|1\rangle) = \alpha|0\rangle,$$

that is we **project** out the $|0\rangle$ component of the state $|\psi\rangle$ with the coefficient α while $\boldsymbol{Q}$ projects out the $|1\rangle$ component with coefficient β as seen from

$$\boldsymbol{Q}|\psi\rangle = |1\rangle\langle 1|(\alpha|0\rangle + \beta|1\rangle) = \beta|1\rangle.$$

The above results can easily be derived by multiplying the pertinent matrices with the vectors $|0\rangle$ and $|1\rangle$, respectively.

Using the above linear expansion we can now define the density matrix of the state $|\psi\rangle$ as the outer product

$$\boldsymbol{\rho} = |\psi\rangle\langle\psi| = \alpha\alpha^*|0\rangle\langle 0| + \alpha\beta^*|0\rangle\langle 1| + \beta\alpha^*|1\rangle\langle 0| + \beta\beta^*|1\rangle\langle 1| = \begin{bmatrix} \alpha\alpha^* & \alpha\beta^* \\ \beta\alpha^* & \beta\beta^* \end{bmatrix}.$$

Finally, we note that the trace of the density matrix is simply given by unity

$$\mathrm{tr}\boldsymbol{\rho} = \alpha\alpha^* + \beta\beta^* = 1.$$

Other important matrices. We end with presenting other operators (as matrices) which play an important role in quantum computing, the so-called Hadamard matrix (or gate as we will use later)

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

The action of the operator $\boldsymbol{H}$ on a computational basis state like $|0\rangle$ gives

$$\boldsymbol{H}|0\rangle = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle),$$

and

$$\boldsymbol{H}|1\rangle = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle),$$

that is we create a superposition of the states $|0\rangle$ and $|1\rangle$.

Another famous operation is the phase matrix given by

$$\boldsymbol{S} = \begin{bmatrix} 1 & 0 \\ 0 & \imath \end{bmatrix}.$$

Eigenvalues and spectral decomposition.

Tensor products

Consider now two vectors with length $n = 2$, with elements

$$|x\rangle = \begin{bmatrix} x_0 \\ x_1 \end{bmatrix},$$

and
$$|y\rangle = \begin{bmatrix} y_0 \\ y_1 \end{bmatrix}.$$

The tensor product of these two vectors is defined as

$$|x\rangle \otimes |y\rangle = |xy\rangle = \begin{bmatrix} x_0 y_0 \\ x_0 y_1 \\ x_1 y_0 \\ x_1 y_1 \end{bmatrix},$$

which is now a vector of length 4.

If we now go back to our original one-qubit basis states, we can form teh following tensor products

$$|0\rangle \otimes |0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = |00\rangle,$$

$$|0\rangle \otimes |1\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} = |01\rangle,$$

$$|1\rangle \otimes |0\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} = |10\rangle,$$

and finally

$$|1\rangle \otimes |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = |11\rangle.$$

We have now four different states and we could try to make a new list by relabeling the states as follows $|00\rangle = |0\rangle$, $|01\rangle = |1\rangle$, $|10\rangle = |2\rangle$, $|11\rangle = |3\rangle$.

In similar ways we can define the tensor product of three qubits (or single-particle states) as

$$|0\rangle \otimes |0\rangle \otimes |0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = |000\rangle,$$

which is a new vector of length eight. We note that with a single-particle basis given the states $|0\rangle$ and $|1\rangle$ we can, with N particles construct 2^N different states. This is something we can generalize to

- discuss ways of labeling states

- how to write a code which does it

The tensor product of two 2×2 matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ is given by

$$\boldsymbol{A} \times \boldsymbol{B} = \begin{bmatrix} a_{00} & a_{01} \\ a_{10} & a_{11} \end{bmatrix} \otimes \begin{bmatrix} b_{00} & b_{01} \\ b_{10} & b_{11} \end{bmatrix} = \begin{bmatrix} a_{00}b_{00} & a_{00}b_{01} & a_{01}b_{00} & a_{01}b_{01} \\ a_{00}b_{10} & a_{00}b_{11} & a_{01}b_{10} & a_{01}b_{11} \\ a_{10}b_{00} & a_{10}b_{01} & a_{11}b_{00} & a_{11}b_{01} \\ a_{10}b_{10} & a_{10}b_{11} & a_{11}b_{10} & a_{11}b_{11} \end{bmatrix}$$