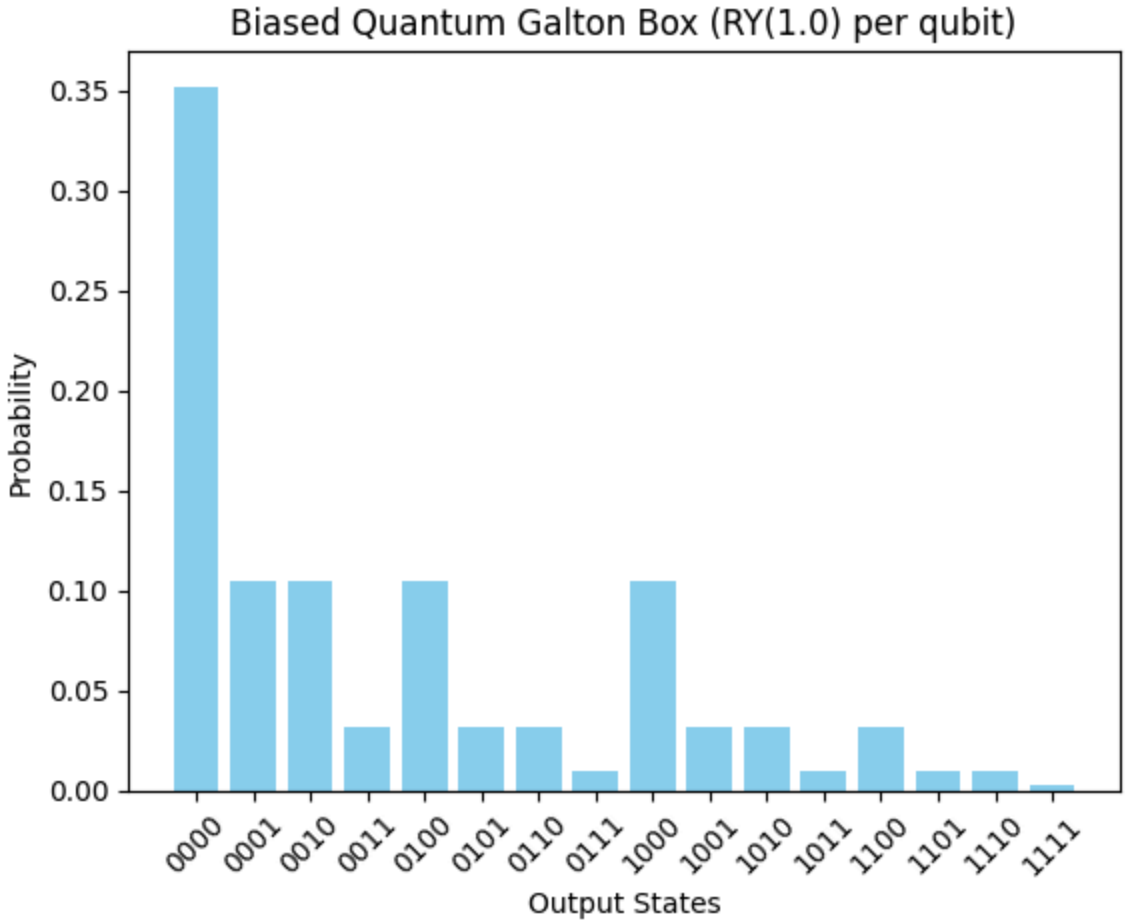
In [1]:

```
!pip install qiskit --quiet

from qiskit import QuantumCircuit
from qiskit.quantum_info import Statevector
from qiskit.visualization import plot_histogram
import matplotlib.pyplot as plt

# Number of qubits (levels)
n_qubits = 4

# Create circuit
qc = QuantumCircuit(n_qubits)

# Bias angle: π/2 = fair, other values = biased
theta = 1.0  # Try 0.5, 1.0, 2.0 for different biases

# Apply biased gates
for i in range(n_qubits):
    qc.ry(theta, i)

# Simulate
state = Statevector.from_instruction(qc)

# Convert probabilities to normal dict for plotting
probs_dict = {format(i, f'0{n_qubits}b'): p for i, p in enumerate(state.prob

# Plot
plt.bar(probs_dict.keys(), probs_dict.values(), color='skyblue')
plt.title(f"Biased Quantum Galton Box (RY({theta}) per qubit)")
plt.xlabel("Output States")
plt.ylabel("Probability")
plt.xticks(rotation=45)
plt.show()
```

```
[notice] A new release of pip is available: 25.1.1 -> 25.2
[notice] To update, run: pip install --upgrade pip
```


Biased Quantum Galton Box (RY(1.0) per qubit)
Probability
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
0000
0001
0010
0011
0100
0101
0110
0111
1000
1001
1010
1011
1100
1101
1110
1111
Output States

In [ ]: