第1回量子ソフトウェアワークショップ:テンソルネットワークと量子計算の展望

主催：東京大学大学院理学系研究科「量子ソフトウェア」寄付講座

機械学習における テンソルネットワークの活用

2021年12月7日

原田健自

京都大学大学院情報学研究科

機械学習

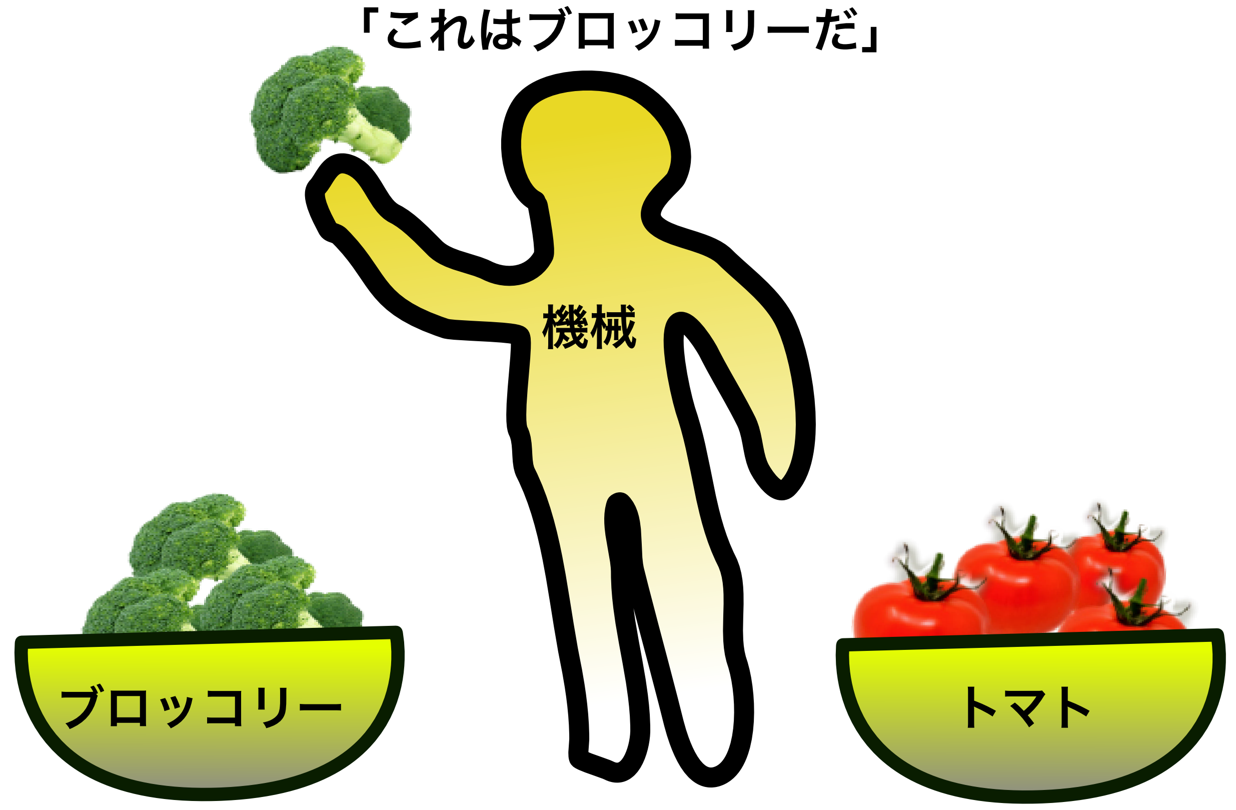
- データを学習することで高度な情報処理をする
  - 分類：入力データに対応するラベルを予測

学習用データ

$$(\mathbf{x}^1, l^1), \cdots (\mathbf{x}^{1000}, l^{1000})$$

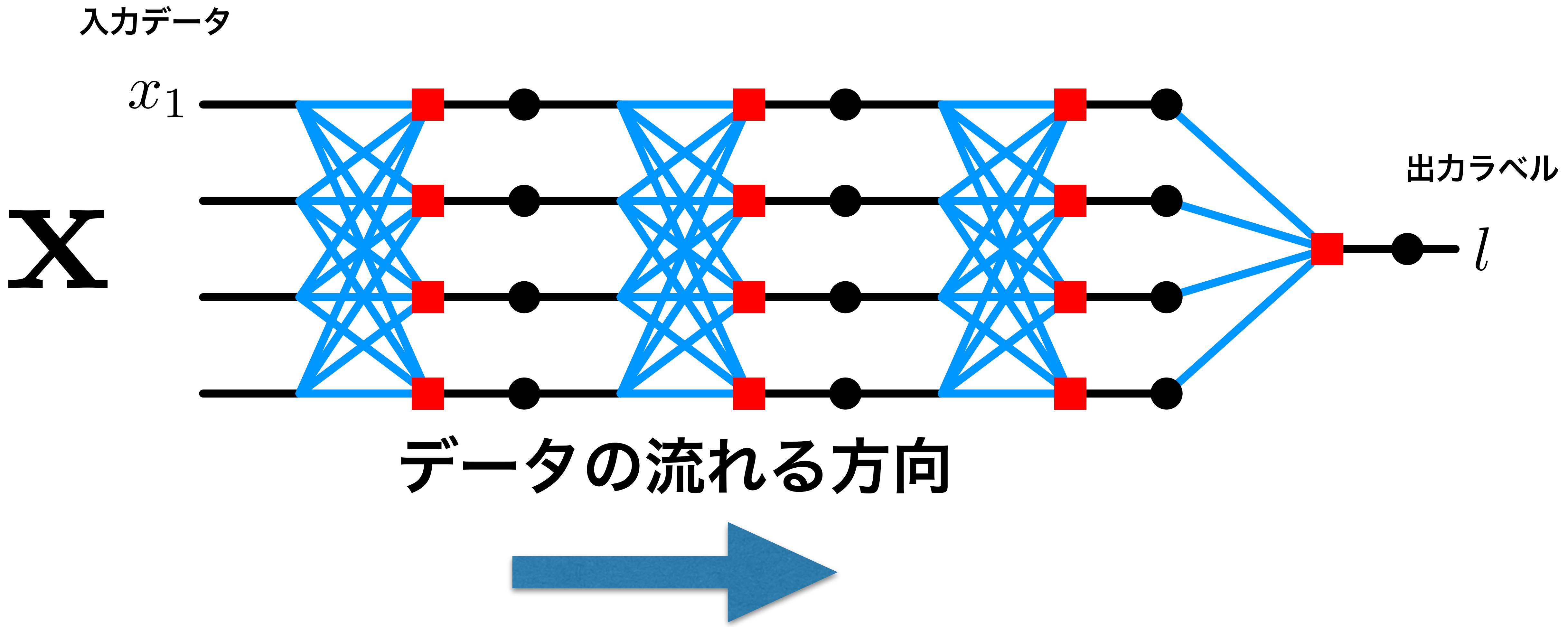
入力データ　ラベル

$$\mathbf{x} \longrightarrow l$$



  - 生成：学習データに基づき，学習データに似た人工データを生成

- ニューラルネットワーク
  - 正しいの出力ラベルが出るように
    モデルパラメータを最適化（学習）



分類問題へのテンソルネットワークの活用

- ニューラルネットワークのテンソルネットワーク圧縮
  - A. Novikov, D. Podoprikhin, A. Osokin, and D. Vetrov, “Tensorizing Neural Networks,” NIPS 2016.

- 量子機械学習
  - E. Stoudenmire and D. J. Schwab, “Supervised Learning with Tensor Networks,” NIPS 2016.

両者ともテンソルデータの圧縮に
テンソルネットワークを活用

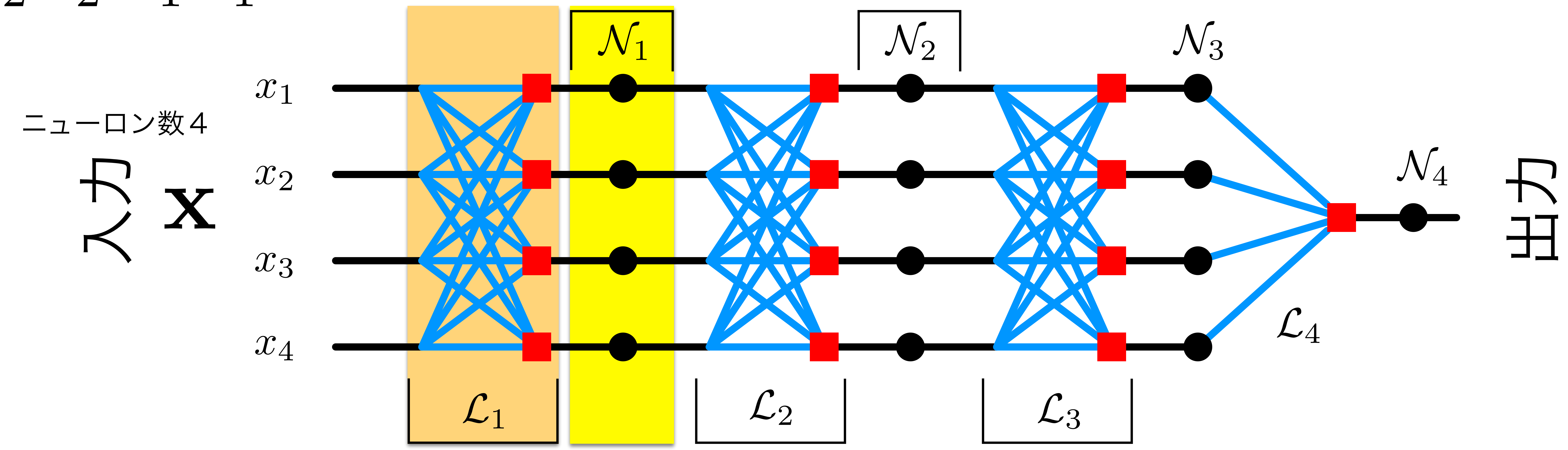
深層フィードフォワード型ニューラルネットワーク

層構造

$$\mathcal{F} = \mathcal{N}_n \mathcal{L}_n \cdots \mathcal{N}_2 \mathcal{L}_2 \mathcal{N}_1 \mathcal{L}_1$$

非線形層：シグモイド、ReLU

$$\mathbf{y} = f(\mathbf{x})$$



$$\mathbf{y} = W\mathbf{x} + \mathbf{b}$$

線形層

全結合の場合の重み行列Wは、ニューロン数の二乗程度の要素数

テンソルネットワークを用いて重み行列を圧縮する事は可能か？

A. Novikov, D. Podoprikhin, A. Osokin, and D. Vetrov, “Tensorizing Neural Networks,” NIPS 2016.
Z.-F. Gao, et al., “Compressing deep neural networks by matrix product operators,” Phys. Rev. Research, vol.2, 023300 (2020).

