Lecture 3: Symmetries, Heisenberg, dynamics, etc.

computational efficiency

~~You definitely should have noticed~~

You probably noticed last time that the computation time increased drastically with the number of sites, N.

Let's understand why.

For the N site system, the matrix for H has size

$$2^N \times 2^N$$

In general, to find the eigenvalues and eigenstates of an $M \times M$ matrix on the computer requires $C \cdot M^3$ computational steps, where C is some constant (that does not change with M).

We say the computation time is "$\mathcal{O}(M^3)$"

In other words, for N sites, the time to compute the eigenvalues and eigenvectors scales like $(2^N)^3 = 8^N$

<u>Exercise:</u> Check this with python. We can use the "time" module:

```
import time
start = time.time()
  # your code here
print(time.time() - start)
```

or store it:

```
times = np.zeros(len(Ns))
for N_idx, N in enumerate(Ns):
    start = time.time()
    times[N_idx] = time.time() - start
```

→ ~~then plt.scatter~~

then we can plot the calculation time vs N

If its really true that $T \approx C \cdot 8^N$

then $\log_2(T) = \log_2(C) + \log_2(2^{3N})$
$= \log_2(C) + 3N$

So a plot of $\log_2(T)$ vs N should have a slope of 3.

Go ahead and program this in the provided space in the .ipynb, and check that its true.

(Hint: mostly copy and paste code from last class.)

<u>Exercise 2</u>: Estimate the $\mathcal{O}(\quad)$ for ZZ_v2.

<u>ie</u> How does the number of calculations scale with N?

<u>Note</u>: you don't need to worry about how long each step takes, that's in "C"

<u>Answer</u>: For each column (2^N) we compute b, then apply ($N-1$) terms, for each one doing some calculation and storing the # in a matrix.

So its $\mathcal{O}(N \cdot 2^N)$

<u>Pt 2</u>: and ZZ (Kronecker method).

Answ: To generate $A \otimes B$ you do one multiplication for each element of the end matrix. So if A is $2^m \times 2^m$ and B is 2×2, do $(2^{m+1})^2$ multiplications.

So for $(N-1)$ terms we do $(2^2)^2 + (2^3)^2 + \ldots + (2^N)^2$ calculations.

and $\sum_{i=1}^{N} a^i = \frac{a-a^{N+1}}{1-a}$ so it is $(N-1)\cdot\frac{4-4^{N+1}}{1-4}$

$$=O(N\cdot 4^N)$$

4^N vs 2^N means that as N gets large, ZZ_v2 should be <u>much</u> faster.

<u>Moral</u>: v1 was simpler to understand but actually worse!

[If you think carefully, you can really improve the computational efficiency.] !!!

We now apply this principle by using <u>symmetries</u>.

<u>Idea</u>: If $\exists\, A$ s.t. $[H,A]=0$, then they can be simultaneously diagonalized, so ~~just look at~~ you can first find eigenspace of A (easy), then create H_{sub} ~~that~~ acts only on that space.

<u>ie</u> H cannot "mix" states that have different A eigenvalues, so we can look at only part of the matrix.

You actually know this well already.

eg $H=\begin{pmatrix}1&0&0&0\\0&2&0&0\\0&0&1&1\\0&0&1&1\end{pmatrix}$

Take 30 seconds, find eigenstates and eigenvalues.

If I ask you how to diagonalize a 4×4 matrix by hand, in general you can't do it. But here you can.

<u>Why?</u>

Well, you say, it's obvious that $\begin{pmatrix}1\\0\\0\\0\end{pmatrix}$ and $\begin{pmatrix}0\\1\\0\\0\end{pmatrix}$ are eigenvectors. So what's leftover are $\begin{pmatrix}0\\0\\a\\b\end{pmatrix}$ and $\begin{pmatrix}0\\0\\c\\d\end{pmatrix}$.

Find $a\ b\ c\ d$ by diagonalizing just the 2×2 block

$$\begin{pmatrix}1&1\\1&1\end{pmatrix} \to \begin{pmatrix}1/\sqrt{2}\\ \frac{1}{\sqrt{2}}\end{pmatrix},\ \begin{pmatrix}1/\sqrt{2}\\ -\frac{1}{\sqrt{2}}\end{pmatrix}.$$

This is exactly the idea. These blocks are separated by being in <u>different symmetry sectors</u>.

