計算科学における情報圧縮

Information Compression in Computational Science

2021.12.16

#10:高度なデータ圧縮：情報のエンタングルメントと行列積表現

Entanglement of information and matrix product states

理学系研究科　物理学専攻　大久保　毅

Department of Physics, **Tsuyoshi Okubo**

Schedule

1st: Huge data in modern physics (Today)

2nd: Information compression in modern physics
(+review of linear algebra)

3rd: Review of linear algebra (+ singular value decomposition)

4th: Singular value decomposition and low rank approximation

5th: Basics of sparse modeling

6th: Basics of Krylov subspace methods

7th: Information compression in materials science

8th: Accelerating data analysis: Application of sparse modeling

9th: Data compression: Application of Krylov subspace method

10th: Entanglement of information and matrix product states

11th: Matrix product states + Application of MPS to eigenvalue problems

12th: Application of MPS to time evolution and data science

13th: Other tensor network representations
+ (Appendix: Information compression by tensor network renormalization)

Outline

- Outline of tensor network decomposition
- Entanglement
  - Schmidt decomposition
  - Entanglement entropy and its area law
- Matrix product states
  - Matrix product states (MPS)
  - ~~Canonical form~~
  - ~~infinite MPS~~

Outline of tensor network decomposition

Classification of Information Compression by Memory Costs

Linear algebra for huge data: $\vec{v} \in \mathbb{C}^M$

(1) A matrix can be stored

Required memory~ $O(M^2)$

(2) Although a matrix cannot be stored, vectors can be stored

Required memory~ $O(M)$

(3) A vector cannot be stored

$\boxed{\text{Required memory} \ll O(M)}$

We try to approximate a vector in a compact form.

$M \sim a^N$ ⇒ Memory ~ $O(N^x)$

Exponential **Polynomial**

N :problem size (e.g. system size)

When we efficiently compress a vector?

$$\vec{v} = \sum_{i=1}^{M} C_i \vec{e}_i \qquad \vec{v} \in \mathbb{C}^M$$

If we can find a basis where the coefficients have a structure (correlation).

(1) Almost all C_i are zero (or very small).

We store only a few finite elements $\{(i, C_i)\}$

E.g.

Fourier transformation $\vec{v} = \sum_{k=1}^{M} D_k \vec{f}_k$

If we can neglect larger wave numbers, we can efficiently approximate the vector with smaller number of coefficients.

Classical state $|\Psi\rangle = |01011\ldots00\rangle$

In this case, we know that only a specific C_i is non-zero.

We need only an integer corresponding to the non-zero element.

When we efficiently compress a vector?

$$\vec{v} = \sum_{i=1}^{M} C_i \vec{e}_i \qquad \vec{v} \in \mathbb{C}^M$$

(2) All of C_i are not necessarily independent.

We store **"structure"** and **"independent elements"**.

$$\{(i, C_i)\}$$

E.g. Product state ("generalized" classical state)

A vector is decomposed into product of small vectors.

$$|\Psi\rangle = |\phi_1\rangle \otimes |\phi_2\rangle \otimes \cdots$$

e.g.

$$|\phi_1\rangle = \alpha|0\rangle + \beta|1\rangle$$

$$|\phi_1\rangle = |01\rangle - |10\rangle$$

(It is identical to the rank-1 CP decomposition.)

structure: **"product state"**

independent elements: **small vectors**

Tensor network decomposition of a vector

Target:

Exponentially large Hilbert space

$\vec{v} \in \mathbb{C}^M$

with $M \sim a^N$

+

Total Hilbert space is decomposed as a product of "local" Hilbert space.

$$\mathbb{C}^M = \mathbb{C}^a \otimes \mathbb{C}^a \otimes \cdots \mathbb{C}^a$$

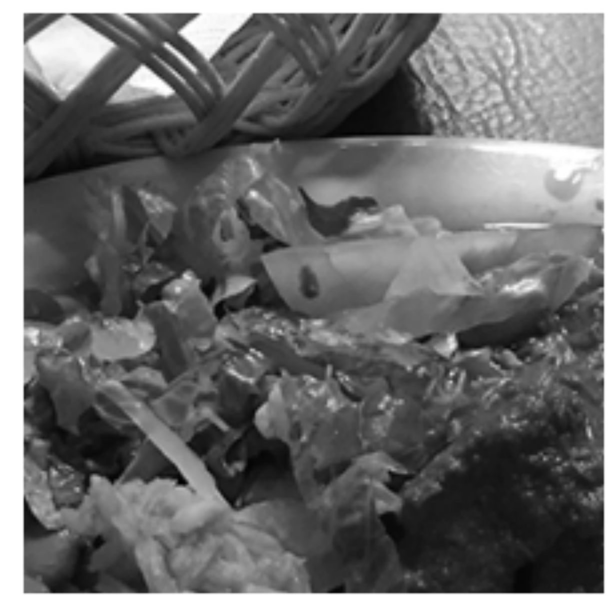
*Local Hilbert space dimensions can be different.

Examples:

Picture image:

256 × *256* pixel image → 2^{16} dimensional vector

→ *16*-leg tensor (with *a* = *2*)



Probability distribution:

e.g. Ising model $P(\{S_i\}) = \dfrac{e^{\beta J \sum_{\langle i,j \rangle} S_i S_j}}{Z}$

→ 2^N vector → *N*-leg tensor (with *a* = *2*)

Wave function:

$$|\Psi\rangle = \sum_{\{m_i=0,1\}} T_{m_1,m_2,\cdots,m_N} |m_1, m_2, \cdots, m_N\rangle \quad \rightarrow \quad T_{m_1,m_2,\cdots,m_N}$$:*N-leg tensor*

Tensor network decomposition of a vector

Target:

Exponentially large Hilbert space

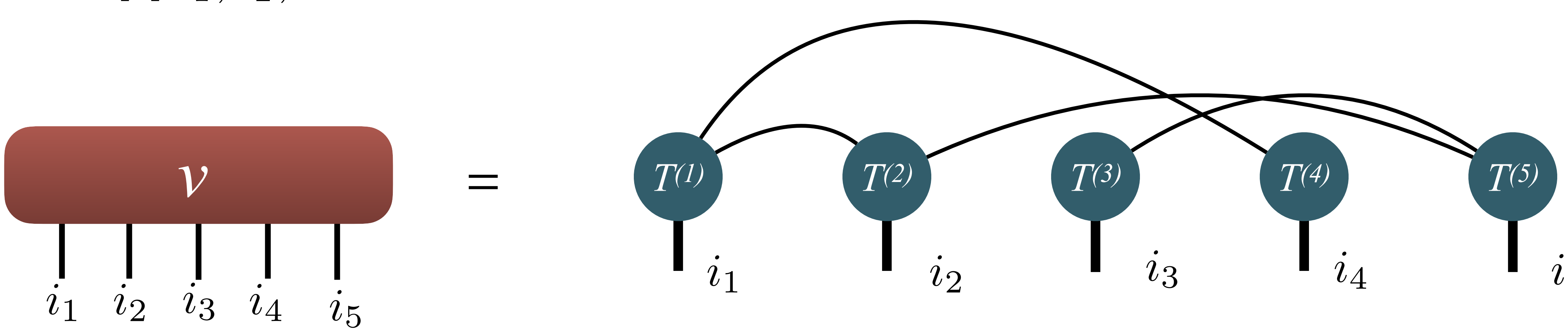
$\vec{v} \in \mathbb{C}^M \qquad \mathbb{C}^M = \mathbb{C}^a \otimes \mathbb{C}^a \otimes \cdots \mathbb{C}^a$

Tensor network decomposition

$$v_i = v_{i_1,i_2,\ldots,i_N} = \sum_{\{x\}} T^{(1)}[i_1]_{x_1,x_2,\ldots} T^{(2)}[i_2]_{x_1,x_3,\ldots} \cdots T^{(N)}[i_N]_{x_3,x_{100},\ldots}$$

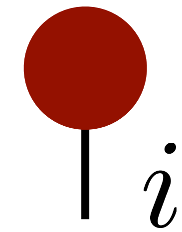
$i_n = 0, 1, \ldots, a-1$: index of local Hilbert space

$T[i]_{x_1,x_2,\ldots}$: local tensor for "state" i

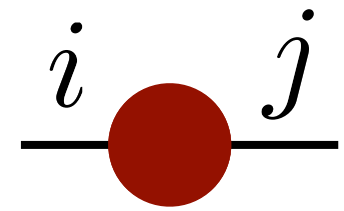


Graphical representations for tensor network

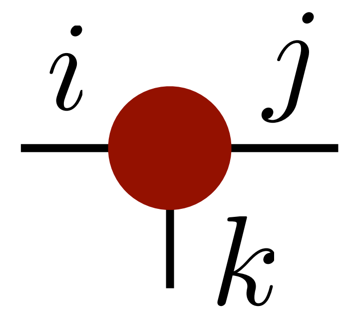
- Vector $\vec{v}$: v_i



- Matrix M : $M_{i,j}$

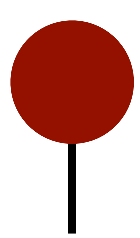
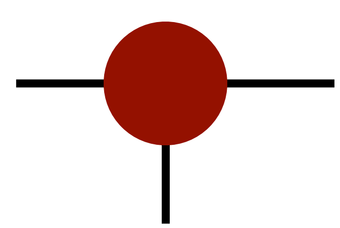


- Tensor T : $T_{i,j,k}$



*** n-rank tensor = n-leg object**

When indices are not presented in a graph, it represent a tensor itself.