情報圧縮：特異値分解

- 行列の特異値分解による近似

$$A = U\Lambda V^t = U \begin{pmatrix} \lambda_1 & 0 & \cdots & & \\ 0 & \lambda_2 & 0 & \cdots & \\ \cdots & 0 & \lambda_3 & 0 & \cdots \\ & & \cdots & & \end{pmatrix} V^t \approx \tilde{U}\;\tilde{\Lambda}\;\tilde{V}^t = \tilde{A}$$

回転行列　回転行列

小さな特異値を省く

特異値 $\lambda_1 \geq \lambda_2 \geq \cdots \geq 0$

- 最良近似

$$\tilde{A} = \arg\min ||A - M||$$

ランクkの行列　特異値がk個

- パラメータ数

$$A : \mathrm{n} \times \mathrm{m} \Rightarrow \tilde{A} : \mathrm{n} \times \mathrm{m},$$
$$\tilde{U} : \mathrm{n} \times \mathrm{k}, \quad \tilde{\Lambda} : \mathrm{k} \times \mathrm{k}, \quad \tilde{V} : \mathrm{m} \times \mathrm{k}.$$

nとmに比例

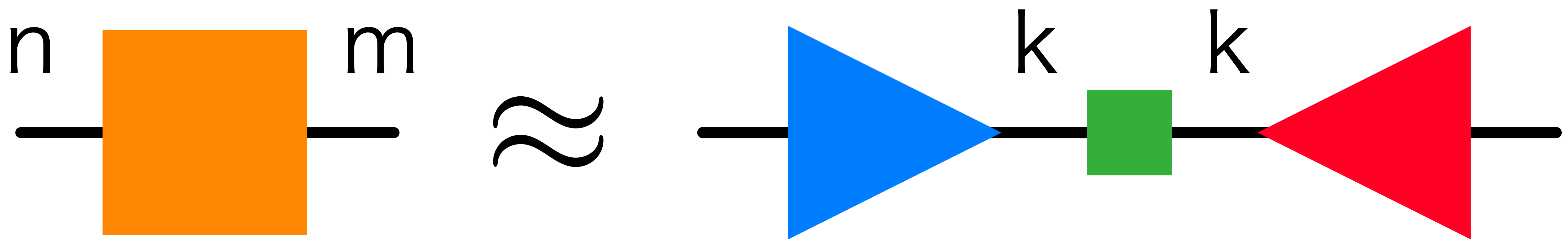
$$\mathrm{n} \times \mathrm{m} \Rightarrow (\mathrm{n} \times \mathrm{k}) + (\mathrm{m} \times \mathrm{k}) + \mathrm{k}$$

例：n=m=1000 → n × m = 100万

k=10の場合，2010　(0.2%)

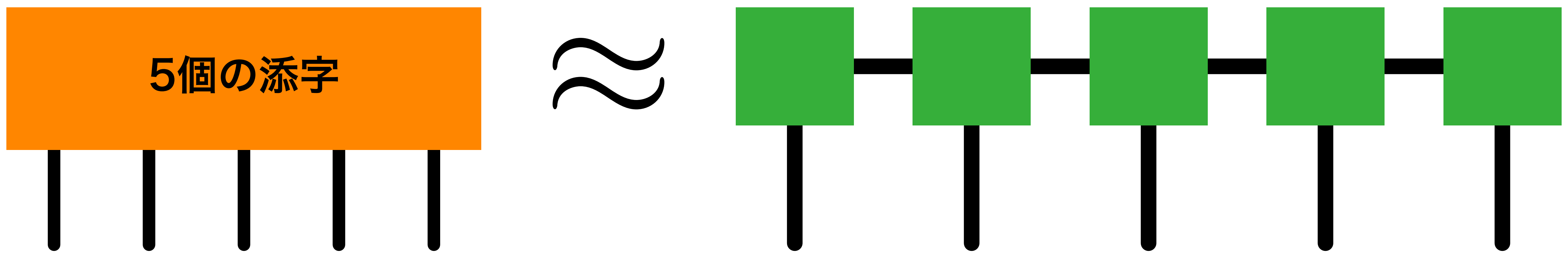
情報圧縮：テンソル分解

- 行列の特異値分解による近似 = テンソル分解 = テンソルネットワーク



$$\mathrm{n} \times \mathrm{m} \Rightarrow (\mathrm{n} \times \mathrm{k}) + (\mathrm{m} \times \mathrm{k}) + \mathrm{k}$$

- テンソルの行列積による近似



行列 j

$$T_{ijklm} \approx (UM^{j}M^{k}M^{l}V)_{ij}$$

パラメータ数 $n^5 \approx 5(nk^2)$ **指数関数的な圧縮** $\exp(\alpha D) \Rightarrow \beta D$

重み行列のテンソル化

重み行列

$$W = (W_i^j)$$

入力ニューロン番号iと出力ニューロン番号j間の結合重み： $\mathbf{y} = W\mathbf{x} + \mathbf{b}$

因数分解によるテンソル化

n：入力添字数

入力ニューロン数 $N_x = a_1 \times a_2 \times \cdots \times a_n$

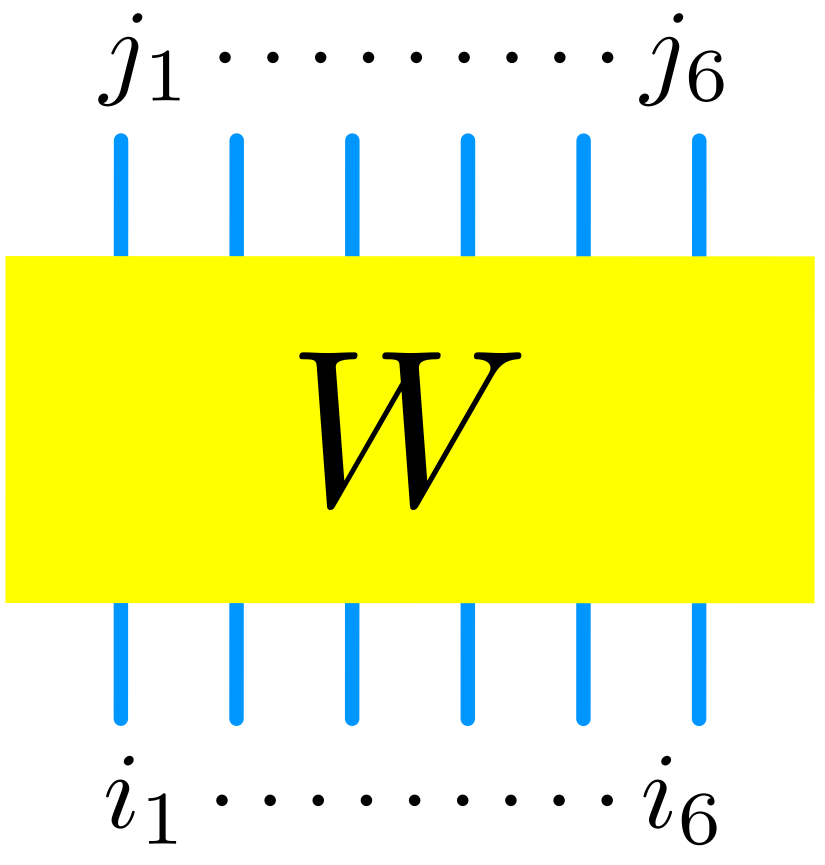
出力ニューロン数 $N_y = b_1 \times b_2 \times \cdots \times b_n$

$$i \in [0, N_x - 1], j \in [0, N_y - 1]$$

$$i = \sum_{l=1}^{n} a_l \cdot i_l \qquad i \to (i_1, \cdots, i_n)$$

$$j = \sum_{l=1}^{n} b_l \cdot j_l \qquad j \to (j_1, \cdots, j_n)$$

$$W_i^j = W_{i_1 \cdots i_n}^{j_1 \cdots j_n}$$

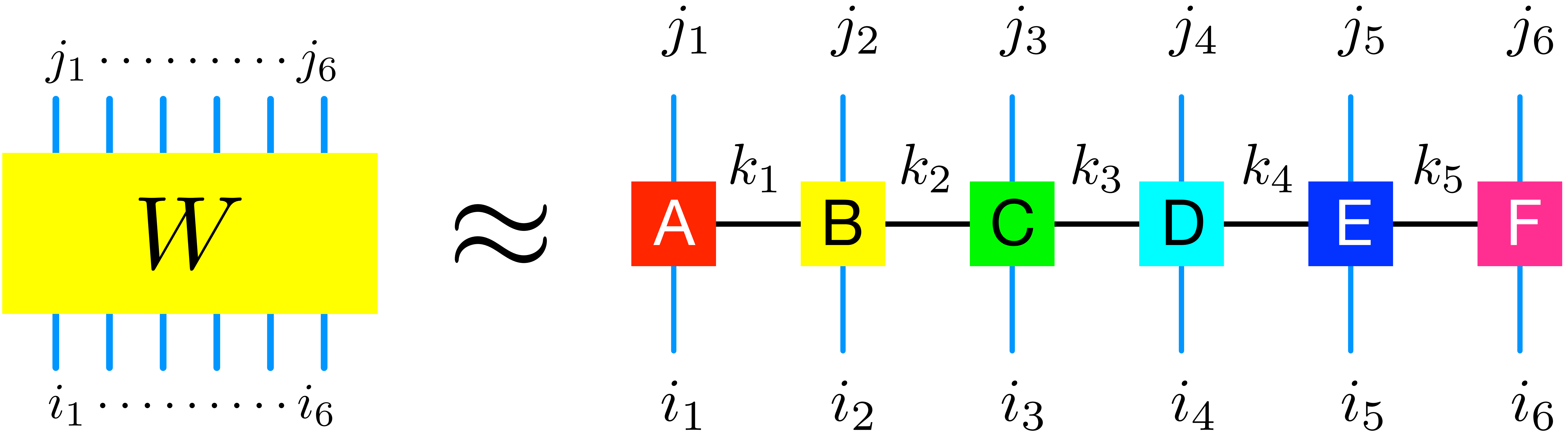


要素数

$N_x \times N_y$

重み行列のテンソルネットワーク化

$$W^{j_1\cdots j_n}_{i_1\cdots i_n} \approx \sum_{k_1\cdots k_5} A_{i_1 j_1 k_1} B_{i_2 j_2 k_1 k_2} C_{i_3 j_3 k_2 k_3} D_{i_4 j_4 k_3 k_4} E_{i_5 j_5 k_4 k_5} F_{i_6 j_6 k_5}$$



$$\chi_l = \dim(k_l)$$

行列積：Matrix Product Operator (MPO)

