1. Pure State

In Qiskit, we can use the quantum_info module to represent quantum states either in state vector notation, or in the density matrix representation. For convenience, we will import this module as qi:

In [1]:

```
from qiskit import QuantumCircuit
import qiskit.quantum_info as qi
```

Let's consider, for example, the following two-qubit, maximally-entangled pure state:

$$|\psi_{AB}\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\0\\0\\1\end{bmatrix}$$

The density matrix representation for this state is then given by:

$$\rho_{AB} = |\psi_{AB}\rangle\langle\psi_{AB}|$$

$$\rho_{AB} = \left(\frac{1}{\sqrt{2}}\begin{bmatrix}1\\0\\0\\1\end{bmatrix}\right)\left(\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 0 & 0 & 1\end{bmatrix}\right)$$

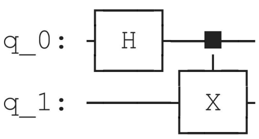
$$\rho_{AB} = \frac{1}{2}\begin{bmatrix}1 & 0 & 0 & 1\\0 & 0 & 0 & 0\\0 & 0 & 0 & 0\\1 & 0 & 0 & 1\end{bmatrix}$$

Let's once again consider the entangled pure state |ψ_AB⟩. We can prepare this state by applying a Hadamard gate to the first qubit, and an CNOT between the first and second qubits:

In [2]:

```
qc_AB = QuantumCircuit(2)
qc_AB.h(0)
qc_AB.cx(0,1)
qc_AB.draw()
```

Out[2]:



To obtain the state constructed by our QuantumCircuit in state vector notation, we can make use of the Statevector.from_instruction() class method from the quantum_info module as follows:

In [3]:

```
psi_AB = qi.Statevector.from_instruction(qc_AB)
psi_AB.draw('latex', prefix='|\\psi_{AB}\\rangle = ')
```

Out[3]:

$$|\psi_{AB}\rangle = \begin{bmatrix}\frac{1}{\sqrt{2}} & 0 & 0 & \frac{1}{\sqrt{2}}\end{bmatrix}$$

Similarly, we can use the DensityMatrix.from_instruction() class method to obtain density matrix representation for this same state:

In [4]:
```
rho_AB = qi.DensityMatrix.from_instruction(qc_AB)
rho_AB.draw('latex', prefix='\\rho_{AB} = ')
```

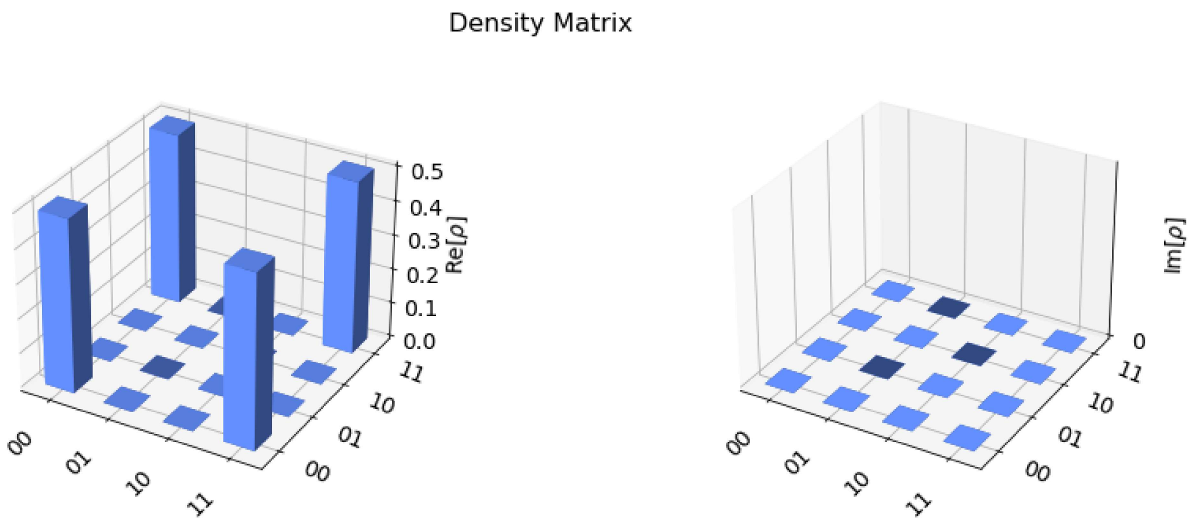
Out[4]:

$$\rho_{AB} = \begin{bmatrix} \frac{1}{2} & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \frac{1}{2} & 0 & 0 & \frac{1}{2} \end{bmatrix}$$

