Computing With Linear Algebra (Pt 1)

Quantum computing uses linear algebra to describe quantum states and operations.
- Use of linear algebra is not unique to QC, other forms of computation utilize linear algebra (Machine Learning for example).
- We will create this connection by showing how linear algebra is used to formulate randomness.
- The "weirdness" of QC comes from linear algebra, not from anything about Quantum Mechanics.
- Linear algebra notation differs in Physics.

A Random Coin

Consider coin C where $P(C=H) = \frac{1}{2}$, $P(C=T) = \frac{1}{2}$

These equations represent the State of the coin.

Consider a dice D for D $\rightarrow$ $\begin{bmatrix} P(D=1) = \frac{1}{6} \\ P(D=2) = \frac{1}{6} \end{bmatrix}$

We would like one compact item that describes what we know.
- We will do this by using vectors where each component represents a predefined probability.

For notation simplicity, let $\hat{e}_H \equiv \langle 0 \rangle$, $\hat{e}_T \equiv \langle 1 \rangle$.

\[
\begin{bmatrix}
\frac{1}{2} \\
\frac{1}{2}
\end{bmatrix} = \frac{1}{2} \hat{e}_H + \frac{1}{2} \hat{e}_T,
\]

where $\hat{e}_H$, $\hat{e}_T$ are eigenvectors $\langle 0 \rangle$, $\langle 1 \rangle$ respectively.

Dirac Notation

$|U\rangle \rightarrow$ to write a dot product of vectors $\hat{u}$, $\hat{v}$ $\rightarrow$ $\langle U | U \rangle$

$|U\rangle$ $\rightarrow$ bra

$\langle U |$ $\rightarrow$ column vector $[ ]$

$|U\rangle$ $\rightarrow$ row vector $[ ]$

bra $|U\rangle$ $\rightarrow$ row vector $[ ]$

$\langle U | W \rangle = \langle U | W \rangle = \hat{U}^\dagger \hat{W}$

Now with New notation, back to our coin $C \rightarrow \frac{1}{2} (|H\rangle + |T\rangle)$

In Dirac notation we can see have a vector $|1\rangle = |2\rangle$ where previously we cant.
States of a Random Bit

State: \( |\psi\rangle = a|0\rangle + b|1\rangle \) with \( a, b \geq 0 \) and \( a + b = 1 \) => \( |\psi\rangle \) Must be Normalized!

Operations on a Random Bit

If we would like to have a way to change from one state to another, in linear algebra we naturally use matrices to do this. In Quantum Mechanics we use Operators which act on states to change their State.

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

\[
|0\rangle \rightarrow 1 \\
|1\rangle \rightarrow 0
\]

\[
F_1 \equiv \begin{bmatrix}
  0 & 0 \\
  1 & 1 \\
\end{bmatrix}
\]

\[
F_1|0\rangle = 1 \\
F_1|1\rangle = 0
\]

Swap operator

\[
|\psi\rangle = 0.6|0\rangle + 0.4|1\rangle = \begin{bmatrix}
  0.6 \\
  0.4 \\
\end{bmatrix}
\]

\[
|\psi\rangle = 0.6|0\rangle + 0.4|1\rangle = \begin{bmatrix}
  0.6 \\
  0.4 \\
\end{bmatrix}
\]

Consider the following:

\[
G|1\rangle = |1\rangle \\
G|0\rangle = \frac{1}{2}(|0\rangle + |1\rangle)
\]

\[
G = \begin{bmatrix}
  1/2 & 0 \\
  0 & 1/2 \\
\end{bmatrix}
\]

For all matrices, valid operators? Or only some operators are?

- Not all are. Your final state after operation must have \( a, b \geq 0 \) and \( a + b = 1 \).

For an operator to be valid, it must take all eigenvectors to a new, normalized, valid state.

For arbitrary operator \( A \), \( A|0\rangle = a|0\rangle + b|1\rangle \). \( A \) is only a valid operator if and only if:

\[
\begin{bmatrix}
  a & c \\
  b & d \\
\end{bmatrix}
\]

\[
\begin{bmatrix}
  a & c \\
  b & d \\
\end{bmatrix} = \begin{bmatrix}
  a & 0 \\
  1 & 0 \\
\end{bmatrix}
\]

Any operator is a valid operator if the matrix associated with that matrix is a STOCHASTIC MATRIX.

Stochastic Matrices

\[
|\psi\rangle = s|0\rangle + t|1\rangle
\]

\[
A|\psi\rangle = sA|0\rangle + tA|1\rangle = s|0\rangle + t|1\rangle = (sa + tc)|0\rangle + (sb + td)|1\rangle
\]

* because \( a, b, c, d, s, t \geq 0 \) => sa + tc \geq 0, sb + td \geq 0

\[
sa + tc + sb + td = s(a + b) + t(c + d) = s + t = 1
\]

Any operator is a valid operator if the matrix associated with that matrix is a STOCHASTIC MATRIX.