要素数の圧縮

$$N_x N_y = \prod_l (a_l b_l) \sim O(a^n b^n) \Leftrightarrow a_1 b_1 \chi_1 + \sum_{l=2}^{5} a_l b_l \chi_{l-1}\chi_l + a_6 b_6 \chi_5 \ \sim O(nab\chi^2)$$

指数n：入力添字数

nに比例

重み行列の要素数

MPOの要素数

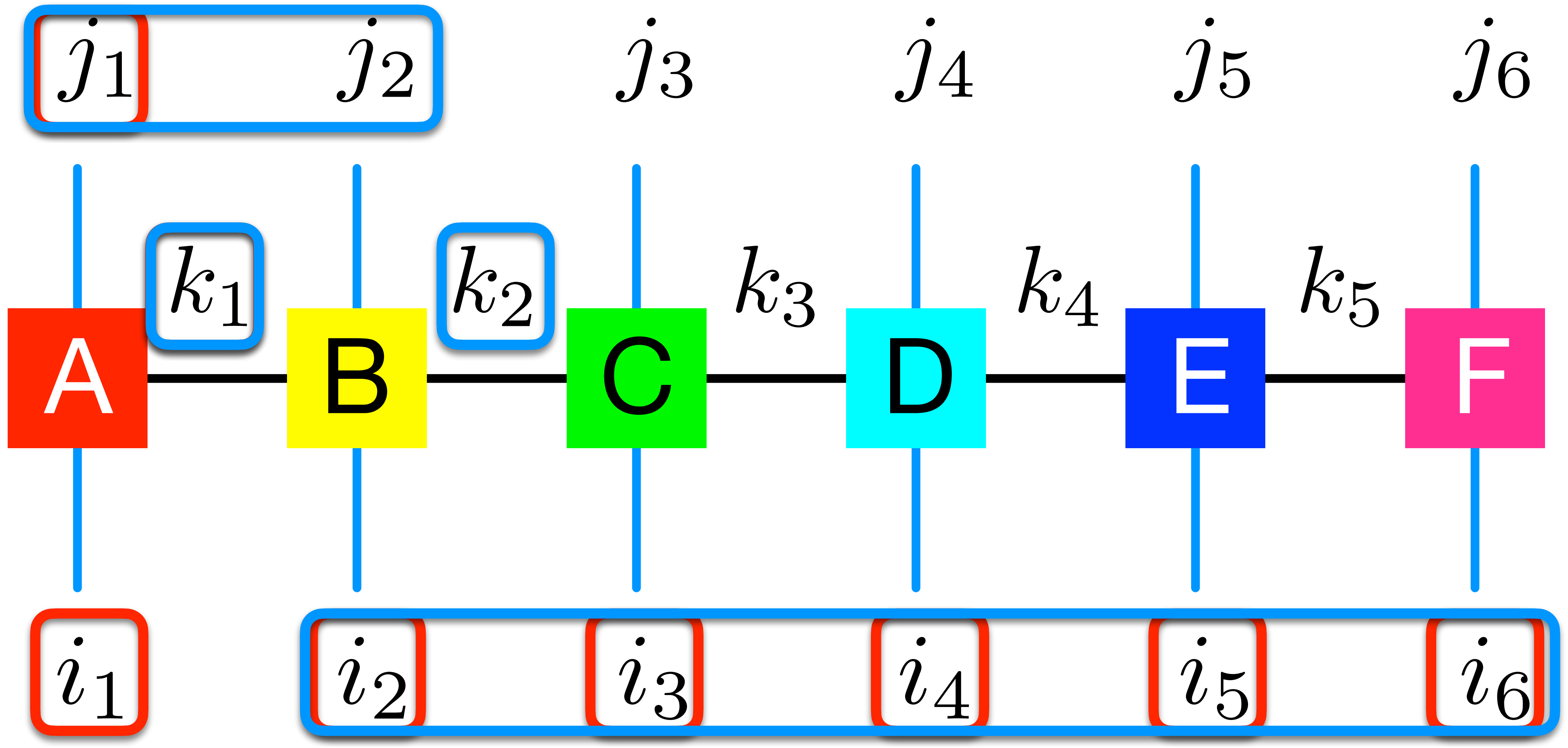
テンソル化ニューラルネットワーク

- 重み行列はMPOとし、MPO中のテンソルをパラメータとする
  - 重み成分の計算コスト

$$N = N_x = N_y = a^n \quad (\because a = b)$$

入力ベクトルにテンソルAをかけるコスト $Na\chi$

さらにテンソルBをかけるコスト $Na\chi^2$



テンソルを全てかけるコスト

$$(n-2)Na\chi^2 + 2Na\chi \to O(Nna\chi^2)$$

元のコスト N^2 → 圧縮後のパラメータ数×N÷a → N²×圧縮率÷a

  - 最適化

    自動微分を含んだ機械学習フレームワークの活用が可能

Z.-F. Gao, et al., "Compressing deep neural networks by matrix product operators," Phys. Rev. Research, vol.2, 023300 (2020).

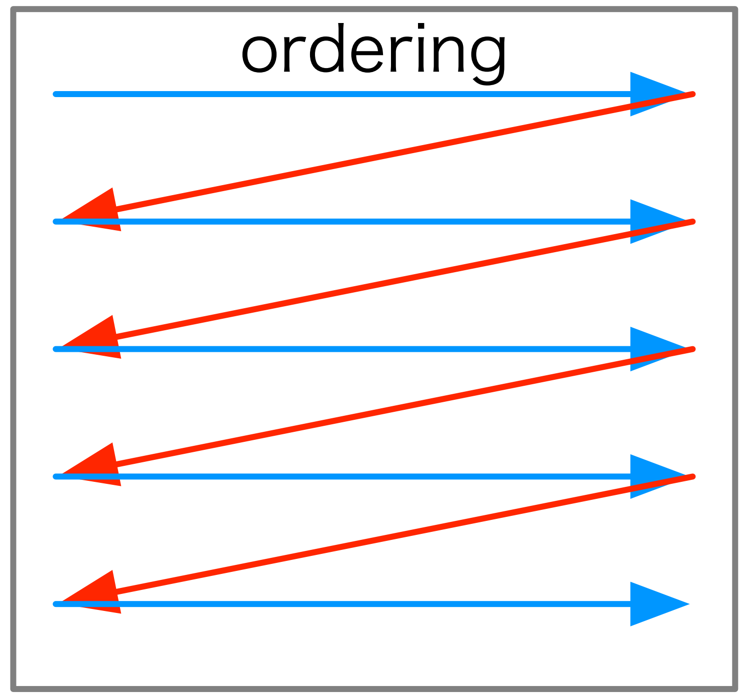
テンソル化の影響

(表はPRR, vol.2, 023300 (2020)から引用)

- MNIST
  - データ：手書き数字0~9の画像（28x28ピクセル=784ピクセル）
  - FC2ネットワーク

No.	Layer name	Input size	Output size	Comment	N_{para}	Represented
1	FC	28×28	256		200704	Yes
	ReLu					
2	FC	256	10		2560	Yes
	Softmax					

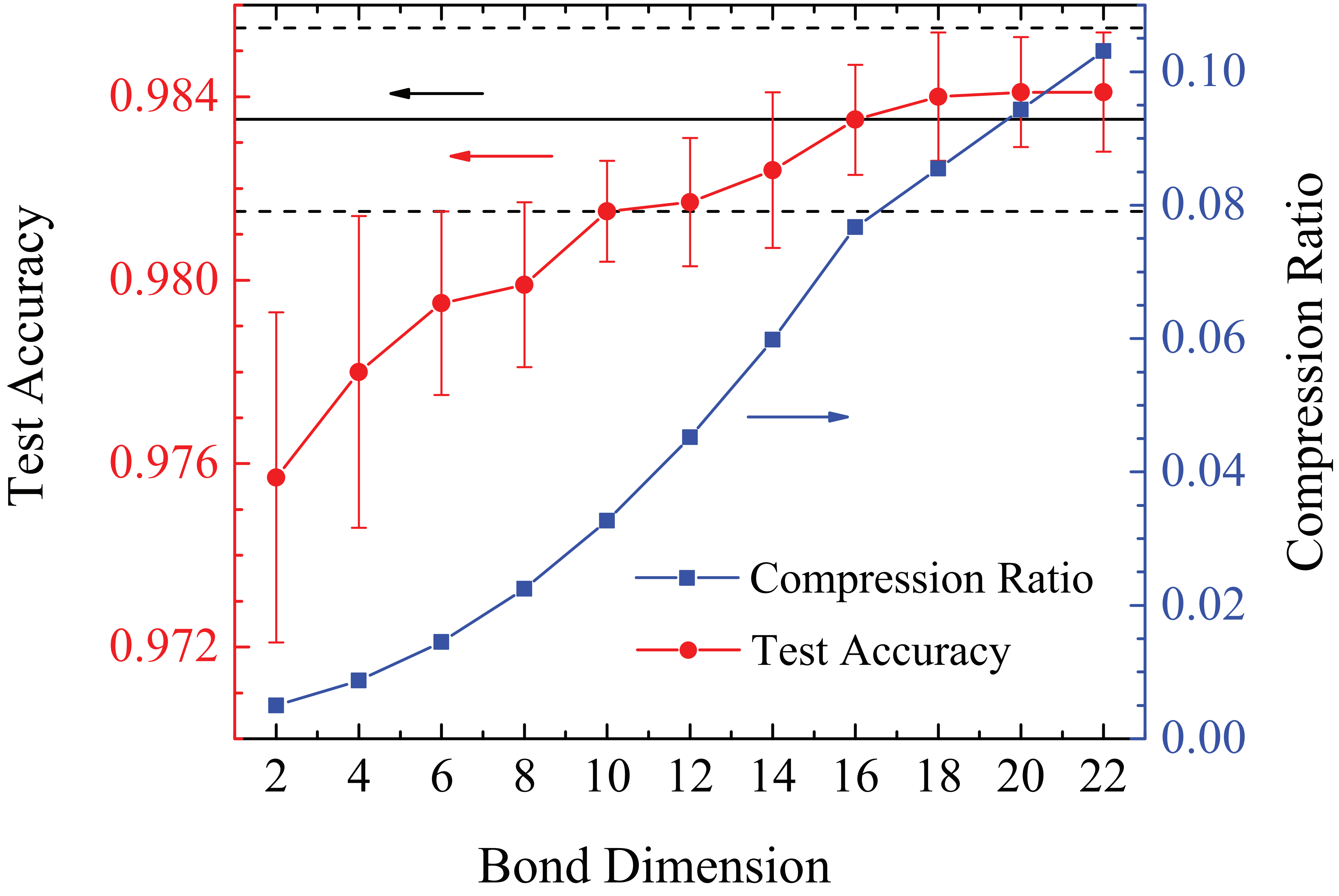
MPO化



第１層 $M^{4,4,4,4}_{4,7,7,4}(\chi = D)$

第２層 $M^{1,1,10,1}_{4,4,4,4}(\chi = 4)$

圧縮率＝MPOのパラメータ数 / 元のパラメータ数



局所的な相関が効率の良い
圧縮を可能とする

テンソル化の影響

(表はPRR, vol.2, 023300 (2020)から引用)