As expected, the result matches our calculation. We can also visualize the density matrix using a cityscape plot of the state:

In [5]:
```
from qiskit.visualization import plot_state_city
plot_state_city(rho_AB.data, title='Density Matrix')
```

Out[5]:



2. Mixed State

In Qiskit, we can define the density matrix of mixed states by directly inputting the matrix values into the DensityMatrix class:

In [6]:
```
import numpy as np
rho_H_matrix = np.array([[1/2,np.sqrt(3)/20 + 2/5],[np.sqrt(3)/20 + 2/5,1/2]])
rho_H = qi.DensityMatrix(rho_H_matrix)
rho_H.draw('latex', prefix='\\rho_H = ')
```

Out[6]:

$$\rho_H = \begin{bmatrix} \frac{1}{2} & 0.4866 \\ 0.4866 & \frac{1}{2} \end{bmatrix}$$

3. Unitary Evolution

We can also use Qiskit to evolve mixed states through unitary operators. Let's first define our state by using the DensityMatrix.from_label() method:

In [7]:
```
rho_0 = 1/3*qi.DensityMatrix.from_label('1') + 2/3*qi.DensityMatrix.from_label('+')
rho_0.draw('latex', prefix='\\rho_0 = ')
```

Out[7]:

$$\rho_0 = \begin{bmatrix} \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{2}{3} \end{bmatrix}$$

We can now define our operator in a similar way, but using the matrix Operator class:

In [8]:
```
from qiskit.visualization import array_to_latex
Y = qi.Operator.from_label('Y')
array_to_latex(Y.data, prefix='Y =')
```

Out[8]:

$$Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$$

Lastly, we can evolve our mixed state by using the DensityMatrix.evolve() method:

In [9]:
```
rho_0p = rho_0.evolve(Y)
rho_0p.draw('latex', prefix='\\rho\'_0 = ')
```

Out[9]:

$$\rho_0' = \begin{bmatrix} \frac{2}{3} & -\frac{1}{3} \\ -\frac{1}{3} & \frac{1}{3} \end{bmatrix}$$

Here, $\hat{U}^\dagger$ is the conjugate transpose of the operator $\hat{U}$. So, for example, let's consider the following mixed state:

$$\rho_0 = \frac{1}{3}|1\rangle\langle 1| + \frac{2}{3}|+\rangle\langle +|$$

$$= \frac{1}{3}\begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} + \frac{2}{3}\begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

$$= \begin{bmatrix} \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{2}{3} \end{bmatrix}$$

If we are interested in knowing what will be the resulting state ρ_0' after a Y Pauli gate is applied, we then have:

$$\rho_0' = Y\rho_0 Y^\dagger = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \begin{bmatrix} \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{2}{3} \end{bmatrix} \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$$

$$= \begin{bmatrix} \frac{2}{3} & -\frac{1}{3} \\ -\frac{1}{3} & \frac{1}{3} \end{bmatrix}$$

4. State Purity

In Qiskit, we can easily extract the purity of a density matrix by using the purity() class method. For example, for the pure state $|+\rangle$, we should expect to see a purity of 1:

In [10]:
```
rho_p = qi.DensityMatrix.from_label('+')
display(rho_p.draw('latex', prefix='\\rho_p = '))
gamma_p = rho_p.purity()
print("State purity: ", np.round(np.real(gamma_p),3))
```

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

```
State purity:  1.0
```

And, for a mixed state, like $\rho_m = \frac{1}{2}|0\rangle\langle 0| + \frac{1}{2}|1\rangle\langle 1|$, we expect a purity of less than 1:

In [11]:
```
rho_m = 1/2*(qi.DensityMatrix.from_label('0') + qi.DensityMatrix.from_label('1'))
display(rho_m.draw('latex', prefix='\\rho_m = '))
gamma_m = rho_m.purity()
print("State purity: ", np.round(np.real(gamma_m),3))
```

$$\rho_m = \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{bmatrix}$$

```
State purity:  0.5
```