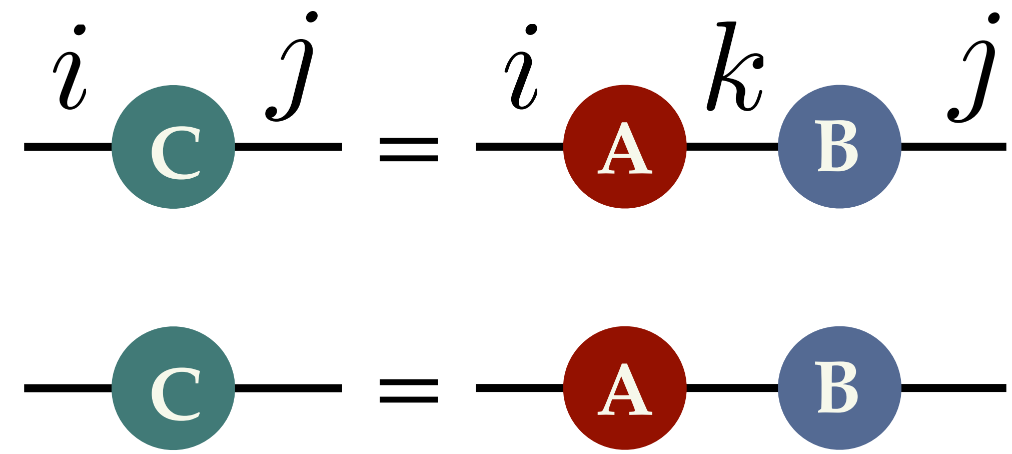
$\vec{v} =$  $T =$ 

Graphical representations for tensor network

Matrix product

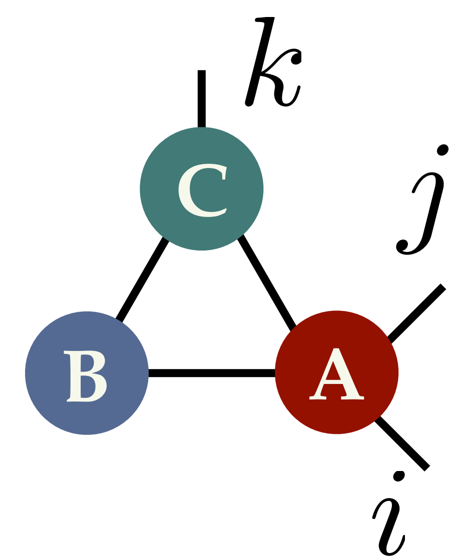
$$C_{i,j} = (AB)_{i,j} = \sum_k A_{i,k} B_{k,j}$$

$$C = AB$$



Generalization to tensors

$$\sum_{\alpha,\beta,\gamma} A_{i,j,\alpha,\beta} B_{\beta,\gamma} C_{\gamma,k,\alpha}$$

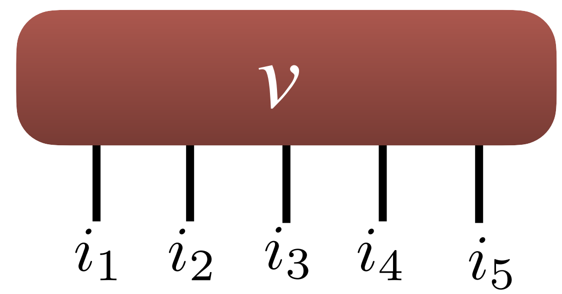


Contraction of a network = **Calculation of a lot of multiplications**

（縮約）

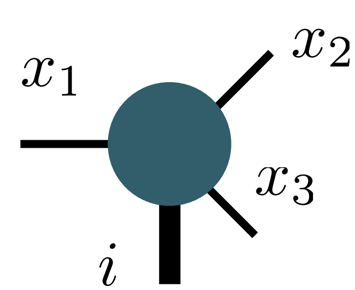
Diagram for a tensor network decomposition

- Vector v_{i_1,i_2,i_3,i_4,i_5}



*Vector looks like a tensor

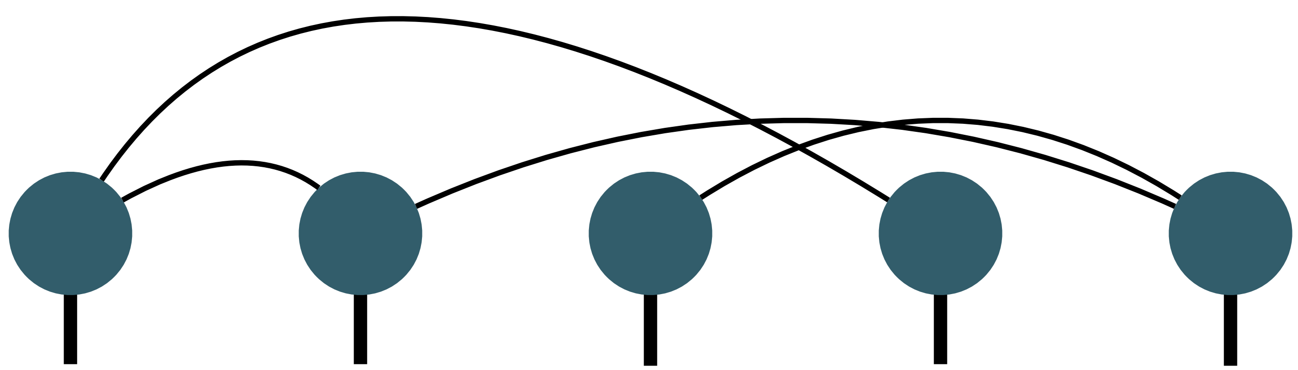
- Tensor $T[i]_{x_1,x_2,x_3}$



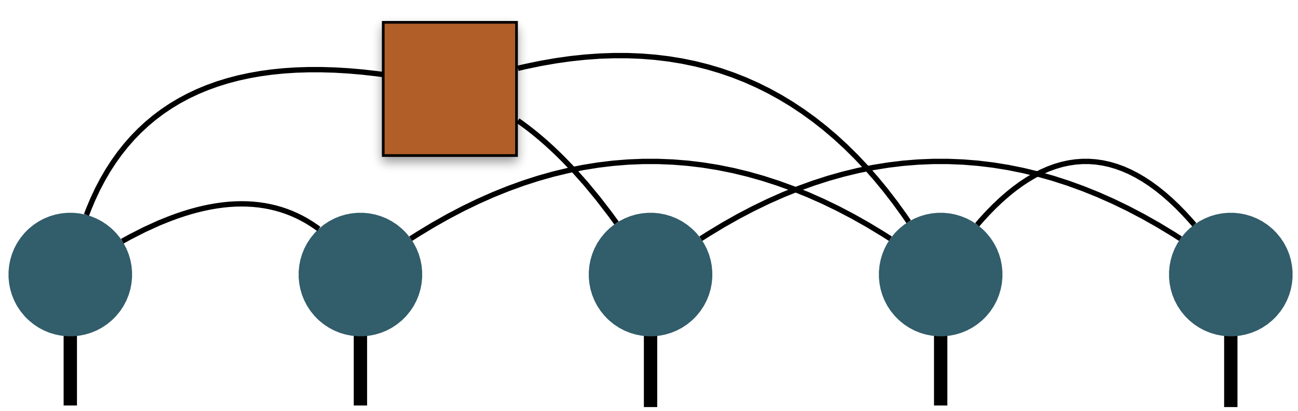
*We treat i as an index of the tensor.

Tensor network decomposition

$\vec{v} =$



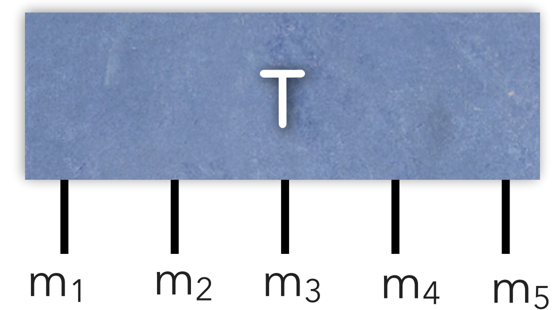
$\vec{w} =$



*We can consider tensors independent on i.

Another "generalization" of SVD to tensors.

$T_{m_1,m_2,\cdots,m_N}$:N-leg tensor (or Vector)

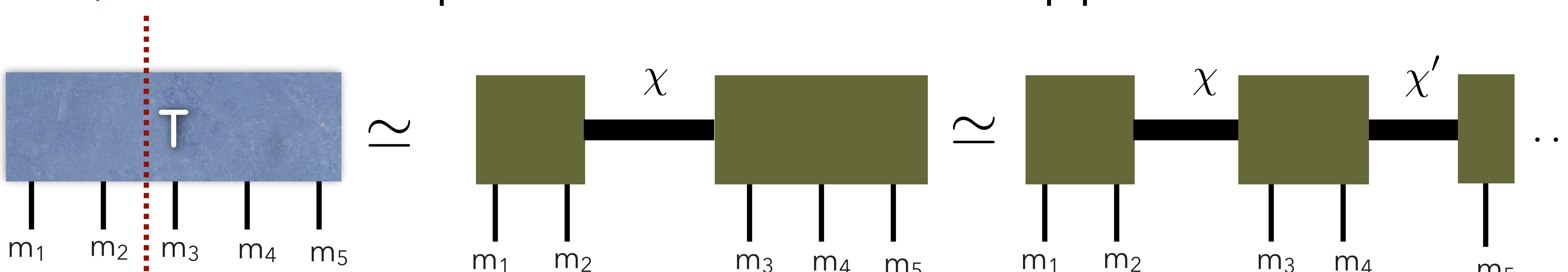


Cf. wave function: $|\Psi\rangle = \sum_{\{m_i=0,1\}} T_{m_1,m_2,\cdots,m_N}|m_1,m_2,\cdots,m_N\rangle$

We can consider it as a matrix by making two groups:

$$T_{\{m_1,m_2,\cdots,m_M\},\{m_{M+1},\cdots,m_N\}}$$

We can perform the low rank approximation of T.



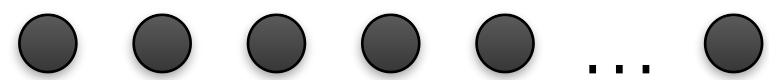
*obtained two objects are again tensors.

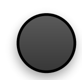
What does it mean? It is related to MPS

Entanglement（エンタングルメント）

N-qubit system (S=1/2 quantum spin system)

Example vector: Wave function of N-qubit systems



 takes two states $|0\rangle, |1\rangle$ $(|\uparrow\rangle, |\downarrow\rangle)$

$$
\begin{aligned}
|\Psi\rangle &= \sum_{\{i_1, i_2, \ldots i_N\}} \Psi_{i_1 i_2 \ldots i_N} |i_1\rangle \otimes |i_2\rangle \otimes \cdots \otimes |i_N\rangle \\
&= \sum_{\{i_1, i_2, \ldots i_N\}} \Psi_{i_1 i_2 \ldots i_N} |i_1 i_2 \ldots i_N\rangle
\end{aligned}
$$

Coefficients = vector: $\vec{\Psi} \in \mathbb{C}^{2^N}$

* Inner product: $\langle \Phi | \Psi \rangle = \vec{\Phi}^* \cdot \vec{\Psi}$

Schmidt decomposition

General vector: $\vec{x} \in \mathbb{V}_1 \otimes \mathbb{V}_2$ $\quad$ $\dim\, \mathbb{V}_1 = n_1, \dim\, \mathbb{V}_2 = n_2$

$(n_1 \geq n_2)$

Schmidt decomposition

There exists special basis which satisfies

$$\vec{x} = \sum_{i=1}^{n_2} \lambda_i \vec{u}_i \otimes \vec{v}_i$$

No off-diagonal coupling!