- LeNet5ネットワーク(CNN)

No.	Layer name	Input size	Output size	Comment	N_{para}	Represented
1	Conv	28×28	28×28×6	[5,5;6;1]	150	
	ReLu					
	MaxPo	28×28×6	14×14×6	[2,2;2]		
2	Conv	14×14×6	10×10×16	$[5,5;16;1]_{np}$	2400	
	ReLu					
	MaxPo	10×10×16	5×5×16	[2,2;2]		
3	Conv	5×5×16	120	$[5,5;120;1]_{np}$	48000	Yes
	ReLu					
4	FC	120	84		10080	Yes
	ReLu					
5	FC	84	10		840	Yes
	Softmax					

MPO化

$$M^{2,5,6,2}_{2,10,10,2}(4), M^{2,3,7,2}_{2,5,6,2}(4), M^{1,5,2,1}_{2,3,7,2}(2)$$

圧縮率 ~ 0.05

- CIFAR-16：10種類の画像データ（32x32RGB)
- VGG16ネットワーク

No.	Layer name	Input size	Output size	Comment	N_{para}	Represented
1	ConvUnit	32×32×3	32×32×64	2×[3,3;64;1]	38592	
	ReLu					
	MaxPo	32×32×64	16×16×64	[2,2;2]		
2	ConvUnit	16×16×64	16×16×128	2×[3,3;128;1]	221184	
	ReLu					
	MaxPo	16×16×128	8×8×128	[2,2;2]		
3	ConvUnit	8×8×128	8×8×256	2×[3,3;256;1]	884736	
	ReLu					
4	Conv	8×8×256	8×8×256	[1,1;256;1]	65536	
	ReLu					
	MaxPo	8×8×256	4×4×256	[2,2;2]		
5	ConvUnit	4×4×256	4×4×512	2×[3,3;512;1]	3538944	
	ReLu					
6	Conv	4×4×512	4×4×512	[1,1;512;1]	262144	
	ReLu					
	MaxPo	4×4×512	2×2×512	[2,2;2]		
7	ConvUnit	2×2×512	2×2×512	2×[3,3;512;1]	4718592	Yes
	ReLu					
8	Conv	2×2×512	2×2×512	[1,1;512;1]	262144	
	ReLu					
	MaxPo	2×2×512	512	[2,2;2]		
9	FC	512	4096		2097152	Yes
	ReLu					
10	FC	4096	4096		16777216	Yes
	ReLu					
11	FC	4096	10		40960	Yes
	Softmax					

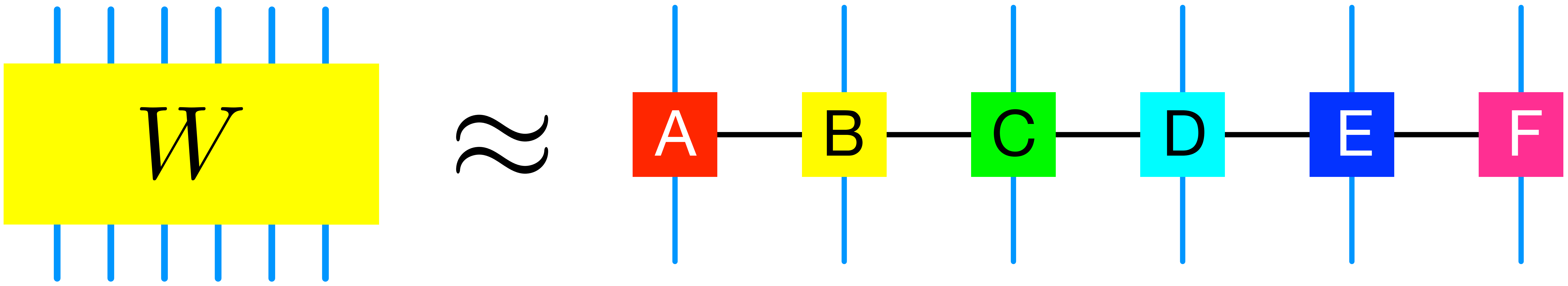
結果

Data set	Network	Original Rep 精度 a (%)	MPO-Net 精度 a (%)	MPO-Net 圧縮率 ρ
MNIST	LeNet-5	99.17 ± 0.04	99.17 ± 0.08	0.05
CIFAR-10	VGG-16	93.13 ± 0.39	93.76 ± 0.16	~0.0005
	VGG-19	93.36 ± 0.26	93.80 ± 0.09	~0.0005

ここまでのまとめ

- テンソルネットワーク（MPO）を用いた重み行列の圧縮
  - 低い計算コスト
  - 既存のニューラルネットワークに適用可能
    テストしたネットワーク：**FC2, VGG, ResNet, DenseNet**
  - MNIST, CIFAR-10, Fashion-MNISTなどで高い圧縮率(0.1~0.005)

A. Novikov, D. Podoprikhin, A. Osokin, and D. Vetrov, “Tensorizing Neural Networks,” NIPS 2016.
Z.-F. Gao, et al., “Compressing deep neural networks by matrix product operators,” Phys. Rev. Research, vol.2, 023300 (2020).



圧縮はどのように行われているのか

- MPOを用いて重み行列を圧縮
  - なぜ圧縮できるのか
  - どこまで圧縮できるのか
  - 何に依存するのか

➡ MPO中の有効な成分を定量的に評価したい

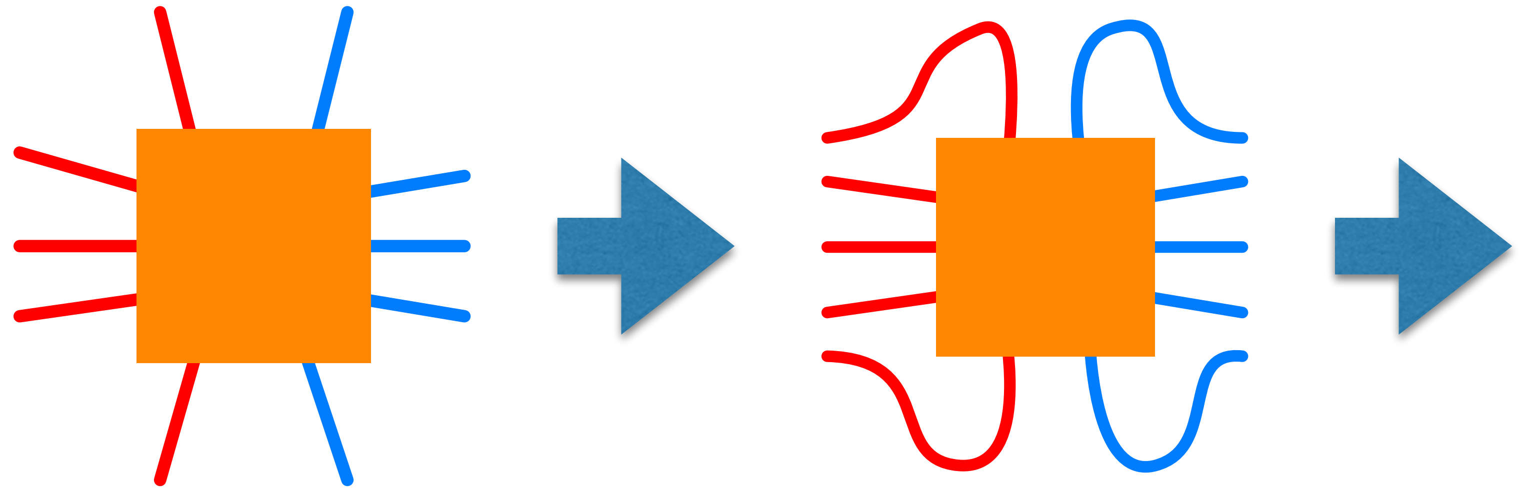
➡ **エンタングルメントを調べてみる**

日本物理学会２０２１年秋季大会
講演(22pL4-9) "行列積表現を用いたニューラルネットワークのエンタングルメント解析"

サポート　産学連携：京大トヨタ「モビリティ基盤数理」
本研究に関するグループメンバー：阿蘇品，真鍋，赤松

テンソルネットワークにおけるエンタングルメントの上限

- テンソルのエンタングルメントエントロピー



添字を２グループに分ける　　行列とみなす

$$\begin{pmatrix} \lambda_1 & 0 & \cdots & & \\ 0 & \lambda_2 & 0 & \cdots & \\ \cdots & 0 & \lambda_3 & 0 & \cdots \\ & & \cdots & & \end{pmatrix}$$

