Computing With Linear Algebra (pt. 2)

CSE 490Q: Quantum Computation
Probabilistic Computing

- States are **vectors** of non-negative numbers that sum to 1

- Operations are **stochastic matrices**
  - columns of non-negative numbers that sum to 1

- Dirac notation for linear algebra:
  - makes clear what is a vector, matrix, and number
  - eliminates the need for inventing some names

$|x\rangle \quad |x\rangle |x\rangle$  
$e_1, e_2$  
$|1\rangle, |2\rangle$
Combining Outcomes

Two bits: 00 01 10 11

One bit, die: 0 1 0 2 ··· 0 6 1 1 1 2 ··· 1 6

n outcomes \[\rightarrow\] \(m\) outcomes

\(nm\) outcomes
Two Bits

Two bits: 00 = 0  01 = 1  10 = 2  11 = 3

Ex: 0.2|00⟩ + 0.3|01⟩ + 0.3|10⟩ + 0.2|11⟩

N values, m outcomes (0,0) ↔ 100. 15°, 15°

basis vectors are ≈ \[ |\vec{1}⟩ \]
Combining States

First

\[ \left( \frac{1}{3} |0\rangle + \frac{2}{3} |1\rangle \right) \otimes \left( \frac{1}{2} |0\rangle + \frac{1}{2} |1\rangle \right) \]

Second

\[ \frac{1}{3} \langle 0 | \otimes \left( \frac{1}{2} |0\rangle + \frac{1}{2} |1\rangle \right) + \frac{2}{3} \langle 1 | \otimes \left( \frac{1}{2} |0\rangle + \frac{1}{2} |1\rangle \right) \]

\[ = \frac{1}{3} |00\rangle + \frac{1}{3} |01\rangle + \frac{2}{3} |10\rangle + \frac{2}{3} |11\rangle \]

\[ = \frac{1}{3} |00\rangle + \frac{1}{3} |01\rangle + \frac{1}{3} |10\rangle + \frac{1}{3} |11\rangle \]

\[ = \frac{1}{2} |00\rangle + \frac{1}{2} |10\rangle + \frac{1}{3} |11\rangle \]
Tensor Products

\( (a | x\rangle \otimes | y\rangle) = a (| x\rangle \otimes | y\rangle) \)

\( | x\rangle \otimes (a | y\rangle) = a (| x\rangle \otimes | y\rangle) \)

\( (| x\rangle + | y\rangle) \otimes | z\rangle = | x\rangle \otimes | z\rangle + | y\rangle \otimes | z\rangle \)

\( | z\rangle \otimes (| x\rangle + | y\rangle) = | z\rangle \otimes | x\rangle + | z\rangle \otimes | y\rangle \)

Notation: \( | x\rangle \otimes | y\rangle \rightarrow | x, y\rangle \).
Just saw \[ \begin{bmatrix} a \\ b \end{bmatrix}_{1 \times 2} \otimes \begin{bmatrix} c \\ d \end{bmatrix}_{2 \times 1} = \begin{bmatrix} ac \\ ad \\ bc \\ bd \end{bmatrix}_{2 \times 2} \]

\[
= \begin{bmatrix} a_1 y_1 \\ b_1 y_1 \\ a_2 y_2 \\ b_2 y_2 \end{bmatrix}_{4 \times 1}
\]
Tensor Products of Operations

\[
(F 1_x) \otimes (G 1_y) = (F \otimes G)(1_x \otimes 1_y)
\]
Kronecker Products of Operations

\[ F = \begin{bmatrix} a & b \\ c & d \end{bmatrix}, \quad G = \begin{bmatrix} e & f \\ g & h \end{bmatrix} \]

\[ F \otimes G = \begin{bmatrix} aG & bG \\ cG & dG \end{bmatrix} = \begin{bmatrix} ae & af & bc & bf \\ ag & ah & bg & bh \\ ce & ct & de & ef \\ cg & ch & dg & dh \end{bmatrix} \]
Example Operations

\[ X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \]

\[ F = \begin{bmatrix} 0 & 0 \\ 1 & 1 \end{bmatrix} \]

\[ (X \otimes F) |1,0\rangle = X |11\rangle \otimes F |10\rangle = 10 \langle \times 11 \rangle = 10 \langle 0 \times 11 \rangle = 10 \langle 1 \times 11 \rangle \]

\[
\begin{bmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
0 \\
1 \\
0 \\
1
\end{bmatrix}
= 
\begin{bmatrix}
0 \\
0 \\
1 \\
1
\end{bmatrix}
\rightarrow 
\begin{bmatrix}
00 \\
01 \\
10 \\
11
\end{bmatrix}
\]
Size of Probabilistic States

• An n-bit state requires $2^n$ numbers to describe
• An n-bit operation requires $2^n \times 2^n$ numbers to write down
• For even n = 1000, these are too large to write down!

• Does that mean that probabilistic computing is not possible?
• Certainly not!
Implementing Probabilistic Computing

- We cannot write down the full distribution on 1000 bits.

- To generate a random bit, we flip a fair coin and record its value as 0 or 1.
  - rather than the distribution on $2^n$ values, we just record one value.
  - that requires only $n$ bits.

- Deterministic operations take us from one specific value to another.
  - stochastic operations are a combination of coin flips and deterministic ops.
Sampling

- Output is not the distribution but a *sample* from that distribution.
- If we were to run it many times, we should find that each output arises the fraction of times indicated by the distribution.
- For decision problems (BPP), the output is just \{0, 1\} and we need only a 67% probability of sampling the correct answer.
Meaning of Probabilistic States

• Probabilistic states represent our knowledge of the system
  • state \([0.6, 0.4]^T\) means that 60% of these situations would give a 0

• However, the actual bit really is a 0 or 1!

• This will not be true for quantum states.
  • that is the crux of why quantum systems are hard to simulate
Sampling of Quantum States

• For quantum computation, the output will also be a random sample.

• The fact that we cannot write down the states of that computation does not rule out the possibility that we can generate the samples another way.
  • some probabilistic computation could produce the same distribution.

• For most problems of simulating quantum systems, we expect that is not possible.

• But in general, the burden of proof is on us to show that it cannot be done.
  • for some problems (e.g., factoring) the evidence is fairly strong.