The Ising model actually has this property. Let's see that.

We'll try to see which states for $N=2$ are "connected" through H.

"Connected": $H = \begin{pmatrix}a&1&\\1&b&1\\&1&c\end{pmatrix}$ $\quad H\begin{pmatrix}1\\0\\0\end{pmatrix}$ has $\begin{pmatrix}1\\0\\0\end{pmatrix}$ and $\begin{pmatrix}0\\1\\0\end{pmatrix}$ parts so these are connected through H.

But $H^2\begin{pmatrix}1\\0\\0\end{pmatrix}$ has $\begin{pmatrix}0\\0\\1\end{pmatrix}$ part too, so all three <u>can be mixed</u> in eigenstates.

or $H = \begin{pmatrix}a&1&\\1&b&\\&&c\end{pmatrix}$

Here $H^N\begin{pmatrix}0\\0\\1\end{pmatrix} \propto \begin{pmatrix}0\\0\\1\end{pmatrix}\ \forall N \Rightarrow$ this is not <u>connected</u> to the other two, it will never be mixed in "eigenstates".

So given any H matrix, we can check for these connected components by looking at $\rightarrow M \times M$

$H, H^2, H^3, \dots, H^M$ and seeing if there are persistent 0s indicating disconnected parts.

(To avoid accidental zeros, can replace all nonzeros in H by 1)

For Ising model (at $g \neq 0$), $H \sim \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{pmatrix}$

$H^2 \sim \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}$

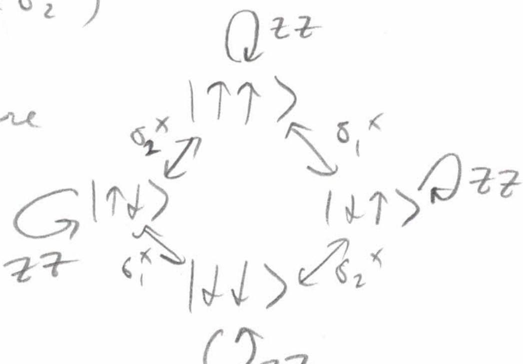
so it doesn't look like there are disconnected parts.

<u>But there are!!</u> They are hidden.

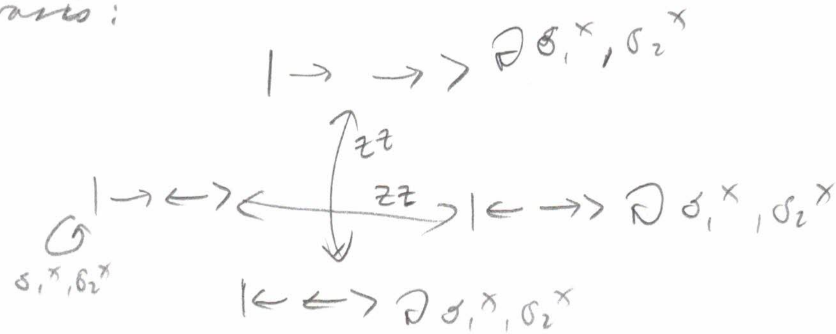
Look back at the model ($N=2$):

$$H = -\sigma_1^z \sigma_2^z - g(\sigma_1^x + \sigma_2^x)$$

In the z basis, states are



<u>But</u> in the X basis:



So let's write H in the X basis.

3 ways to do it: ① Change of basis.

$H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ does one particle CoB.

$\underbrace{H \otimes H \otimes H \otimes \cdots \otimes H}_{N}$ does it for N particles

② ZZ_v2-type method, but ZZ and X now act very differently

③ Write Z basis matrix for

$$H = -\sum_i \sigma_i^x \sigma_{i+1}^x - g \sum_i \sigma_i^z$$

As you can guess from the analysis of time to do the ZZ_v1 vs ZZ_v2, $\boxed{\text{Will do ②}}$ ← ∀N

~~① is not actually good, but we'll use that now, since its the fastest to program.~~ ← ~~$\boxed{\text{Exercise}}$~~

☆ [Exercises for home later: do ① and ③, show they match ②]

Ans for N=2

for $\rightarrow\rightarrow = 0$

$\rightarrow\leftarrow = 1$

$\leftarrow\rightarrow = 2$

$\leftarrow\leftarrow = 3$

basis:

$$H = \begin{pmatrix} -2g & & & -1 \\ & 0 & 1 & \\ & 1 & 0 & \\ -1 & & & 2g \end{pmatrix}$$

You can see that 0, 3 are connected, 1, 2, are, but in general the states are not all connected.

Exercise: will do 2 because it will be useful in a few minutes.

X is now the diagonal one:

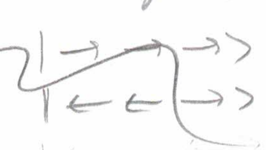
```
def X_X_basis(N):
    X_diag = np.zeros(2**N)
    for i in range(2**N):
        b = d2b(i, N)
        val = 0
        for term in range(N):
            val += (-1)**b[term]
        X_diag[i] = val
    return (np.diag(X_diag))
```

ZZ is not diagonal, rather it flips 2 spins.

```
def ZZ_X_basis(N):
    ZZ = np.zeros((2**N, 2**N))
    for col in range(2**N):   # columns
        b = d2b(col, N)
        for term in range(N-1):
            new_b = b.copy()
            new_b[term] = 1 - new_b[term]
            new_b[term+1] = 1 - new_b[term+1]
            row = b2d(new_b)
            ZZ[row, col] += 1
    return ZZ
```

For $N>2$: It will still produce 2 connected parts of the space.

eg $N=3$



eg $N=3$

$$|\rightarrow\rightarrow\rightarrow\rangle \xrightarrow{z_0 z_1} |\leftarrow\leftarrow\rightarrow\rangle \xrightarrow{z_1 z_2} |\leftarrow\rightarrow\leftarrow\rangle$$

$$|\rightarrow\rightarrow\rightarrow\rangle \xrightarrow{z_1 z_2} |\rightarrow\leftarrow\leftarrow\rangle \xrightarrow{z_0 z_1} |\leftarrow\rightarrow\leftarrow\rangle$$

$$|\leftarrow\leftarrow\leftarrow\rangle \xrightarrow{z_0 z_1} |\rightarrow\rightarrow\leftarrow\rangle \xrightarrow{z_1 z_2} |\rightarrow\leftarrow\rightarrow\rangle$$

$$|\leftarrow\leftarrow\leftarrow\rangle \xrightarrow{z_1 z_2} |\leftarrow\rightarrow\rightarrow\rangle \xrightarrow{z_0 z_1} |\rightarrow\leftarrow\rightarrow\rangle$$

same component $\Updownarrow$ total S_x differ by 4

Total S_x is $3, -1, -1, -1$

Total S_x is $-3, 1, 1, 1$

$N=4$ groups

Total S_x	Total S_x
$4, 0, -4$	$2, -2$
1 state, $\binom{4}{2}$ states $= 6$, 1 state	4 states
total: 8 states	8 states

Conclusion: we can reduce size of H to diagonalize by a factor of 2 (for any N)

$\rightarrow$ very helpful, but doesn't improve exponential scaling.

There is a different model for which this will be much more helpful.

$$H = \sum_i J_z \sigma^z_i \sigma^z_{i+1} + J\left(\sigma_i^x \sigma_{i+1}^x + \sigma_i^y \sigma_{i+1}^y\right)$$

The <u>XXZ model</u>

Called this b/c coeffs of $\sigma^x\sigma^x$ and $\sigma^y\sigma^y$ are equal (XX) and for $\sigma^z\sigma^z$ is different (Z)

This also has 1 effective parameter, $J_z/J \equiv \Delta$.

$$H/J = \sum_i \Delta \sigma_i^z \sigma_{i+1}^z + \left(\sigma_i^x \sigma_{i+1}^x + \sigma_i^y \sigma_{i+1}^y\right)$$

$\boxed{\text{Exercise}}$ Let's go ahead and construct the matrix for $N=2$

$\boxed{\text{Answer}}$

$$\Delta \cdot \begin{pmatrix} 1 & & & \\ & -1 & & \\ & & -1 & \\ & & & 1 \end{pmatrix} + \begin{pmatrix} & & & 1 \\ & & 1 & \\ & 1 & & \\ 1 & & & \end{pmatrix} + \begin{pmatrix} & & & -1 \\ & & 1 & \\ & 1 & & \\ -1 & & & \end{pmatrix}$$

$$= \begin{pmatrix} \Delta & & & \\ & -\Delta & 2 & \\ & 2 & -\Delta & \\ & & & \Delta \end{pmatrix}$$

This should feel familiar. Remember, on day 1, we found $(\vec{S}_1+\vec{S}_2)^2 \equiv \vec{S}_{tot}^{\,2}$ which $= 3\cdot \mathrm{Id} + \vec{S}_1\cdot\vec{S}_2$

is basically $\vec{S}_1\cdot\vec{S}_2$ but w/ $\Delta \neq 1$

Anyway, you can already see that ~~the~~ separately dealing w/ non-connected parts of H will be more beneficial here, because there are 3 <u>blocks</u>, vs 2 for N=2 Ising.

Let's try to understand where the blocks come from.

$$\begin{array}{c} \begin{pmatrix} \boxed{\Delta} & & & \\ & -\Delta & 2 & \\ & 2 & -\Delta & \\ & & & \boxed{\Delta} \end{pmatrix} \end{array} \begin{array}{l} \uparrow\uparrow \\ \uparrow\downarrow \\ \downarrow\uparrow \\ \downarrow\downarrow \end{array}$$

what do you notice that is the same here, but different from $\uparrow\uparrow$, $\downarrow\downarrow$?

[Solicit suggestions] ~ 30s

$\boxed{\text{Answer}}$ $\uparrow\uparrow \to$ total $S_z = 2$

$\uparrow\downarrow, \downarrow\uparrow \longrightarrow$ total $S_z = 0$

$\downarrow\downarrow \to$ total $S_z = -2$

How can we understand this?

$\sigma_i^x \sigma_{i+1}^x$ & $\sigma_i^y \sigma_{i+1}^y$ can be rewritten in terms of

σ^+ and σ^-

σ^+ is defined by $\sigma^+|\downarrow\rangle = |\uparrow\rangle$, $\sigma^+|\uparrow\rangle = 0$, $\sigma^- = (\sigma^+)^\dagger$; $\sigma^-|\uparrow\rangle = |\downarrow\rangle$, $\sigma^-|\downarrow\rangle = 0$

As a result, $\sigma^x = \sigma^+ + \sigma^-$

$\sigma^y = -i\sigma^+ + i\sigma^-$

$\boxed{\text{Exercise}}$ Show that $\sigma_i^x \sigma_{i+1}^x + \sigma_i^y \sigma_{i+1}^y = 2(\sigma_i^+ \sigma_{i+1}^- + \sigma_i^- \sigma_{i+1}^+)$

(Use any method you like)

Answer

$$(\sigma_i^+ + \sigma_i^-)(\sigma_{i+1}^+ + \sigma_{i+1}^-) + (-i\sigma_i^+ + i\sigma_i^-)(-i\sigma_{i+1}^+ + i\sigma_{i+1}^-)$$

$$= \sigma_i^+\sigma_{i+1}^- + \sigma_i^-\sigma_{i+1}^+ + \sigma_i^+\sigma_{i+1}^+ + \sigma_i^-\sigma_{i+1}^-$$

$$+\left(\;'' \;+\; '' \;-\; '' \;-\; ''\;\right)$$

$$= 2\left(\sigma_i^+\sigma_{i+1}^- + \sigma_i^-\sigma_{i+1}^+\right)$$

Other methods:
- show both sides of = act the same on all 4 basis states
- write the matrices for σ^+, σ^-, do Kronecker product etc, show matrices equal for both sides.

Result:

$$\frac{H}{J} = \sum_i \Delta\, \sigma_i^z\sigma_{i+1}^z + 2\left(\sigma_i^+\sigma_{i+1}^- + \sigma_i^-\sigma_{i+1}^+\right)$$

no effect on z basis states

maintains total z by flipping one ↑→↓ and one ↓→↑

Conclusion: H conserves total σ_z

$$\text{ie}\quad \left[\sum_i \sigma_i^z , H\right] = 0$$

How do we use this? It means that if two basis states have different $\sum_i \sigma_i^z$, then $\langle \varphi_1 | H | \varphi_2 \rangle = 0$.

→ so we don't need to find these at all. Instead, just look at groups of states with the same total σ_z

For example, let $N=4$. What are the blocks we can get?

Exercise: Find all the different symmetry blocks. Which states are in each one? Confirm the total # of states is 16.

Exercise 2: Construct the restricted H ~~that~~ acts on each block.

Exercise 3: In general, if N is even, what is the ~~total~~ σ^z of the largest block? ~~How~~ big is it, as a function of N?

Answers: ① 4: $\uparrow\uparrow\uparrow\uparrow$

2: $\uparrow\uparrow\uparrow\downarrow$, $\uparrow\uparrow\downarrow\uparrow$, $\uparrow\downarrow\uparrow\uparrow$, $\downarrow\uparrow\uparrow\uparrow$

0: $\uparrow\uparrow\downarrow\downarrow$, $\uparrow\downarrow\uparrow\downarrow$, $\uparrow\downarrow\downarrow\uparrow$, $\downarrow\uparrow\uparrow\downarrow$, $\downarrow\uparrow\downarrow\uparrow$, $\downarrow\downarrow\uparrow\uparrow$

-2: $\downarrow\downarrow\downarrow\uparrow$, $\downarrow\downarrow\uparrow\downarrow$, $\downarrow\uparrow\downarrow\downarrow$, $\uparrow\downarrow\downarrow\downarrow$

-4: $\downarrow\downarrow\downarrow\downarrow$

#: $\binom{4}{4} + \binom{4}{3} + \binom{4}{2} + \binom{4}{1} + \binom{4}{0} = 16$

$$\left[2^n = (1+1)^n = \sum_k 1^k 1^{n-k} \binom{n}{k} \right]$$

② 4: $[3\Delta]$

2: $ZZ \begin{bmatrix} \Delta & & & \\ & -\Delta & & \\ & & -\Delta & \\ & & & \Delta \end{bmatrix}$, $\underset{(+-)+(-+)}{XX+YY} \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$

$$\begin{bmatrix} \Delta & 1 & 0 & 0 \\ 1 & -\Delta & 1 & 0 \\ 0 & 1 & -\Delta & 1 \\ 0 & 0 & 1 & \Delta \end{bmatrix}$$

3.7

0: ZZ
$$\begin{bmatrix} \Delta &&&&& \\ & -3\Delta &&&& \\ && -\Delta &&& \\ &&& -\Delta && \\ &&&& -3\Delta & \\ &&&&& \Delta \end{bmatrix}$$

$(+-)+(-+)$
$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

③ $\binom{N}{N} + \binom{N}{N-1} \cdots \binom{N}{0}$

$1 \quad N \quad \frac{N(N-1)}{2} \quad \cdots \quad N \quad 1$

$\binom{N}{N/2}$

$\binom{N}{N/2} \times \binom{N}{N/2}$

of the whole H: $2^N \times 2^N$

How much benefit do we get?

Use Stirling's formula

$$\binom{N}{N/2} = \frac{N!}{\left(\frac{N}{2}\right)!\left(\frac{N}{2}\right)!}$$

Stirling $\ln(N!) \approx N\ln(N) - N$

$$\ln\left(\binom{N}{N/2}\right) = \ln(N!) - 2\ln\left(\tfrac{N}{2}!\right)$$

$$\approx N\ln(N) - \cancel{N} - 2\left(\tfrac{N}{2}\ln\left(\tfrac{N}{2}\right) - \cancel{\tfrac{N}{2}}\right)$$

$$= N\left(\ln(N) - \ln\left(\tfrac{N}{2}\right)\right)$$

$$= N\left(\ln(N) - \ln(N) + \ln(2)\right)$$

$$= N\ln(2)$$

$$= \ln(2^N)$$

$$\Rightarrow \binom{N}{\frac{N}{2}} \sim 2^N$$

Let's be more precise:

$$\ln(N!) \simeq N\ln(N) - N + \tfrac{1}{2}\ln(2\pi N)$$

So $\ln\left(\binom{N}{N/2}\right) \simeq \ln(2^N) + \frac{1}{2}\ln(2\pi N) - 2\cdot\frac{1}{2}\ln\left(2\pi\frac{N}{2}\right)$

$$\binom{N}{\frac{N}{2}} \simeq 2^N \cdot \frac{\sqrt{2\pi N}}{2\pi N/2} \approx \frac{2^N}{\sqrt{2\pi N}/2}$$

So the reduction in the size of H is by a factor

$$\sim \sqrt{N}$$

This is somewhat better than for Ising (factor of 2), but still a lot of improvement needed. Want a method that does not scale exponentially.