特異値

$$p_i = \frac{\lambda_i^2}{Z} \quad \left(Z = \sum_i \lambda_i^2 \to \sum_i p_i = 1 \right)$$

$$E.E = -\sum p_i \ln(p_i)$$

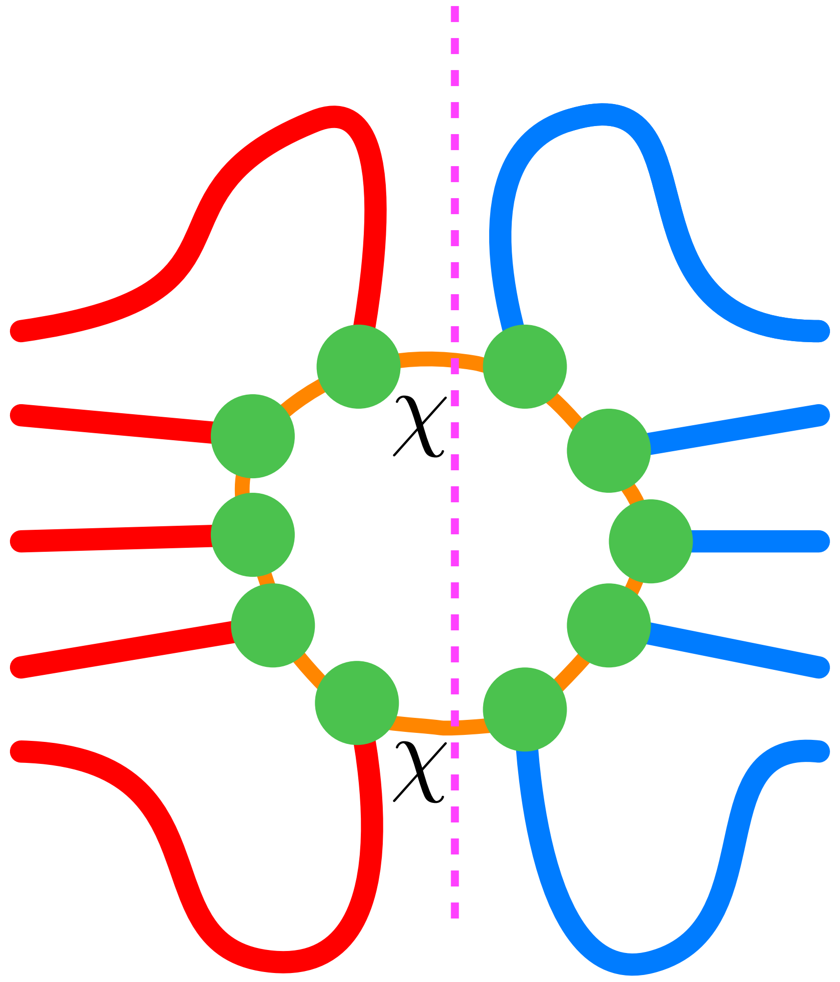
k個の特異値がある場合は最大 $\ln(k)$

どれぐらい２グループの添字に関係があるかを表す

- テンソルネットワークでの上限

エンタングルメントエントロピーの上限は，ネットワークを２つに分ける際にカットする枝の最小自由度の対数

例.

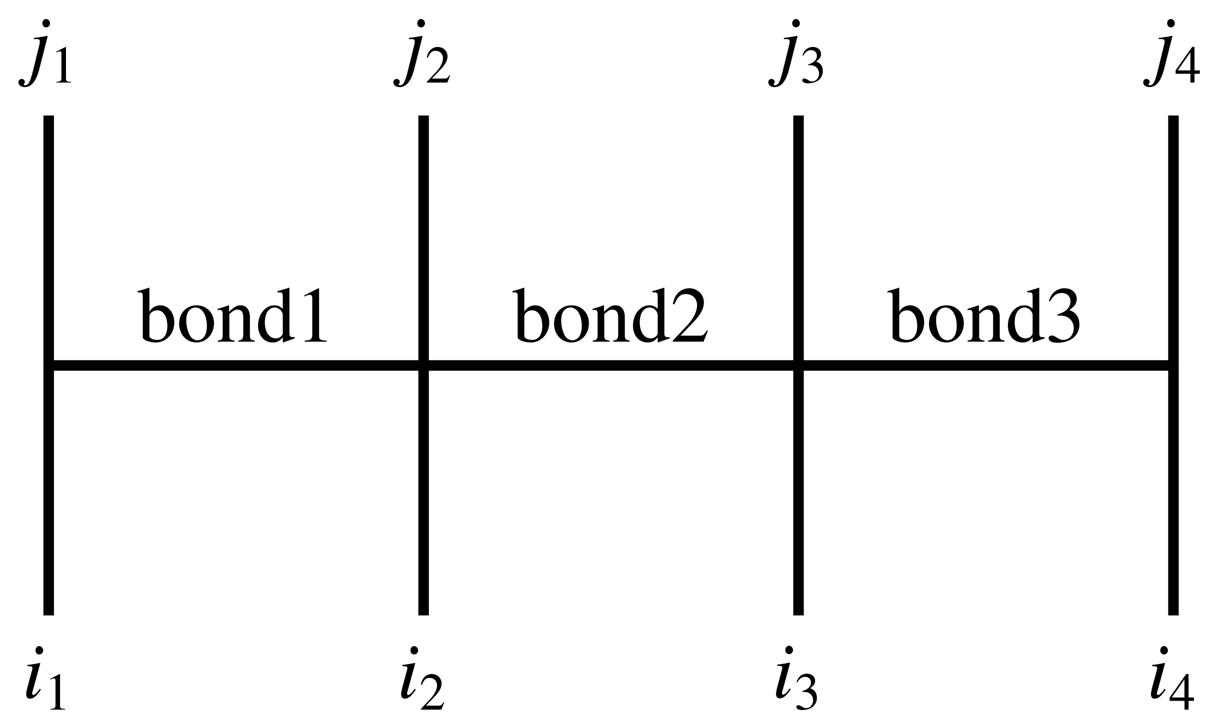


$$E.E. \leq \ln(\chi^2) = 2\ln(\chi)$$

E.E.がこの上限に比べて小さければ
枝（ボンド）の自由度を小さくできる

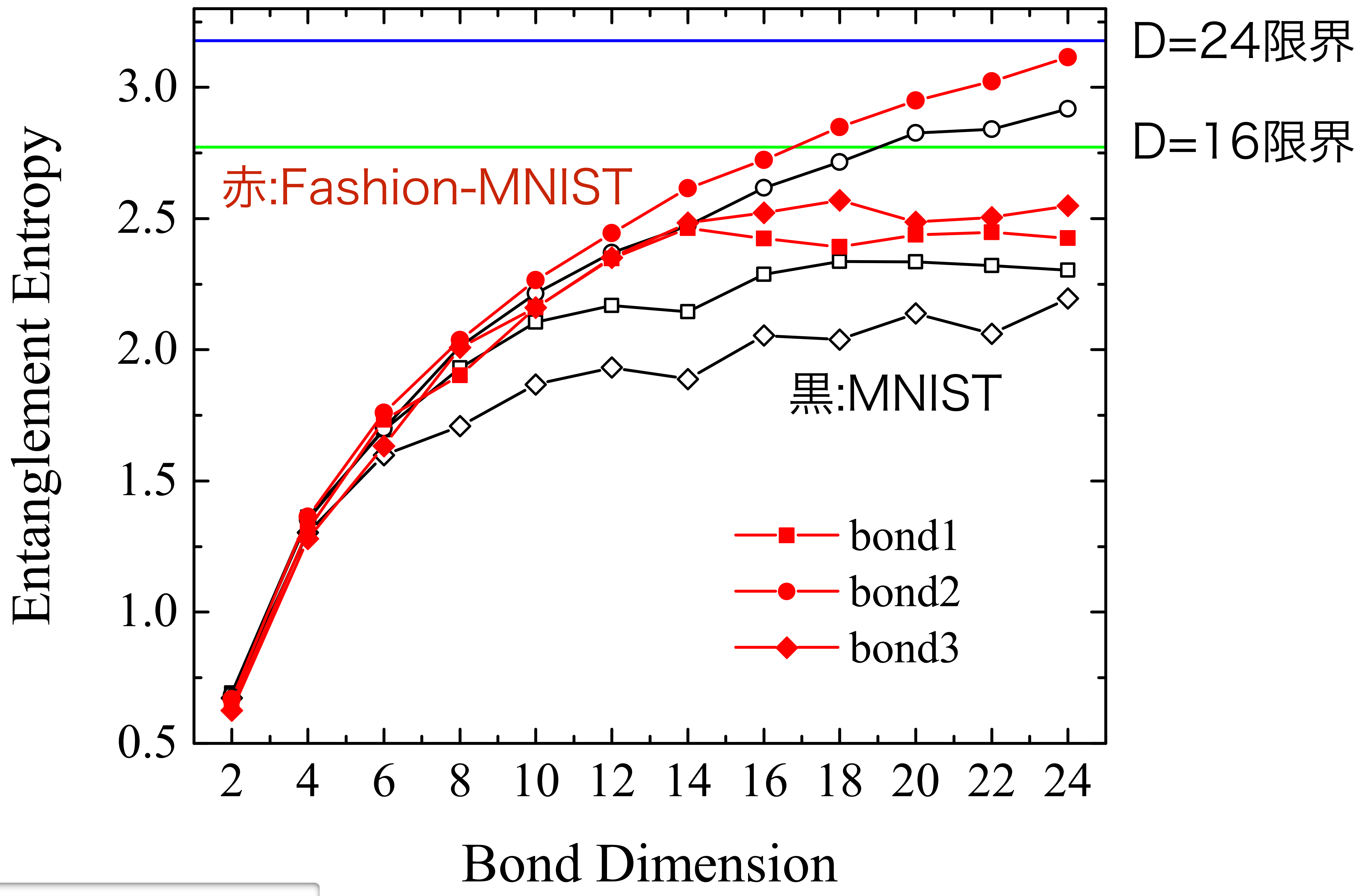
エンタングメントエントロピーによるMPO圧縮の評価

FC2第１層　$M^{4,4,4,4}_{4,7,7,4}(\chi = D)$



先行研究

(図，表はPRR, vol.2, 023300 (2020)から引用)