Orthonormal basis $\{\vec{u}_1, \vec{u}_2, \ldots, \vec{u}_{n_1}\} \in \mathbb{V}_1$

$\{\vec{v}_1, \vec{v}_2, \ldots, \vec{v}_{n_2}\} \in \mathbb{V}_2$

Schmidt coefficient $\lambda_i \geq 0$

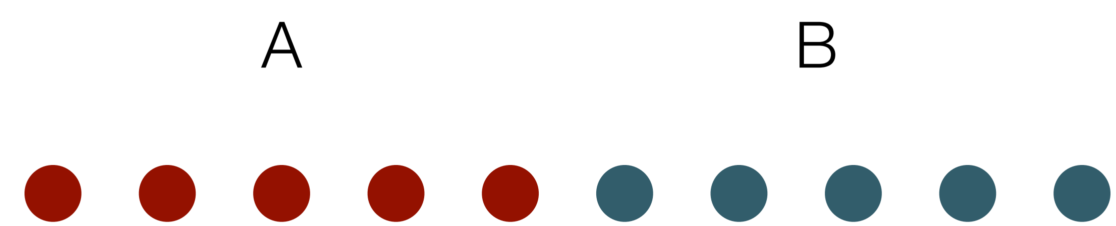
Schmidt decomposition is unique.

Schmidt decomposition for a wave function

Wave function: $|\Psi\rangle = \sum_{\{i_1,i_2,\dots i_N\}} \Psi_{i_1 i_2 \dots i_N} |i_1 i_2 \dots i_N\rangle$

Schmidt decomposition

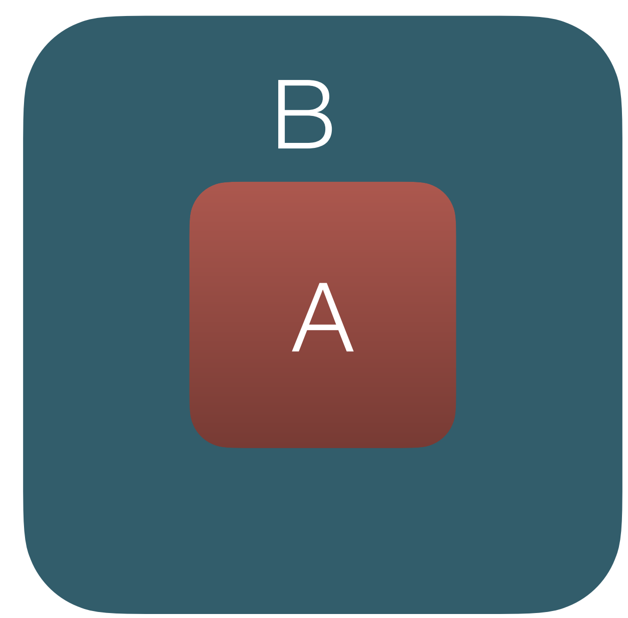
Divide a system into two parts, A and B:



General wave function can be represented by a superposition of orthonormal basis set.

$$|\Psi\rangle = \sum_{i,j} M_{i,j} |A_i\rangle \otimes |B_j\rangle = \sum_i \lambda_i |\alpha_i\rangle \otimes |\beta_i\rangle$$

$$\boxed{M_{i,j} \equiv \Psi_{\underbrace{(i_1,\dots)}_{A},\underbrace{(\dots,i_N)}_{B}} \qquad \begin{aligned} |A_i\rangle &= |i_1, i_2, \dots\rangle \\ |B_j\rangle &= |\dots, i_{N-1}, i_N\rangle \end{aligned}}$$



Orthonormal basis: $\langle A_i | A_j \rangle = \langle B_i | B_j \rangle = \delta_{i,j}$, $\langle \alpha_i | \alpha_j \rangle = \langle \beta_i | \beta_j \rangle = \delta_{i,j}$

Schmidt coefficient: $\lambda_i \geq 0$

Relation between SVD and Schmidt decomposition

Singular value decomposition (SVD):

For a K × L matrix M,

$$M_{i,j} = \sum_m U_{i,m} \lambda_m V^\dagger_{m,j}$$

Singular values: $\lambda_m \geq 0$

Singular vectors:
$$\sum_m U_{i,m} U^\dagger_{m,j} = \delta_{i,j}$$
$$\sum_m V_{i,m} V^\dagger_{m,j} = \delta_{i,j}$$

Relation to the Schmidt decomposition:

$$|\Psi\rangle = \sum_{i,j} M_{i,j} |A_i\rangle \otimes |B_j\rangle = \sum_m \lambda_m |\alpha_m\rangle \otimes |\beta_m\rangle$$

$$|\alpha_m\rangle = \sum_i U_{i,m} |A_i\rangle$$

$$|\beta_m\rangle = \sum_j V^\dagger_{m,j} |B_j\rangle$$

$$\Rightarrow \quad \langle \alpha_i | \alpha_j \rangle = \langle \beta_i | \beta_j \rangle = \delta_{i,j}$$

$\boxed{\textbf{By using SVD, we can perform Schmidt decomposition.}}$

Partial trace and reduced density matrix

For $\vec{x} \in \mathbb{V}_1 \otimes \mathbb{V}_2$ $\quad \dim\, \mathbb{V}_1 = n_1, \dim\, \mathbb{V}_2 = n_2 \quad |\vec{x}| = 1$

Density matrix: $\rho \equiv \vec{x}\vec{x}^{\dagger}$ $\quad (\rho_{ij} = x_i x_j^*)$

（密度行列） $(\rho = |x\rangle\langle x|)$ $\qquad$ *Note: $\mathrm{rank}\ \rho = 1$

Orthonormal basis: $\{\vec{e}_1, \vec{e}_2, \ldots, \vec{e}_{n_1}\} \in \mathbb{V}_1 \quad \{\vec{f}_1, \vec{f}_2, \ldots, \vec{f}_{n_2}\} \in \mathbb{V}_2$

Basis for $\vec{x}$: $\vec{g}_{i_1,i_2} = \vec{e}_{i_1} \otimes \vec{f}_{i_2}$

Index: $i = (i_1, i_2)$

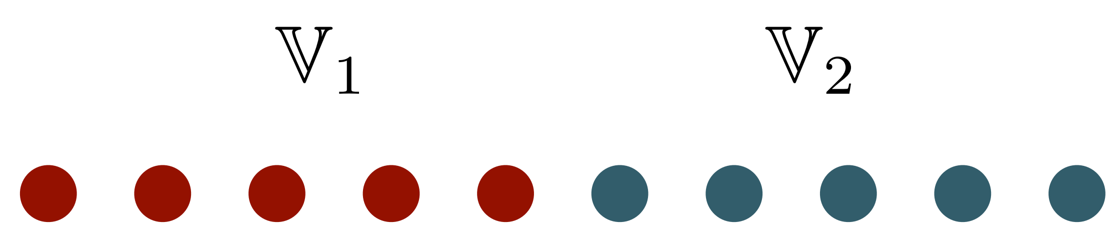
Reduced Density matrix:

（縮約密度行列）

$\rho_{\mathbb{V}_1} \equiv \mathrm{Tr}_{\mathbb{V}_2}\rho$: a positive-semidefinite square matrix in $\mathbb{V}_1$

*Note: generally, $\mathrm{rank}\ \rho_{\mathbb{V}_1} > 1$

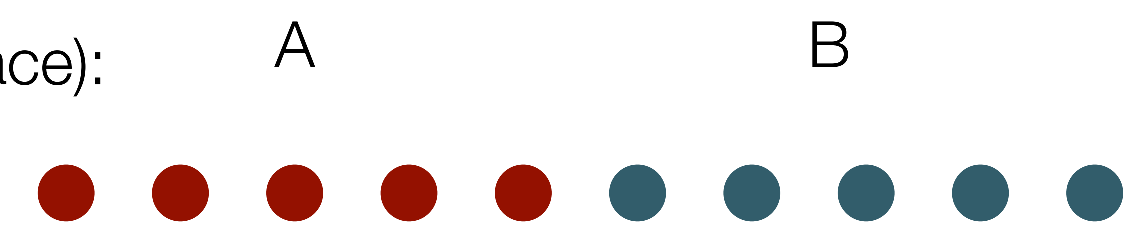
$$\boxed{(\rho_{\mathbb{V}_1})_{i_1,j_1} = \sum_{\underline{i_2}} \rho_{(i_1,\underline{i_2}),(j_1,\underline{i_2})}}$$



Entanglement entropy

Entanglement entropy:

Reduced density matrix of a sub system (sub space):



$$\rho_A = \mathrm{Tr}_B |\Psi\rangle\langle\Psi|$$

Entanglement entropy = von Neumann entropy of ρ_A

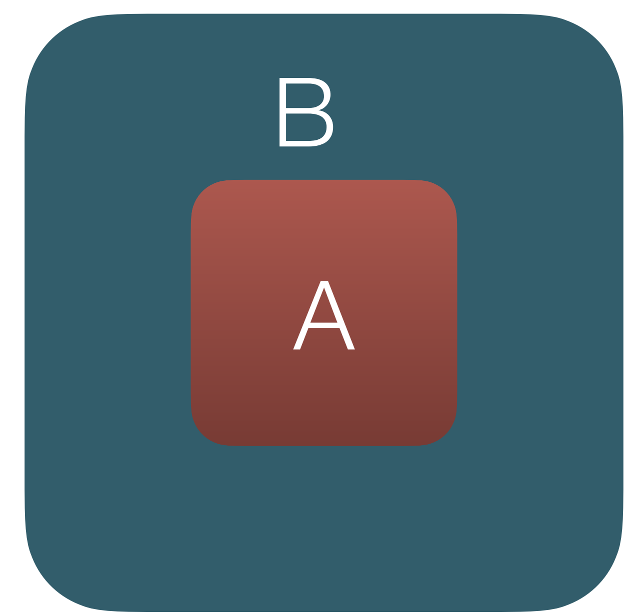
$$S = -\mathrm{Tr}\left(\rho_A \log \rho_A\right)$$

Schmidt decomposition $|\Psi\rangle = \sum_i \lambda_i |\alpha_i\rangle \otimes |\beta_i\rangle$

$$\rho_A = \sum_i \lambda_i^2 |\alpha_i\rangle\langle\alpha_i|$$ (*Exercise)

$$S = -\sum_i \lambda_i^2 \log \lambda_i^2$$

$$(\sum_i \lambda_i^2 = 1)$$



Entanglement entropy is calculated through the spectrum of Schmidt coefficients.

(It also indicates $S = -\mathrm{Tr}(\rho_B \log \rho_B)$)

Intuition for EE

Entanglement entropy is related to spectrum of singular values.

$$S = -\mathrm{Tr}(\rho_A \log \rho_A) = -\sum_i \lambda_i^2 \log \lambda_i^2$$

- $\mathrm{rank}\rho_A = 1$

  $\lambda_1 = 1, \lambda_j = 0 \ (j \neq 1)$ ➡ $S = 0$

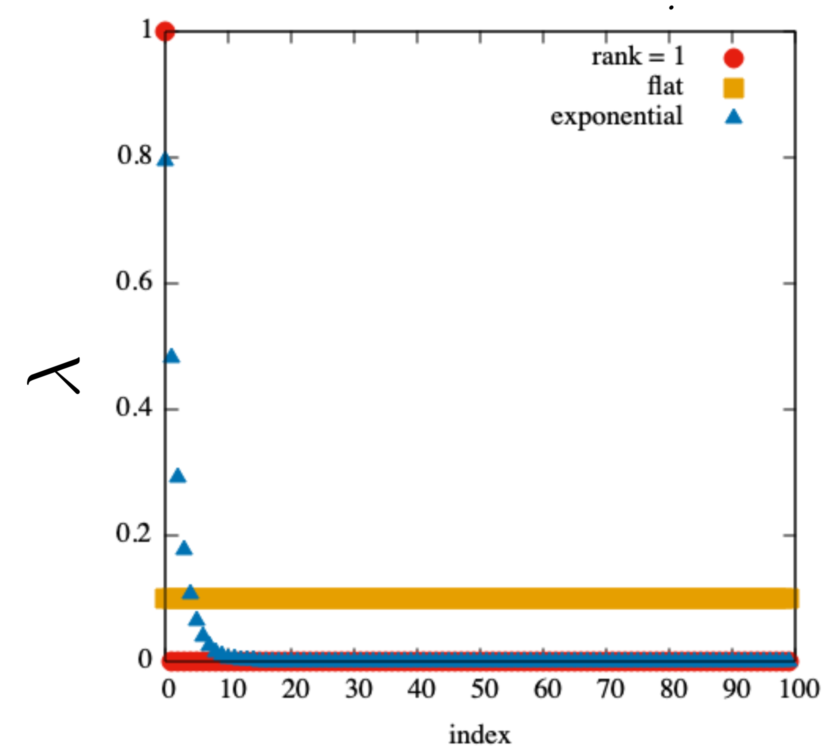
- Flat spectrum

  $\lambda_1 = \lambda_2 = \cdots = \lambda_n = \frac{1}{\sqrt{n}}$ ➡ $S = \log n$

- Exponential decay

  $\lambda_i \propto e^{-\alpha i}$ ➡ $S = 1 - \log 2\alpha \ (\alpha \ll 1, \alpha n \to \infty)$

Normalization: $(\sum \lambda_i^2 = 1)$



Smaller exponent gives larger entropy.

Intuition for EE: two S=1/2 spins

1. $|\Psi\rangle = |\uparrow\rangle \otimes |\downarrow\rangle$

A product state → $\lambda = 1, \; S = 0$

2. $|\Psi\rangle = \frac{1}{2}(|\uparrow\rangle - |\downarrow\rangle) \otimes (|\uparrow\rangle - |\downarrow\rangle)$

Another product state → $\lambda = 1 \; , \; S = 0$

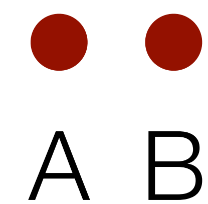
3. $|\Psi\rangle = \frac{1}{\sqrt{2}}(|\uparrow\rangle \otimes |\downarrow\rangle - |\downarrow\rangle \otimes |\uparrow\rangle)$

Spin singlet → $\lambda_1 = \lambda_2 = \frac{1}{\sqrt{2}} \; , \quad S = \log 2$

4. $|\Psi\rangle = \left(x|\uparrow\rangle \otimes |\downarrow\rangle + \sqrt{1-x^2}|\downarrow\rangle \otimes |\uparrow\rangle\right)$

Complicated state → $\lambda_1 = |x|, \lambda_2 = \sqrt{1-x^2}$

$S = x^2 \log x^2 + \sqrt{1-x^2} \log(1-x^2)$



Product state : S=0

Maximally entangled State

Larger entanglement entropy ~ Larger correlation between two parts

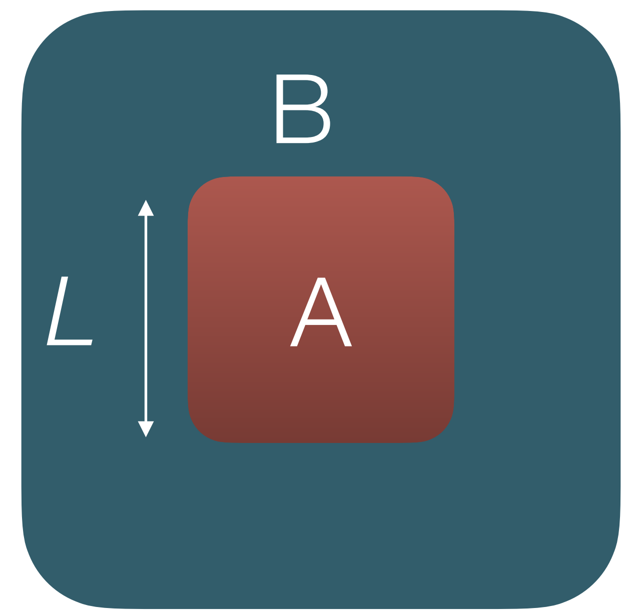
Area law of the entanglement entropy in physics

General wave functions (vector):

EE is proportional to its **volume (# of qubits)**.

$$S = -\mathrm{Tr}\left(\rho_A \log \rho_A\right) \propto L^d$$

(c.f. random vector)



Ground state wave functions:

For a lot of ground states, EE is proportional to its area.

J. Eisert, M. Cramer, and M. B. Plenio, Rev. Mod. Phys, 277, **82** (2010)

$$S = -\mathrm{Tr}\left(\rho_A \log \rho_A\right) \propto L^{d-1}$$

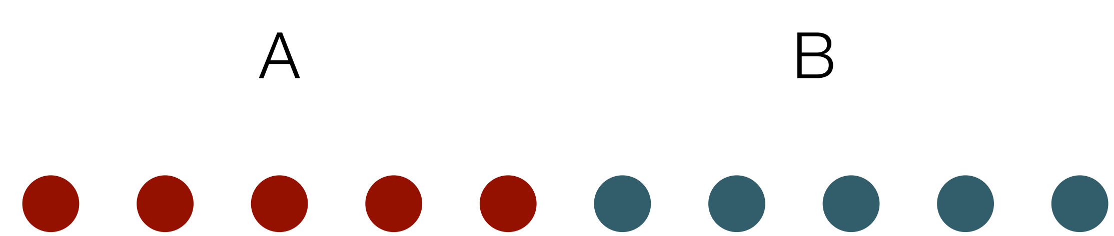
In the case of one-dimensional system:

Gapped ground state for local Hamiltonian

M.B. Hastings, J. Stat. Mech.: Theory Exp. P08024 (2007)

$$S = O(1)$$

Ground state are in a small part of the huge Hilbert space



Expected entanglement scaling for spin systems

Table 1
Entanglement entropy scaling for various examples of states of matter, either disordered, ordered, or critical, with smooth boundaries (no corners).

Physical state	Entropy	Example
Gapped (brok. disc. sym.)	$aL^{d-1} + \ln(\text{deg})$	Gapped XXZ [143]
$d = 1$ CFT	$\frac{c}{3} \ln L$	$s = \frac{1}{2}$ Heisenberg chain [21]
$d \geq 2$ QCP	$aL^{d-1} + \gamma_{\text{QCP}}$	Wilson–Fisher O(N) [136]
Ordered (brok. cont. sym.)	$aL^{d-1} + \frac{n_G}{2} \ln L$	Superfluid, Néel order [147]
Topological order	$aL^{d-1} - \gamma_{\text{top}}$	$\mathbb{Z}_2$ spin liquid [159]

(Nicolas Laflorencie, Physics Reports **646**, 1 (2016))

cf. free fermion

$$S \propto L^{d-1} \log L$$

For d ≥ 2, leading contribution satisfies area low
even for gapless (critical) systems.