5. Reduced Density Matrix

In Qiskit, we can easily obtain the reduced density matrix of a system by using the partial_trace() function and passing the density matrix of the composite system, and a list with the subsystems to trace over:

In [12]:
```
rho_B = qi.partial_trace(rho_AB,[0])
rho_A = qi.partial_trace(rho_AB,[1])

display(rho_B.draw('latex', prefix=" \\rho_{B} = "),
        rho_A.draw('latex', prefix=" \\rho_{A} = "))
```

$$\rho_B = \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{bmatrix}$$

$$\rho_A = \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{bmatrix}$$

In this specific example, ρA and ρB are equal, but this is not always the case.

$$|\psi_{AB}\rangle = \frac{1}{\sqrt{2}}(|0_A 0_B\rangle + |1_A 1_B\rangle)$$

←

This system is then composed of single-qubit subsystem A with basis vectors $\{|\xi_1\rangle, |\xi_2\rangle\} = \{|0_A\rangle, |1_A\rangle\}$, and single-qubit subsystem B with basis vectors $\{|\chi_1\rangle, |\chi_2\rangle\} = \{|0_B\rangle, |1_B\rangle\}$. We know that this system is not separable (i.e., $|\chi_{AB}\rangle \neq |\chi_A\rangle \otimes |\chi_B\rangle$); however, by using the reduced density matrix, we can find a full description for subsystems A and B as follows.

The density matrix of our state $|\psi_{AB}\rangle$ can be expressed in terms of outer products of the basis vectors as:

$$\rho_{AB} = |\psi_{AB}\rangle\langle\psi_{AB}| = \frac{1}{2}[|0_A 0_B\rangle\langle 0_A 0_B| + |0_A 0_B\rangle\langle 1_A 1_B| + |1_A 1_B\rangle\langle 0_A 0_B| + |1_A 1_B\rangle\langle 1_A 1_B|]$$

Now, to calculate the reduced density matrix for, let's say, subsystem B, we have:

$$\begin{aligned}
\rho_B &= \text{Tr}_A(\rho_{AB}) \\
&= \frac{1}{2}[\text{Tr}_A(|0_A 0_B\rangle\langle 0_A 0_B|) + \text{Tr}_A(|0_A 0_B\rangle\langle 1_A 1_B|) + \text{Tr}_A(|1_A 1_B\rangle\langle 0_A 0_B|) + \text{Tr}_A(|1_A 1_B\rangle\langle 1_A 1_B|)] \\
&= \frac{1}{2}[\text{Tr}(|0_A\rangle\langle 0_A|)|0_B\rangle\langle 0_B| + \text{Tr}(|0_A\rangle\langle 1_A|)|0_B\rangle\langle 1_B| + \text{Tr}(|1_A\rangle\langle 0_A|)|1_B\rangle\langle 0_B| + \text{Tr}(|1_A\rangle\langle 1_A|)|1_B\rangle\langle 1_B|] \\
&= \frac{1}{2}[\langle 0_A|0_A\rangle|0_B\rangle\langle 0_B| + \langle 1_A|0_A\rangle|0_B\rangle\langle 1_B| + \langle 0_A|1_A\rangle|1_B\rangle\langle 0_B| + \langle 1_A|1_A\rangle|1_B\rangle\langle 1_B|] \\
&= \frac{1}{2}[|0_B\rangle\langle 0_B| + |1_B\rangle\langle 1_B|] \\
&= \frac{1}{2}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}
\end{aligned}$$

6. Mixed state in the Bloch Sphere

$$\rho = \frac{1}{2}\begin{bmatrix} 1 + r_z & r_x - ir_y \\ r_x + ir_y & 1 - r_z \end{bmatrix}$$

←

Let's reconsider the example from section 2, where we obtained the density matrix:

$$\rho_H = \begin{bmatrix} \frac{1}{2} & \frac{\sqrt{3}}{20} + \frac{2}{5} \\ \frac{\sqrt{3}}{20} + \frac{2}{5} & \frac{1}{2} \end{bmatrix}$$