難しい問題では、エンタングルメント
エントロピーが収束していない

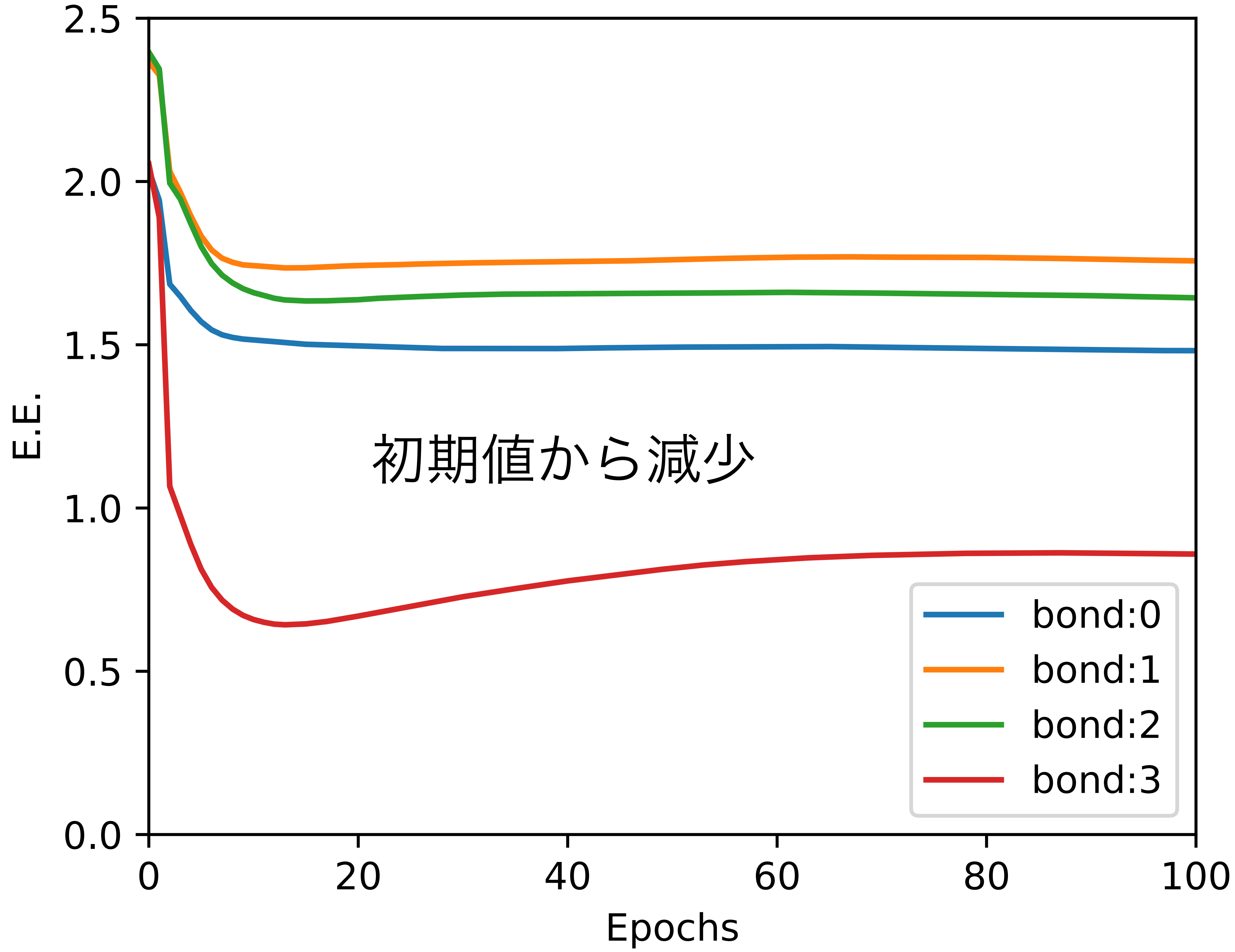
（注）Fashion-MNIST: 28x28ピクセルの
10種類の画像データセットで、FC2による
精度は90%程度

エンタングルメントエントロピーの時間変化

モデル

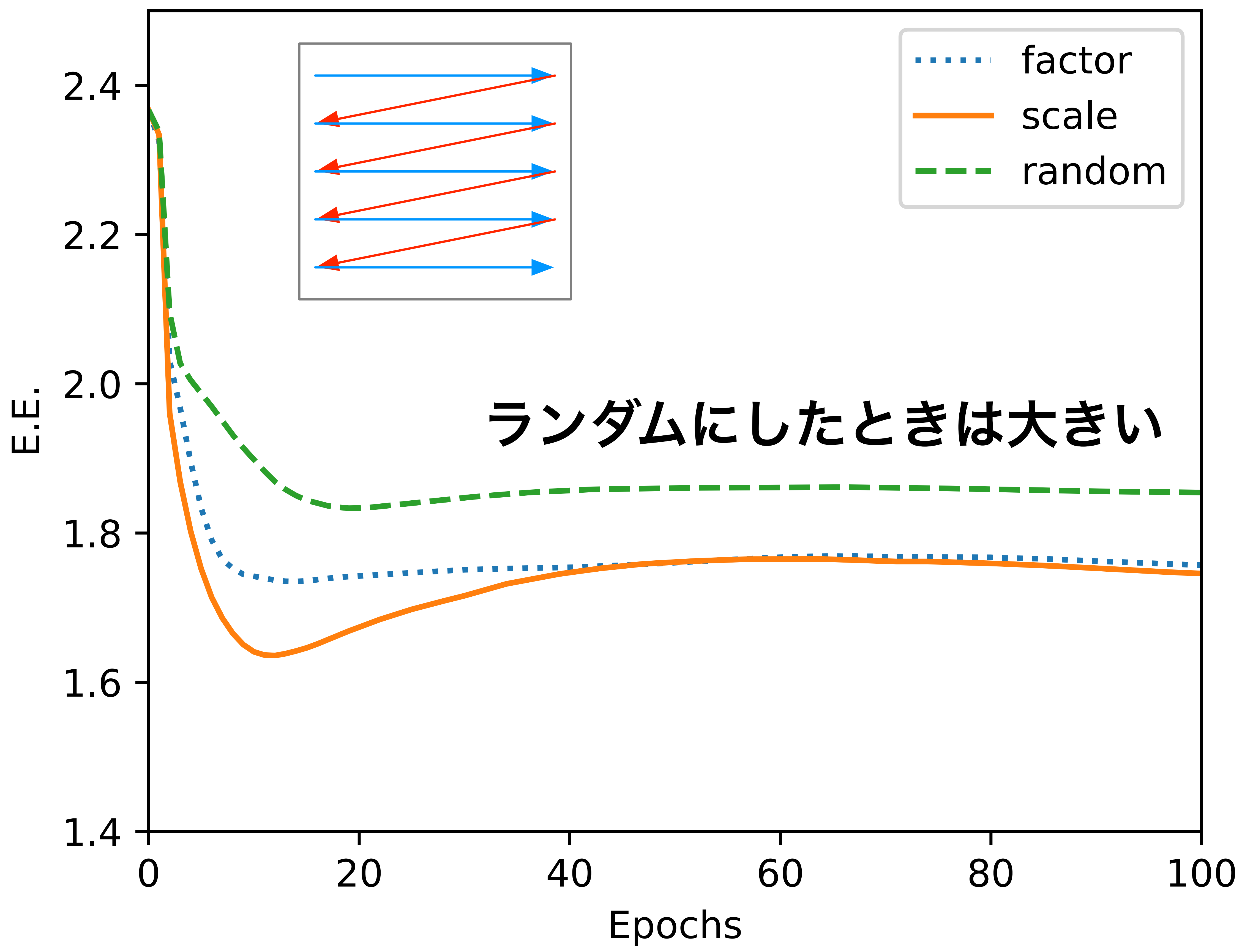
MPO	4,4,4,4,4 → 4,4,4,4,4, χ=12
ReLU	
FC	1024 → 10
Softmax	

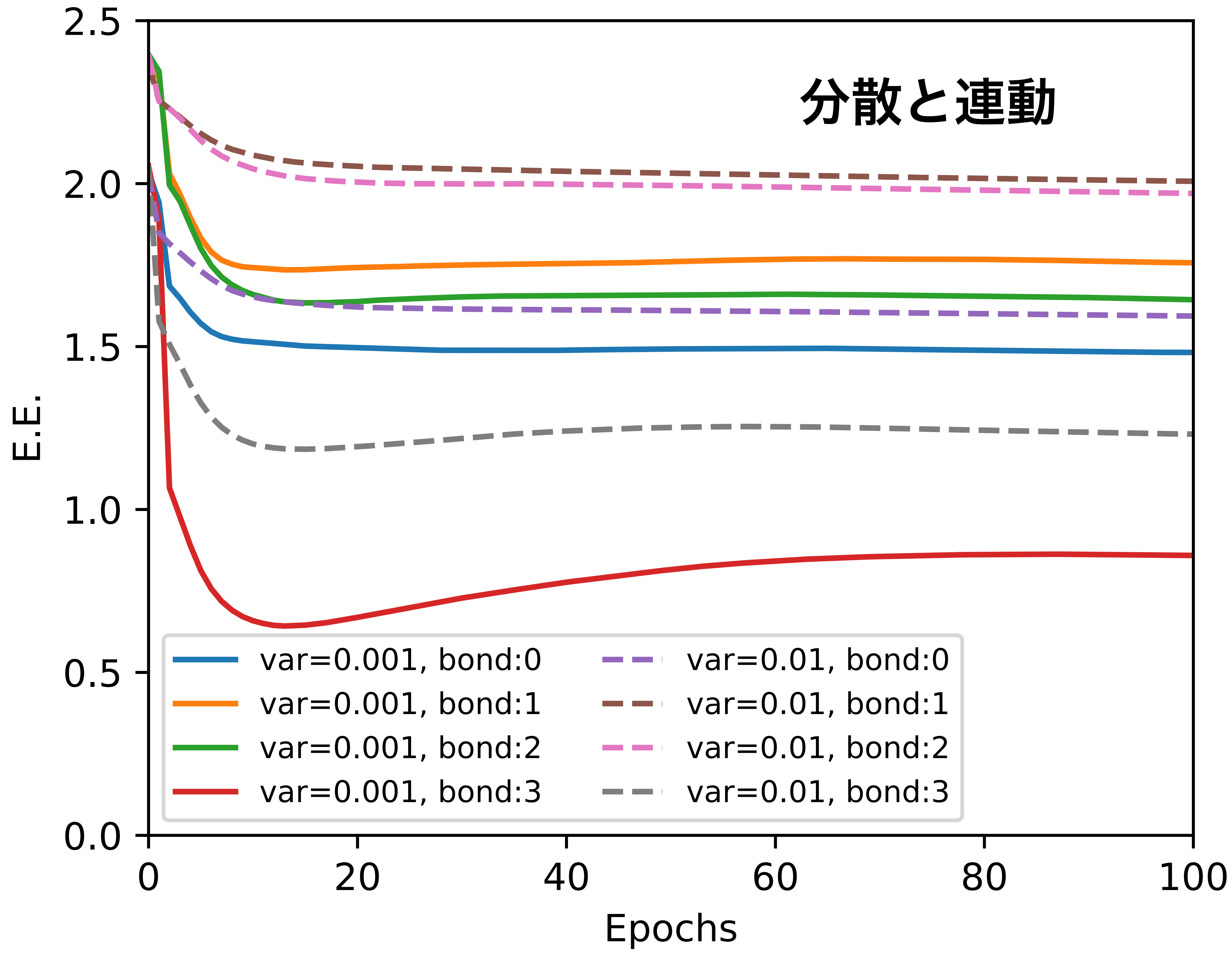
学習データ: MNIST (32x32)