Exercise: examples of Schmidt decomposition

1-1: Random wave function

(Sample code: Ex1-1.ipynb)

- Make a random vector
- SVD it and see singular value spectrum and EE

1-2: Ground state of the transverse field Ising model

$$\mathcal{H} = -\sum_{i=1}^{L-1} S_{i,z} S_{i+1,z} - \Gamma \sum_{i=1}^{L} S_{i,x}$$

(Sample code: Ex1-2.ipynb)

- Calculate GS by diagonalizing Hamiltonian
- SVD it and see singular value spectrum and EE

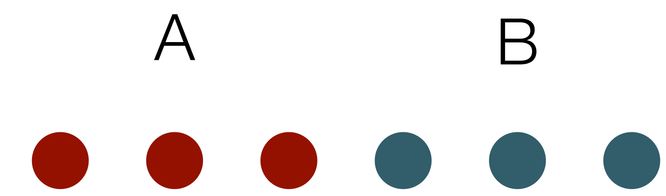
(Sample code: Ex1-3.ipynb)

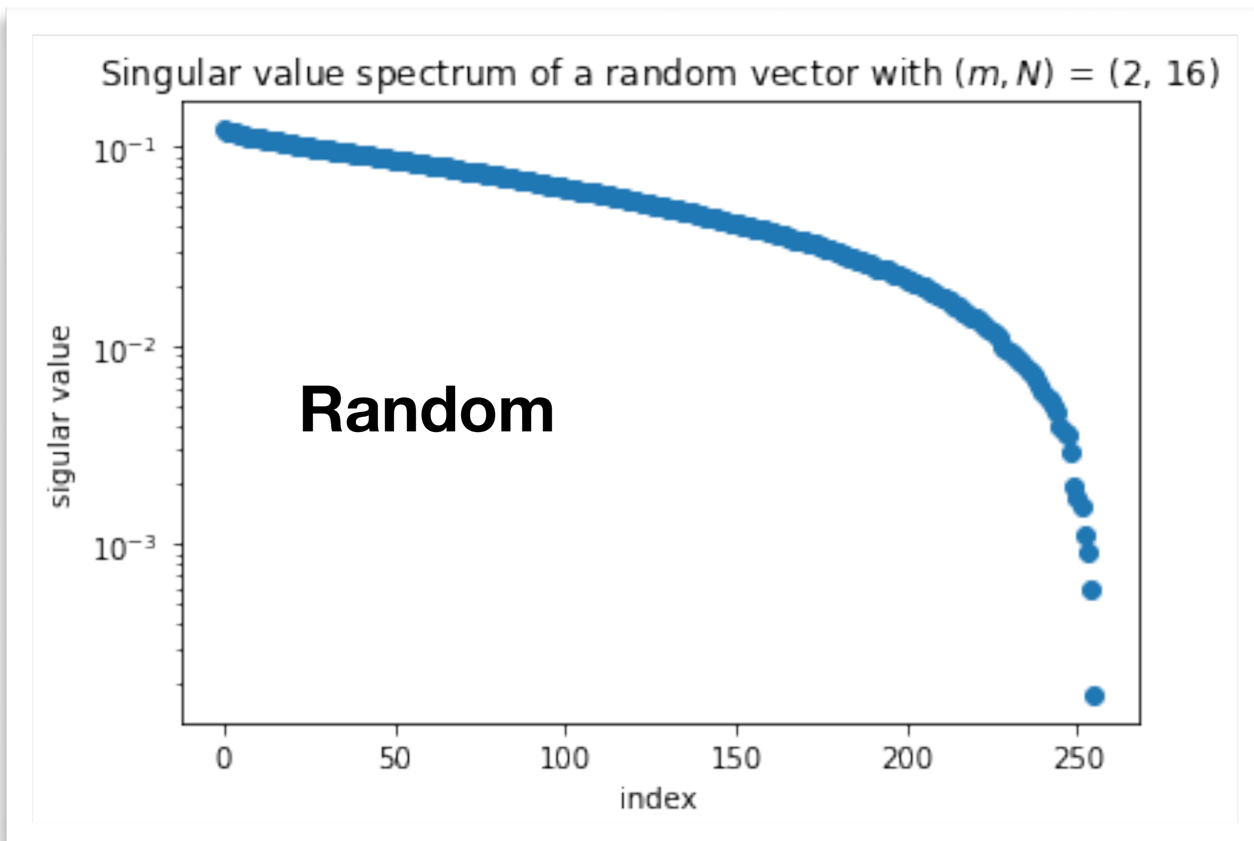
1-3: Picture image

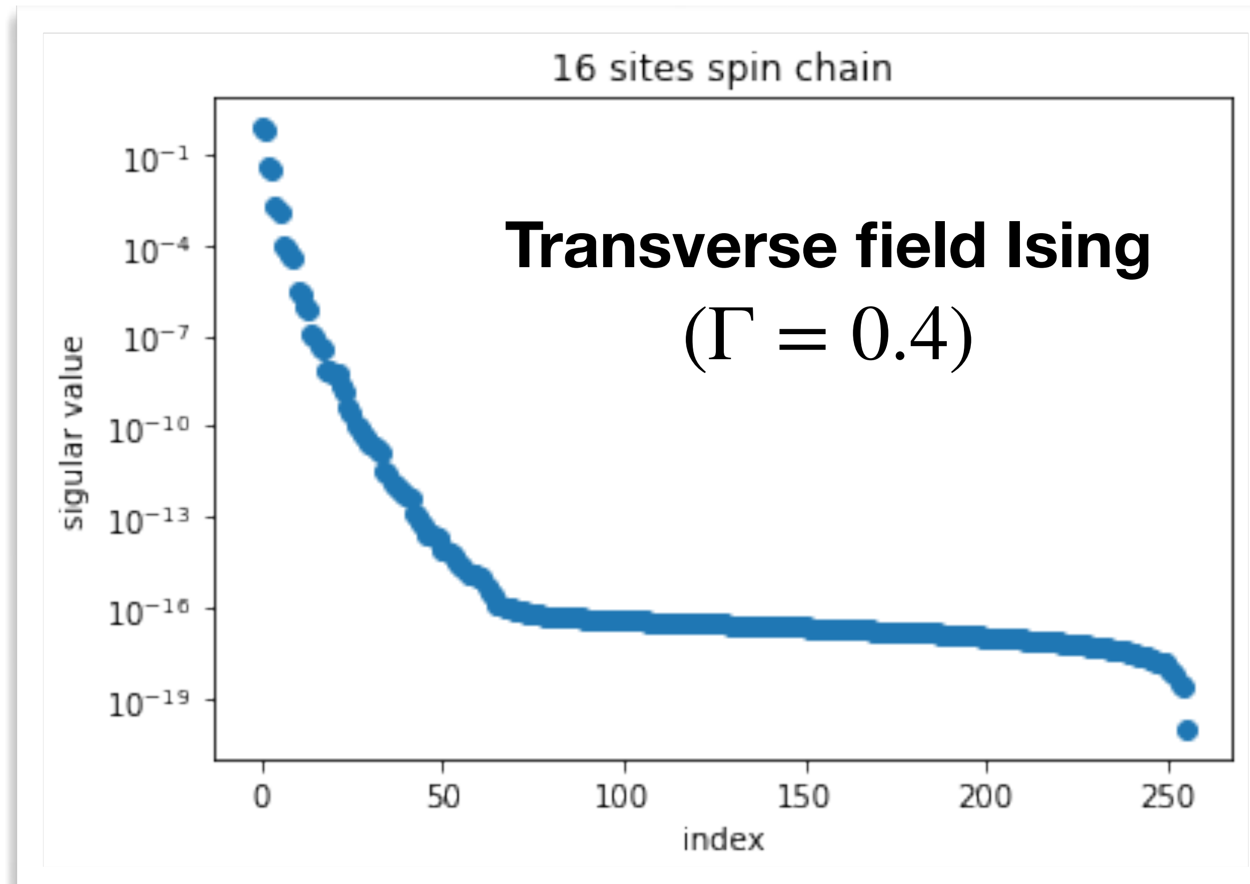
- Transform an image data to the vector in m^N dimension.
- SVD it and see singular value spectrum and EE