It is easy to see that, for this particular case:

$$r_x = 2\left(\frac{\sqrt{3}}{20} + \frac{2}{5}\right), \quad r_y = 0, \quad r_z = 0$$

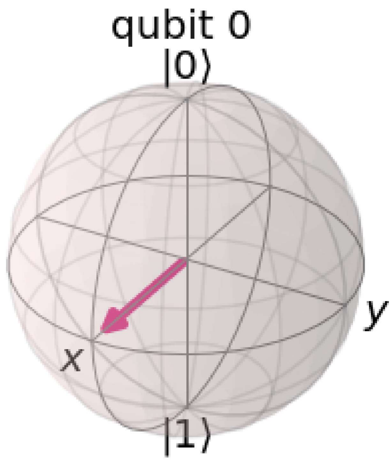
This means that, in the bloch sphere, ρ_H is represented by a vector $\vec{r}$ extending from the origin in the positive x direction, with length $r_x \approx 0.973$.

In Qiskit, we can use the plot_bloch_multivector() function to plot the density matrix of our mixed state in the Bloch sphere:

In [14]:

```
from qiskit.visualization import plot_bloch_multivector
plot_bloch_multivector(rho_H.data)
```

Out[14]:



So, as expected, we get a vector along the positive x axis, with length slightly smaller than 1. This is a very convenient way of expressing that this state is not pure because it has been corrupted by noise.

One last scenario we might want to consider is the possibility of visualizing multi-qubit states using multiple Bloch spheres. We previously learned that, by the use of the reduced density matrix, we can actually find a representation for each individual part that makes up a composite state, even if the state is entangled. For instance, let's look at the following partially-entangled state:

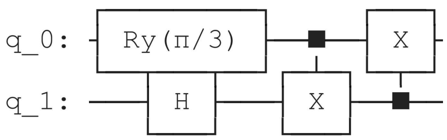
$$|\psi_{CD}\rangle = \frac{1}{2\sqrt{2}}\left(\sqrt{3}|0_C 0_D\rangle + |0_C 1_D\rangle + |1_C 0_D\rangle + \sqrt{3}|1_C 1_D\rangle\right).$$

Here, $|0_C 0_D\rangle$ and $|1_C 1_D\rangle$ have the same probability of occurrence, but are 3 times more likely to occur than $|0_C 1_D\rangle$ and $|1_C 0_D\rangle$. Since this is an entangled state, we know that it is not separable (i.e., $|\psi_{CD}\rangle \neq |\psi_C\rangle \otimes |\psi_D\rangle$); therefore, this state cannot be represented in terms of unit vectors in two individual Bloch spheres. However, by expressing C and D in terms of their reduced density matrices ρ_C and ρ_D, we can then visualize the composite state as two Bloch vectors, one for each of these matrices.

In Qiskit, state $|\psi_{CD}\rangle$ can be generated using the following circuit:

In [15]:
```
qc_CD = QuantumCircuit(2)
qc_CD.ry(np.pi/3,0)
qc_CD.h(1)
qc_CD.cx(0,1)
qc_CD.cx(1,0)
qc_CD.draw()
```

Out[15]:



In [16]:
```
psi_CD = qi.Statevector.from_instruction(qc_CD)
psi_CD.draw('latex', prefix='|\\psi_{CD}\\rangle =')
```

Out[16]:

$$|\psi_{CD}\rangle = \begin{bmatrix} 0.61237 & \frac{1}{\sqrt{8}} & \frac{1}{\sqrt{8}} & 0.61237 \end{bmatrix}$$

In [17]:
```
rho_CD = qi.DensityMatrix.from_instruction(qc_CD)
rho_CD.draw('latex', prefix='\\rho_{CD} =')
```

Out[17]:

$$\rho_{CD} = \begin{bmatrix} \frac{3}{8} & 0.21651 & 0.21651 & \frac{3}{8} \\ 0.21651 & \frac{1}{8} & \frac{1}{8} & 0.21651 \\ 0.21651 & \frac{1}{8} & \frac{1}{8} & 0.21651 \\ \frac{3}{8} & 0.21651 & 0.21651 & \frac{3}{8} \end{bmatrix}$$