学習をするだけで，重み行列の情報圧縮が起きている

テンソル化・初期分散の影響





重み行列の情報圧縮はテンソル化の仕方や初期分散にも影響を受ける

Normalizationの導入

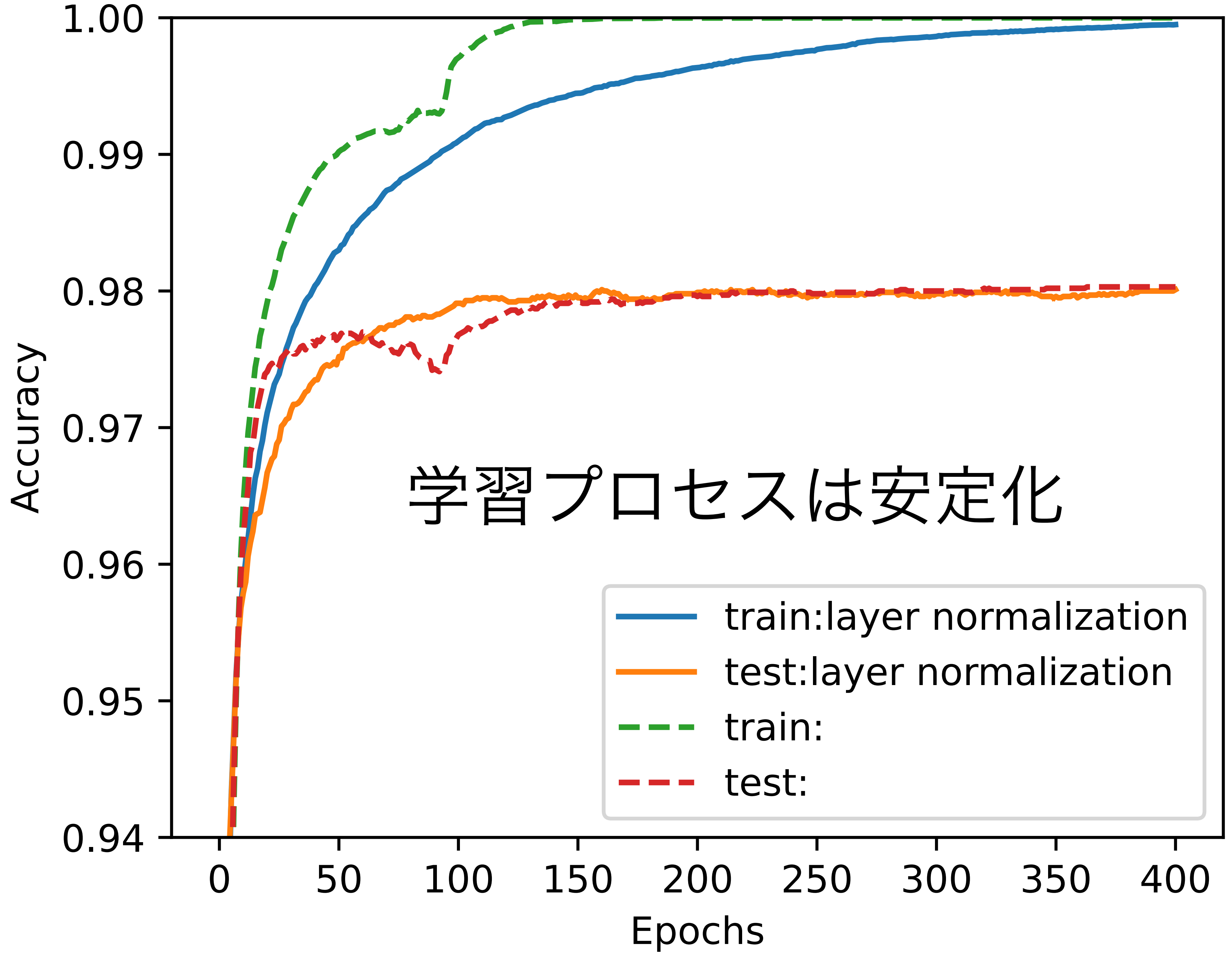
- Layer normalization　　Ba, Kiros, Hinton (2016)
  - レイヤー中のニューロンのバラツキを正規化

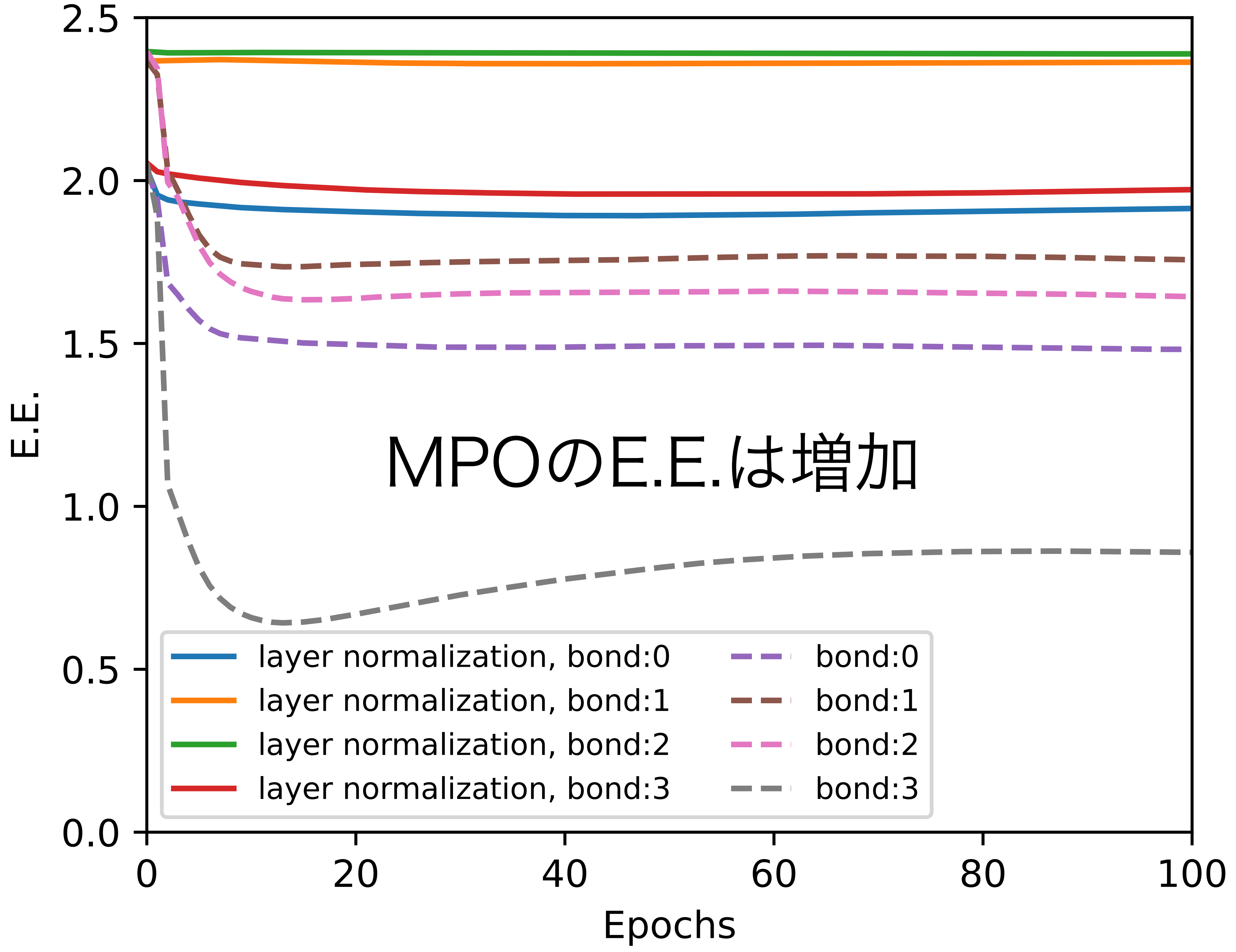
$$\tilde{x}_i = \frac{x_i - \mathrm{mean}(\mathbf{x})}{\sqrt{\mathrm{var}(\mathbf{x})}}$$

モデル

MPO	4,4,4,4,4 → 4,4,4,4,4, χ=12
Layer normalization	
ReLU	
FC	1024 → 10
Softmax	

学習データ: MNIST (32x32)





汎化能力とMPOのE.E.の関係は単純ではない

量子機械学習とは？

- 量子力学にインスパイアされた機械学習
  - 量子状態空間

    重ね合わせ： $|\psi\rangle = T_{000\cdots}|000\cdots\rangle + \cdots + T_{111\cdots}|111\cdots\rangle$

    非常に高次元

    N quibitでは，2^N次元

    テンソルネットワーク：量子振幅Tを近似

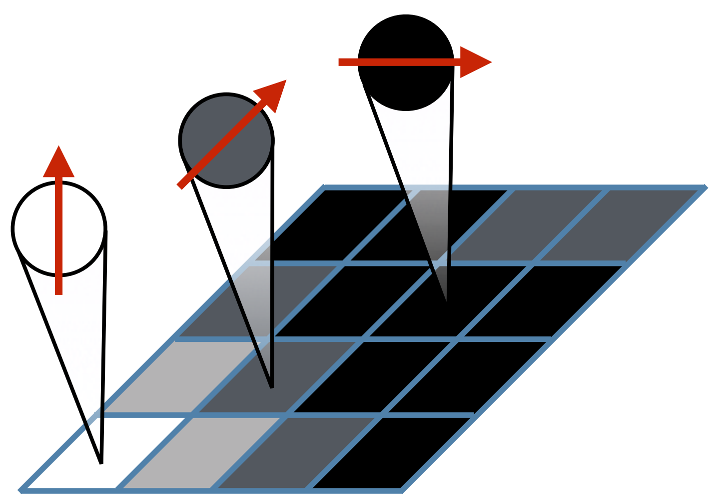
  - 高次元性の利用

    データを量子状態空間にマップして線形回帰

    E. Stoudenmire and D. J. Schwab,
    “Supervised Learning with Tensor Networks,” NIPS 2016.

テンソルネットワーク分類器

- アルゴリズム
  - 古典データを量子状態に変換



$$\mathbf{x} = (x_i)_{i=1,\cdots,N}, \quad x_i \in [0,1] \quad \Rightarrow \quad |\Phi(\mathbf{x})\rangle = \prod_{i=1}^{N} \otimes |\phi(x_i)\rangle, \quad |\phi(x)\rangle = \sin\left(\frac{\pi}{2}x\right)|0\rangle + \cos\left(\frac{\pi}{2}x\right)|1\rangle = \begin{bmatrix} \sin\left(\frac{\pi}{2}x\right) \\ \cos\left(\frac{\pi}{2}x\right) \end{bmatrix}$$

  - 量子状態を線形変換し、成分が最大値を出力ラベルとする

$$|f\rangle = W|\Phi(\mathbf{x})\rangle \quad \Rightarrow \quad \arg\max_{l} |f^l|$$

- 特徴
  - 高次元（2^N）空間から特徴を抽出
  - ラベル空間への線形変換Wをテンソルネットワークを用いて圧縮

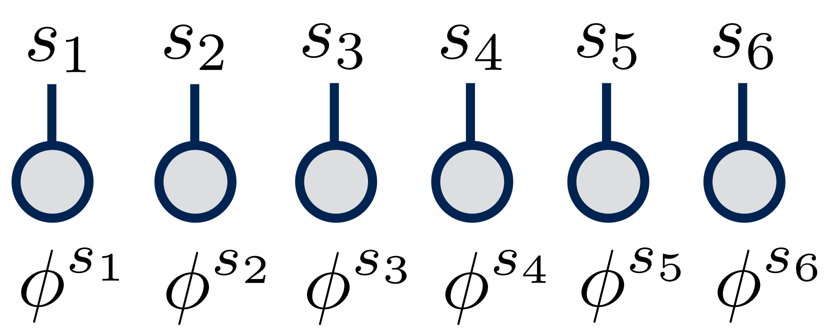
E. Stoudenmire and D. J. Schwab, “Supervised Learning with Tensor Networks,” NIPS 2016.

アルゴリズムのテンソルネットワーク表現

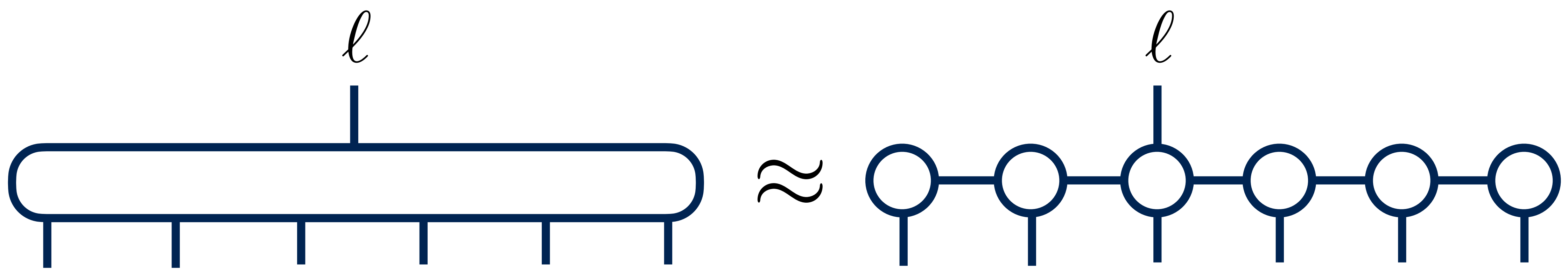
- 入力データ

$$|\Phi(\mathbf{x})\rangle = \prod_{i=1}^{N} \otimes |\phi(x_i)\rangle, \quad |\phi(x)\rangle = \sin\left(\frac{\pi}{2}x\right)|0\rangle + \cos\left(\frac{\pi}{2}x\right)|1\rangle = \begin{bmatrix} \sin\left(\frac{\pi}{2}x\right) \\ \cos\left(\frac{\pi}{2}x\right) \end{bmatrix}$$

$\Phi =$

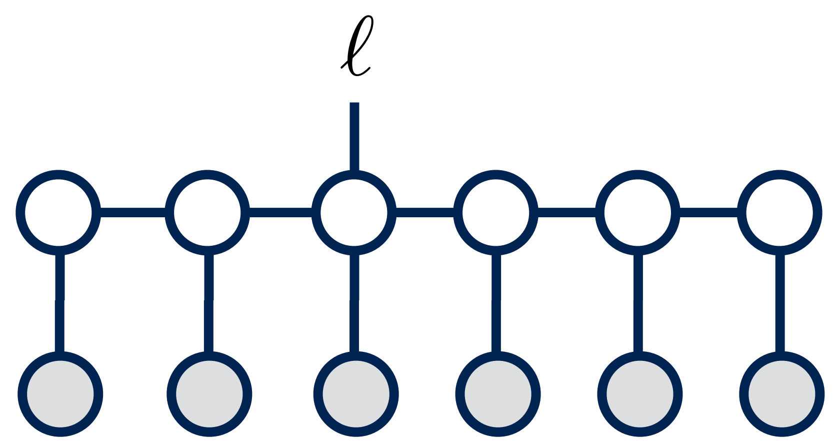


- 線形変換W



- ラベル計算

$f^l =$



$$\mathbf{f} = \langle W | \Psi(\mathbf{x}) \rangle$$

$$f^l = W_{l0\ldots0} \sin\left(\frac{\pi}{2}x_1\right) \cdots \sin\left(\frac{\pi}{2}x_N\right)$$
$$+ \cdots + W_{l1\ldots1} \cos\left(\frac{\pi}{2}x_1\right) \cdots \cos\left(\frac{\pi}{2}x_N\right)$$

2^N項

E. Stoudenmire and D. J. Schwab,
“Supervised Learning with Tensor Networks,”
NIPS 2016.

MNISTへの適用・特徴空間からの写像

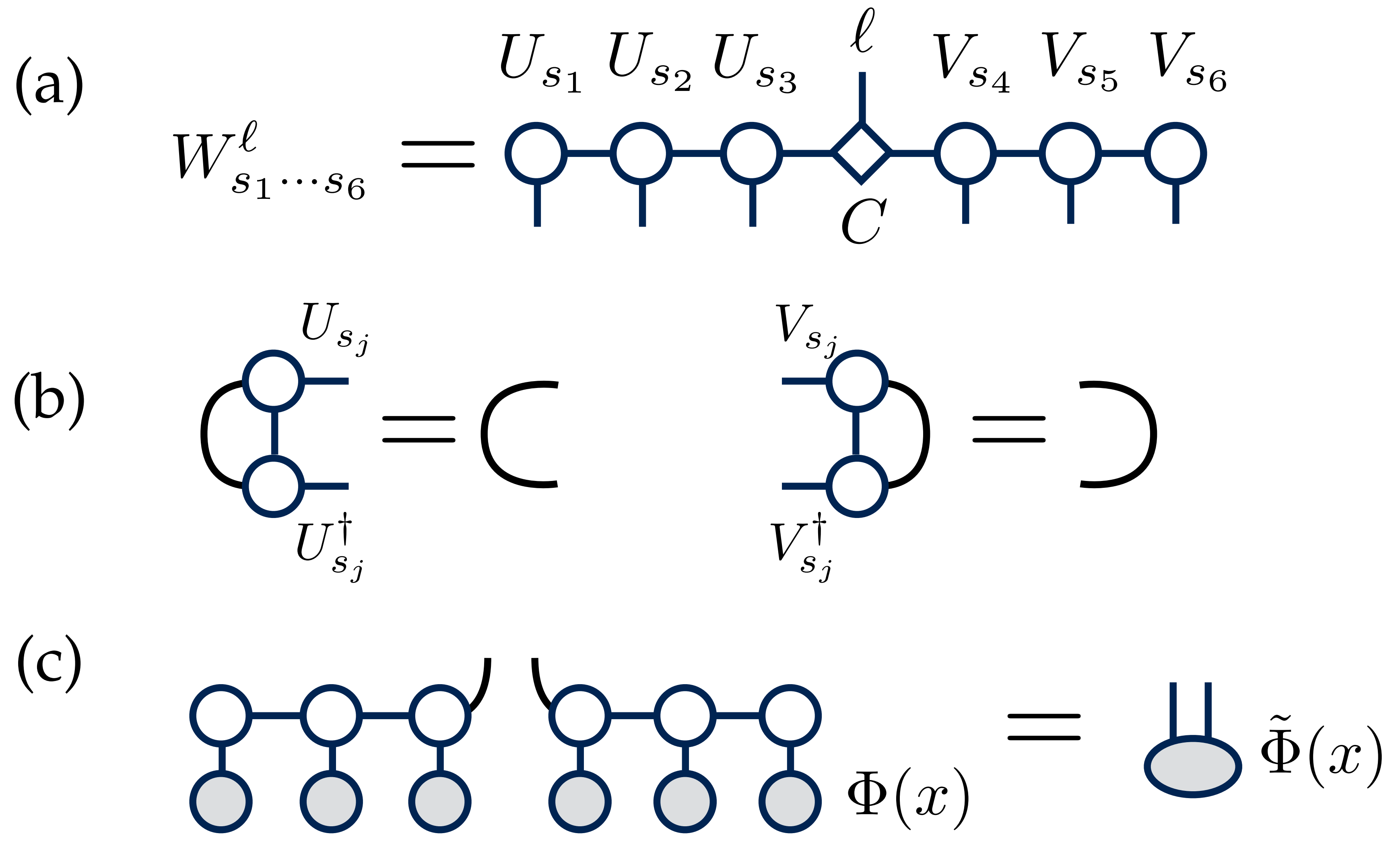
- コスト関数

$$C = \frac{1}{2} \sum_{n=1}^{N_T} \sum_{l} |f^l(\mathbf{x}_n) - y_n^l|^2$$

- テスト結果：28x28を14x14にしたMNIST

ボンド次元	10	20	120
エラー	5%	2%	0.97%

- 特徴空間からの射影



E. Stoudenmire and D. J. Schwab, “Supervised Learning with Tensor Networks,” NIPS 2016.

機械学習でのテンソルネットワークの活用

古典機械学習のテンソル圧縮
A. Novikov, D. Podoprikhin, A. Osokin, and D. Vetrov,
“Tensorizing Neural Networks,” NIPS 2016.
量子機械学習：テンソルネットワーク識別器
E. Stoudenmire and D. J. Schwab,
“Supervised Learning with Tensor Networks,” NIPS 2016.

- ニューラルネットワーク
  - 重み行列の圧縮 → **エンタングルメントとして定量化**
- 量子機械学習
  - 線形変換の圧縮
- 展望
  - 学習後のテンソルネットワークはデータの特性を反映

    テンソルネットワークの解析を通じて学習過程を理解

入門書：雑誌 **数理科学**２０２２年２月号「テンソルネットワークの進展」