*** Try to simulate different system size "N"**
*** You can simulate other S by changing "m"**

Spectrum for N=16 $\vec{v} \in \mathbb{C}^{2^{16}}$



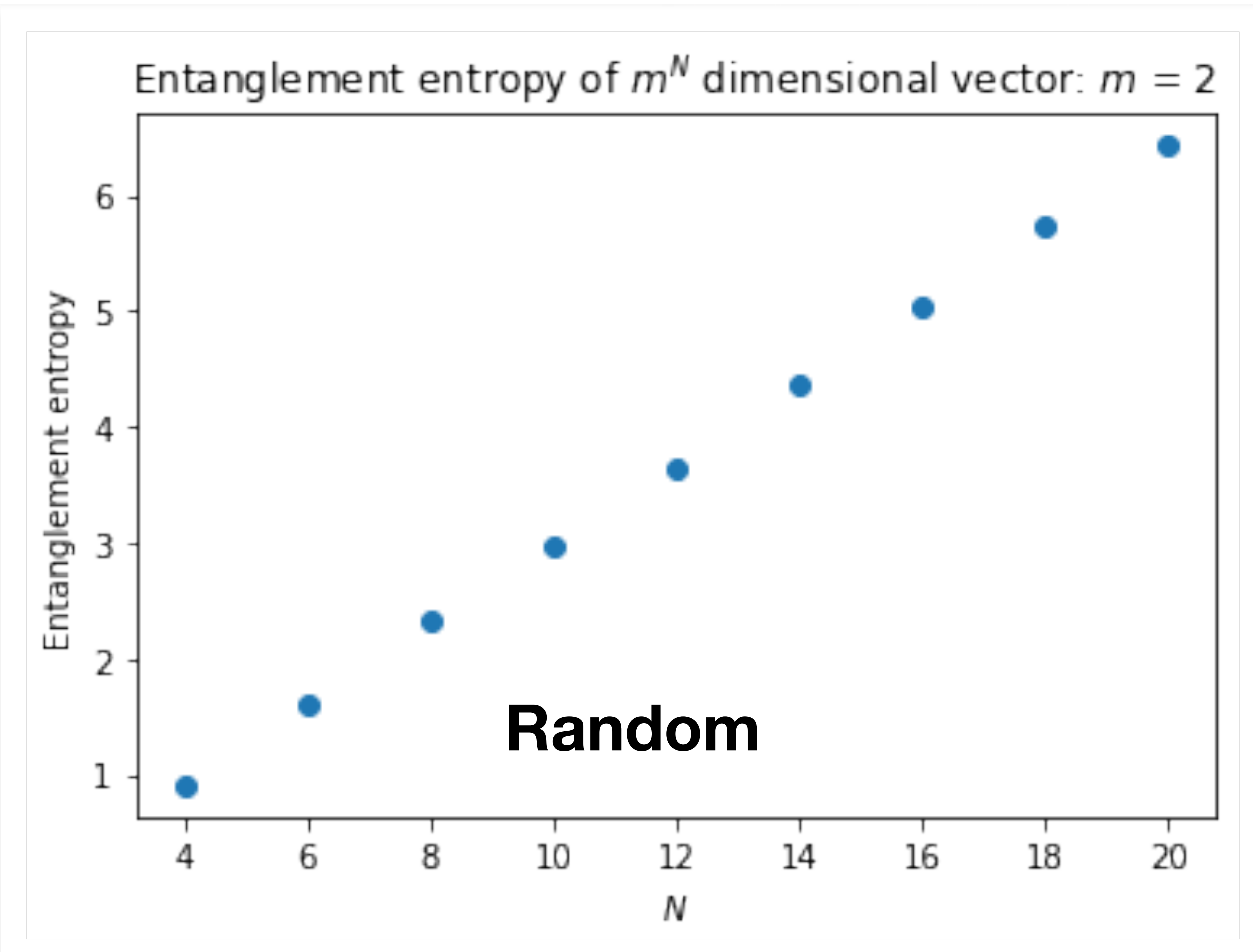


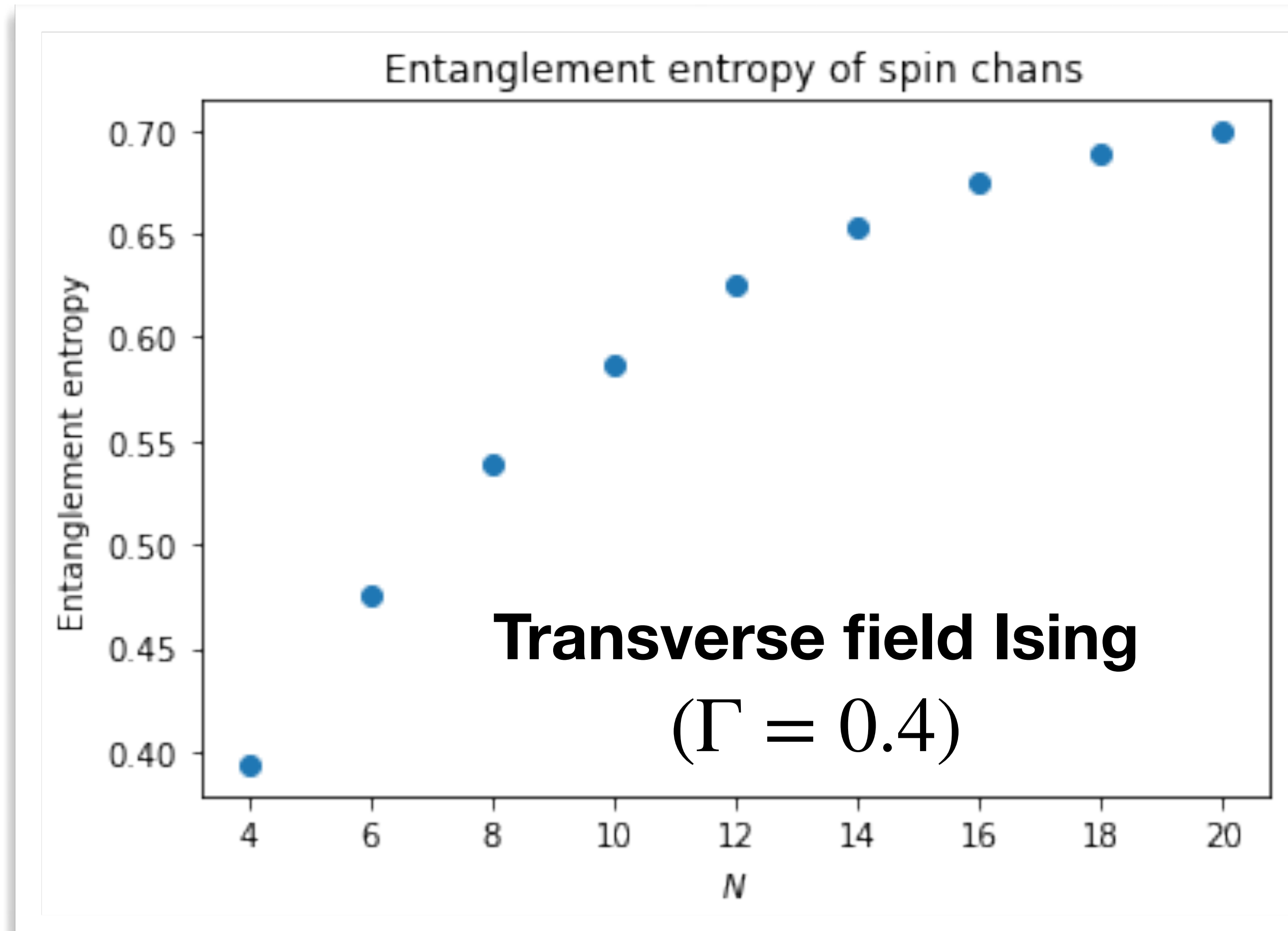


Ground state wave function has lower entanglement!

Scaling of the entanglement entropy

$\vec{v} \in \mathbb{C}^{2^N}$





Random vector: Volume low
Ground state: Area low

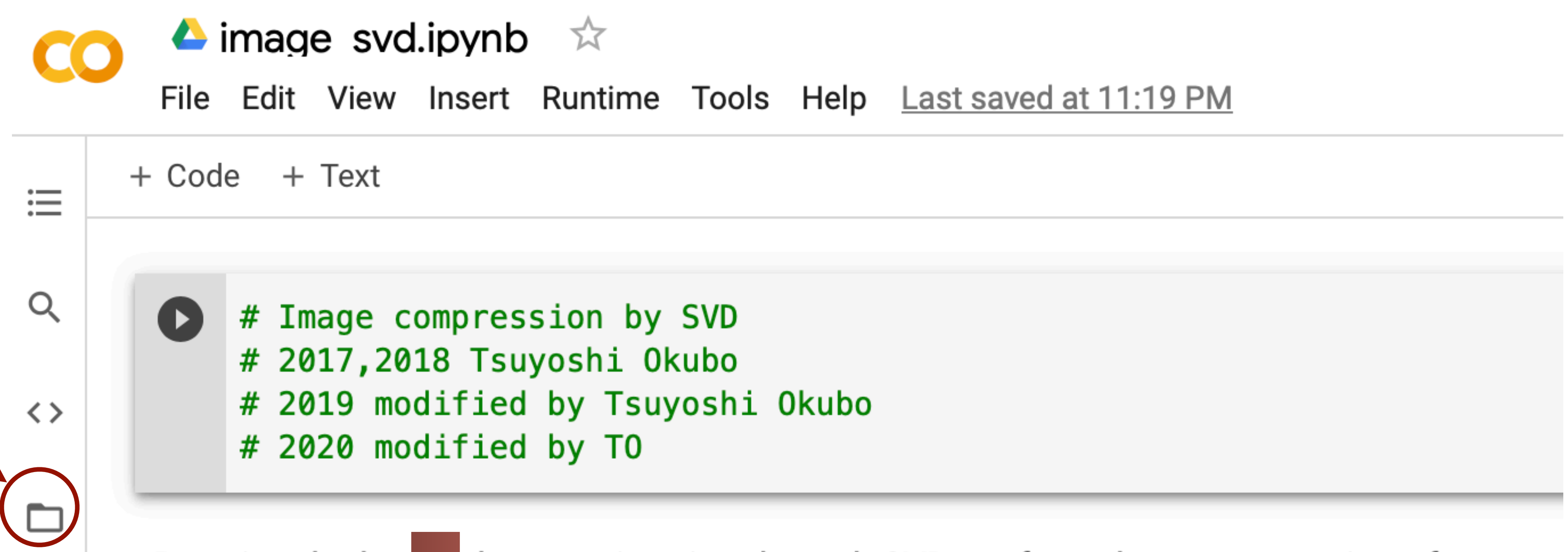
Exercises with Google Colab

I recommend you to use google colaboratory,
https://colab.research.google.com
where you can run .ipynb from your web browser.

When you use Google Colab, you need to also upload
"ED.py"
for the case of "Ex1-2.ipynb", and
your image file (sample.jpg),
for the case of "Ex1-3.ipynb".

How to use Google Colab

1. Open Ex1-3.ipynb in Google colab
    - Select "**File/upload notebook**" ("ファイル/ノートブックをアップロード") and upload Ex1-3.*ipynb*

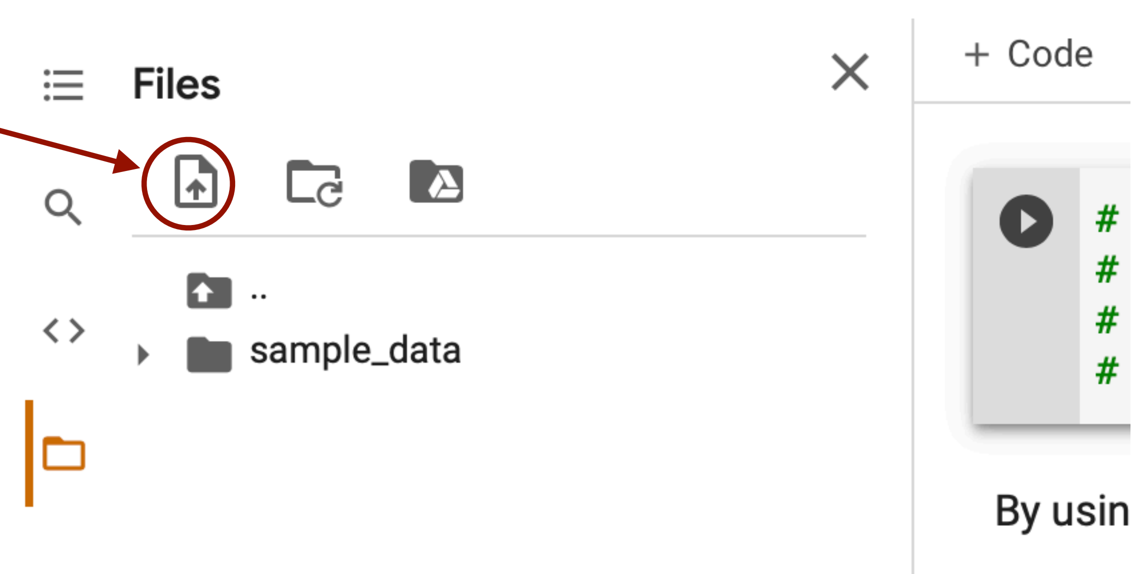


2. Click here

    (Wait a moment for the connection)

3. Click here and upload your image file (e.g. sample.jpg).

    (Uploaded file will be deleted after the session finishes.)



4. Select "**Runtime/Restart and Run all**".

Matrix product states（行列積状態）

(Tensor train decomposition)

Data compression of tensors (vectors)

Eg. General wave function:

$$|\Psi\rangle = \sum_{\{i_1, i_2, \ldots i_N\}} \Psi_{i_1 i_2 \ldots i_N} |i_1 i_2 \ldots i_N\rangle$$

Coefficient vector can represent any points in the Hilbert space.

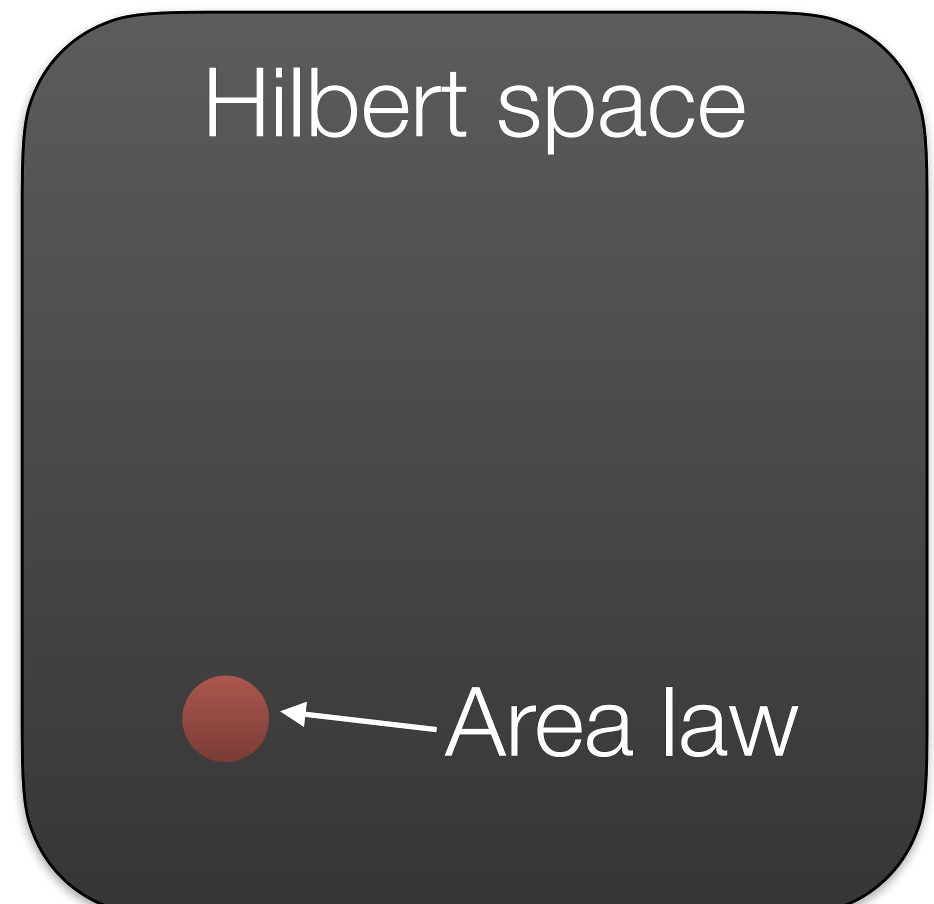
Ground states satisfy the area law.

In order to represent the ground state accurately,
we might not need all of a^N elements.

Data compression by tensor decomposition:

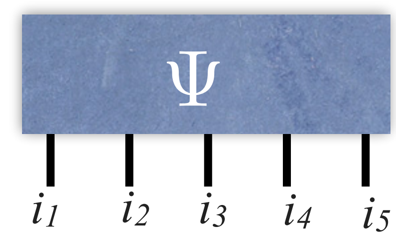
Tensor network decomposition

***Same idea holds for any tensors.**



Tensor network decomposition (tensor network states)

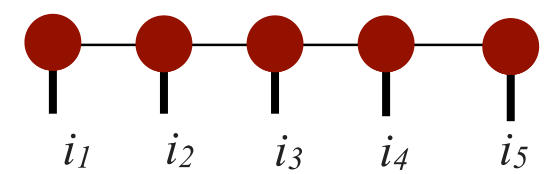
Vector (or N-leg tensor): $\Psi_{i_1 i_2 \ldots i_N}$ =



# of Elements$=a^N$

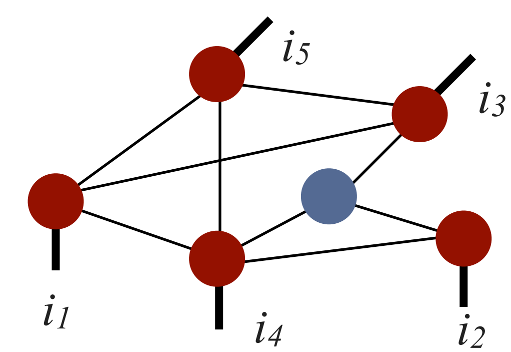
"Tensor network" decomposition

- Matrix Product State (MPS) $A_1[i_1]A_2[i_2]\cdots A_N[i_N]$ =



$A[m]$: Matrix for state m

- General network $\mathrm{Tr}X_1[i_1]X_2[i_2]X_3[i_3]X_4[i_4]X_5[i_5]Y$



X,Y : Tensors

Tr : Tensor network contraction

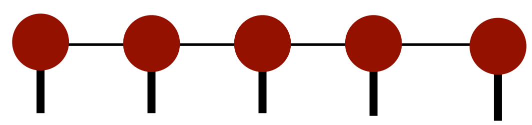
By choosing a "good" network, we can express target vector efficiently.

ex. MPS: # of elements $=2ND^2$ D: dimension of the matrix A

Exponential→ Linear *If D does not depend on N…

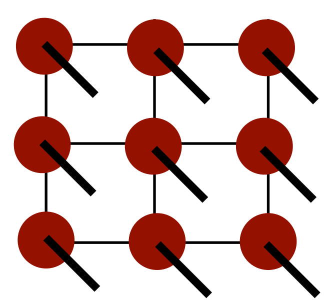
Examples of TNS

MPS:



Good for *1*d short range correlation
(e.g. *1*d gapped systems)

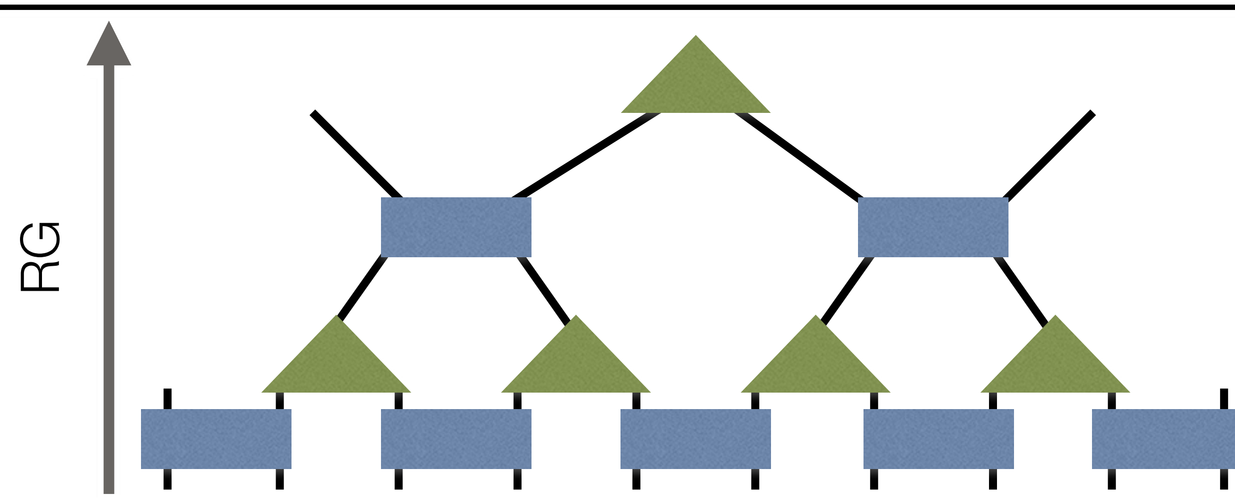
PEPS, TPS:



For higher dimensional correlation

Extension of MPS

MERA:



Scale invariant systems

Matrix product state (MPS)

Good reviews:

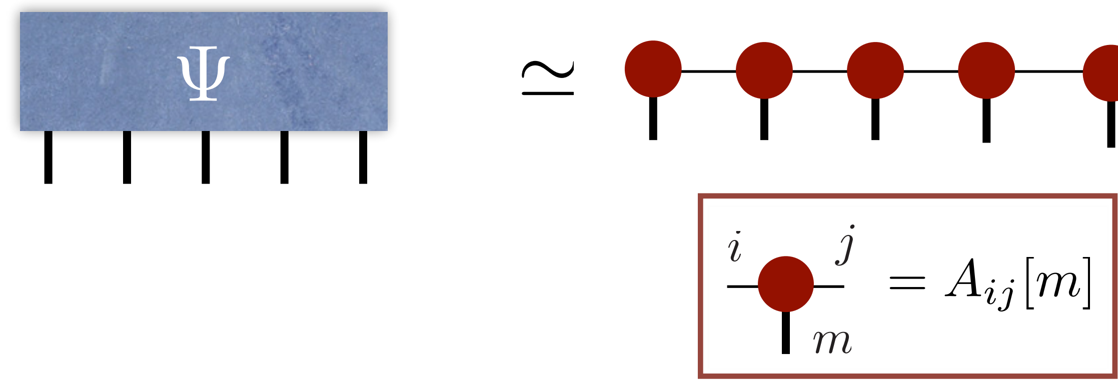
(U. Schollwöck, Annals. of Physics **326**, 96 (2011))
(R. Orús, Annals. of Physics **349**, 117 (2014))

$$|\Psi\rangle = \sum_{\{i_1, i_2, \dots i_N\}} \Psi_{i_1 i_2 \dots i_N} |i_1 i_2 \dots i_N\rangle$$

$$\Psi_{i_1 i_2 \dots i_N} \simeq A_1[i_1] A_2[i_2] \cdots A_N[i_N]$$

$A[i]$: Matrix for state i

MPS



Note:

- MPS is called "tensor train decomposition" in applied mathematics

(I. V. Oseledets, SIAM J. Sci. Comput. **33**, 2295 (2011))

- A product state is represented by MPS with 1×1 "Matrix" (scalar)

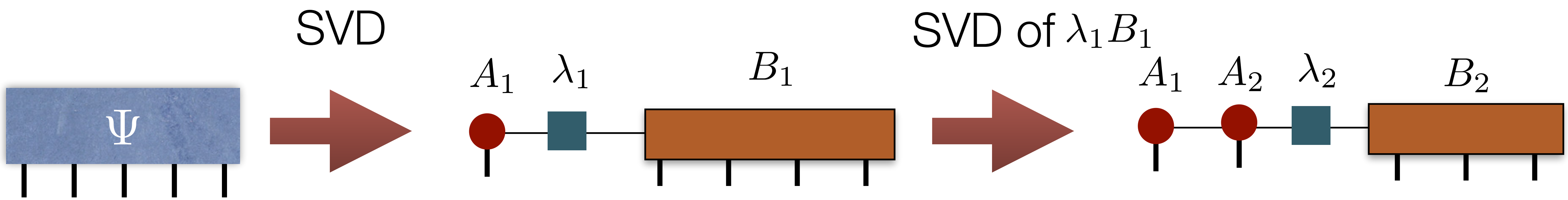
$$|\Psi\rangle = |\phi_1\rangle \otimes |\phi_2\rangle \otimes \cdots$$

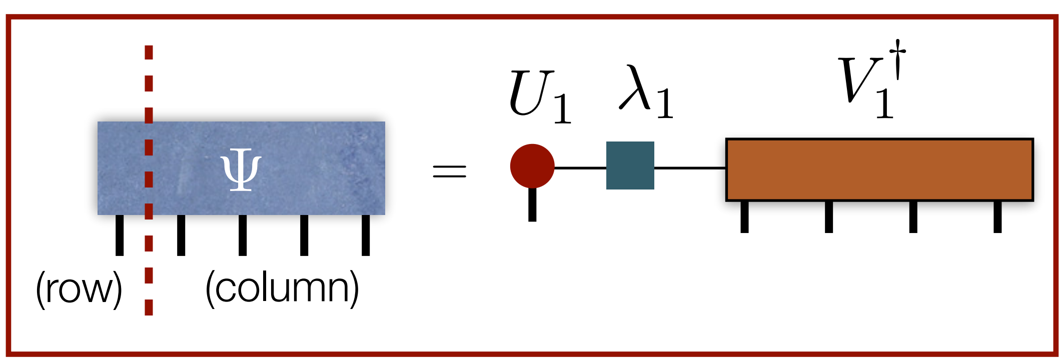
$$\Psi_{i_1 i_2 \dots i_N} = \phi_1[i_1] \phi_2[i_2] \cdots \phi_N[i_N]$$

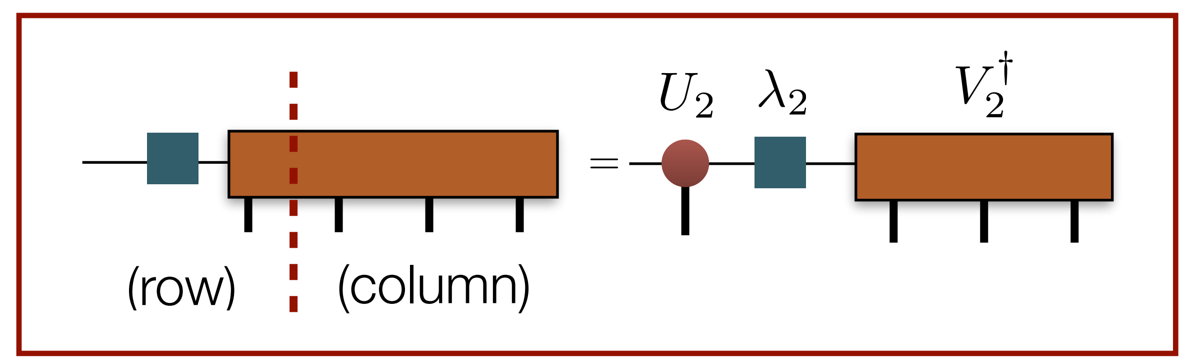
$$\phi_n[i] \equiv \langle i | \phi_i \rangle$$

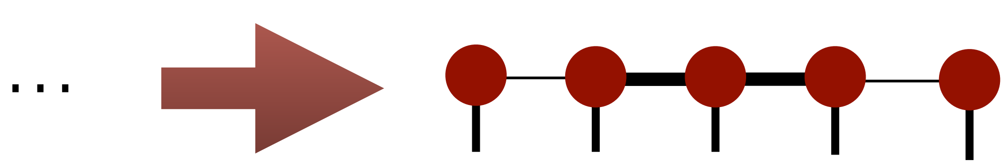
Matrix product state without approximation

General vectors can be represented by MPS exactly
through successive Schmidt decompositions







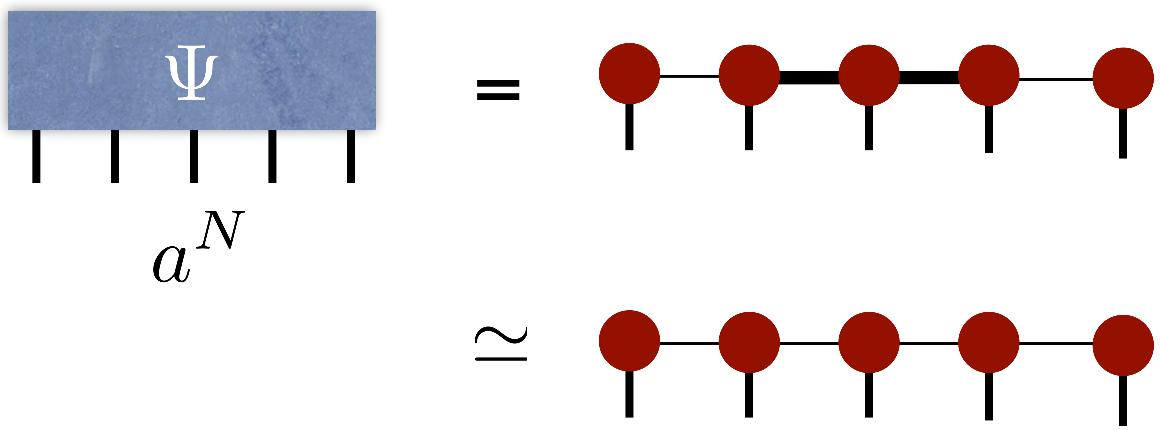


In this construction, the sizes of matrices
depend on the position.

Maximum **bond dimension** = $a^{N/2}$

At this stage, **no data compression.**

Matrix product state: Low rank approximation



a^N

Maximum bond dimension = $a^{N/2}$

a^N

Constant bond dimension = χ

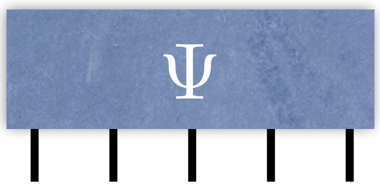
$\propto Na\chi^2$

If the entanglement entropy of the system is **O(1)** (independent of *N*), matrix size "χ" can be small for accurate approximation.

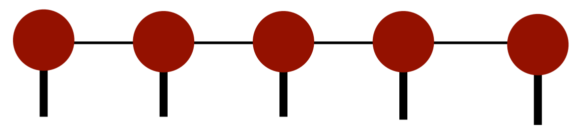
MPS is good for gapped 1d systems.

On the other hand, if the EE increases as increase *N*, "χ" must be increased to keep the same accuracy.

Upper bound of Entanglement entropy

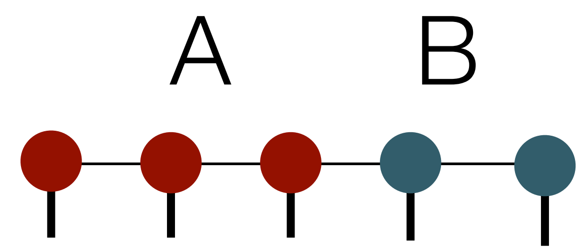


$\simeq$

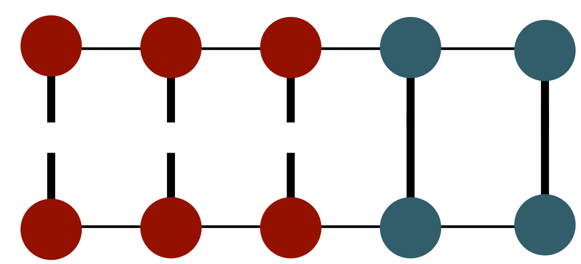


$\equiv |\tilde{\Psi}\rangle$:MPS with χ

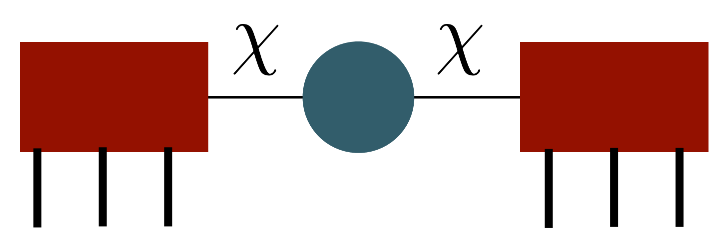
Reduced density matrix of region A:



$$\rho_A = \mathrm{Tr}_B |\tilde{\Psi}\rangle\langle\tilde{\Psi}| =$$



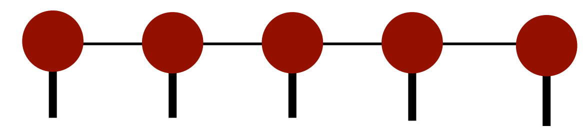
Structure of ρ_A:



$$\boxed{\begin{aligned} &\mathrm{rank}\ \rho_A \leq \chi \\ &S_A = -\mathrm{Tr}\ \rho_A \log \rho_A \leq \log \chi \end{aligned}}$$

Required bond dimension in MPS representation

$$\boxed{S_A = -\mathrm{Tr}\, \rho_A \log \rho_A \leq \log \chi}$$



The upper bound is independent of the "length".

length of MPS ⇔ size of the problem

N $\qquad\qquad$ a^N

EE of the original vector	Required bond dimension in MPS representation
$S_A = O(1)$	$\chi = O(1)$
$S_A = O(\log N)$	$\chi = O(N^\alpha)$
$S_A = O(N^\alpha)$	$\chi = O(c^{N^\alpha})$

$(\alpha \leq 1)$

Next week

1st: Huge data in modern physics (Today)

2nd: Information compression in modern physics
(+review of linear algebra)

3rd: Review of linear algebra (+ singular value decomposition)

4th: Singular value decomposition and low rank approximation

5th: Basics of sparse modeling

6th: Basics of Krylov subspace methods

7th: Information compression in materials science

8th: Accelerating data analysis: Application of sparse modeling

9th: Data compression: Application of Krylov subspace method

10th: Entanglement of information and matrix product states

11th: Matrix product states + Application of MPS to eigenvalue problems

12th: Application of MPS to time evolution and data science

13th: Other tensor network representations
+ (Appendix: Information compression by tensor network renormalization)