In [18]:

```
rho_D = qi.partial_trace(rho_CD,[0])
rho_C = qi.partial_trace(rho_CD,[1])

display(rho_D.draw('latex', prefix=" \\rho_{D} = "),
        rho_C.draw('latex', prefix=" \\rho_{C} = "))
```

$$\rho_D = \begin{bmatrix} \frac{1}{2} & 0.43301 \\ 0.43301 & \frac{1}{2} \end{bmatrix}$$

$$\rho_C = \begin{bmatrix} \frac{1}{2} & 0.43301 \\ 0.43301 & \frac{1}{2} \end{bmatrix}$$

Each of these reduced density matrices has a Bloch vector associated with them, each with components:

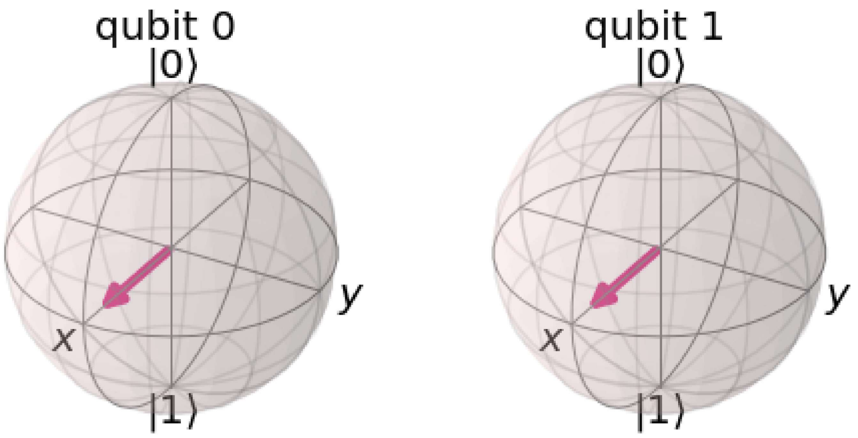
$$r_x \approx 0.86602, \quad r_y = 0, \quad r_z = 0$$

So, if now we use the `plot_bloch_multivector()` function in Qiskit to plot the composite state ρ_{CD}, we see we get two Bloch vectors that correctly represent each of these reduced density matrices ρ_C and ρ_D:

In [19]:

```
plot_bloch_multivector(rho_CD.data)
```

Out[19]:



Understanding the Bloch vector representation for multi-qubit states explains why, when we try to plot a two-qubit maximally entangled state in the Bloch sphere, we get an "empty" plot. Since the reduced density matrices of a state like:

$$|\psi_{AB}\rangle = \frac{1}{\sqrt{2}}(|0_A 0_B\rangle + |1_A 1_B\rangle)$$

are given by:

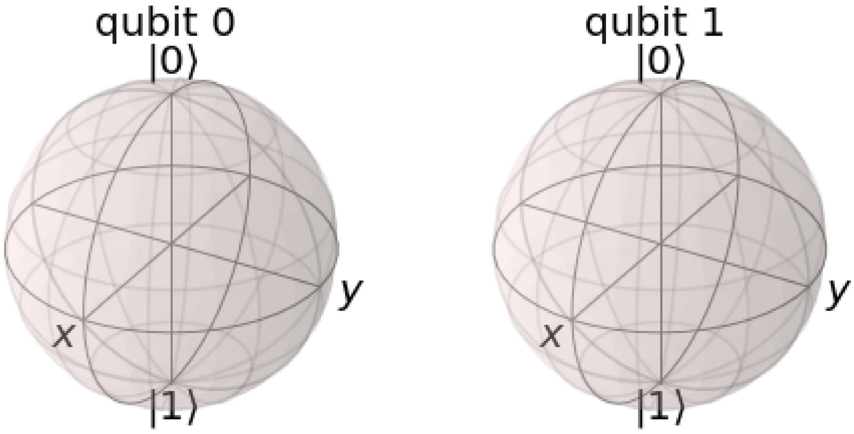
$$\rho_A = \rho_B = \frac{1}{2}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \frac{1}{2}\hat{I},$$

we see that the Bloch vector components r_x, r_y, r_z are all 0. Therefore, ρ_A and ρ_B actually have $\vec{r}$ vectors of zero length, represented by points at the origin of the sphere:

In [20]:

```
plot_bloch_multivector(rho_AB.data)
```

Out[20]:



In [ ]: