Weight Matrices Compression Based on PDB Model in Deep Neural Networks

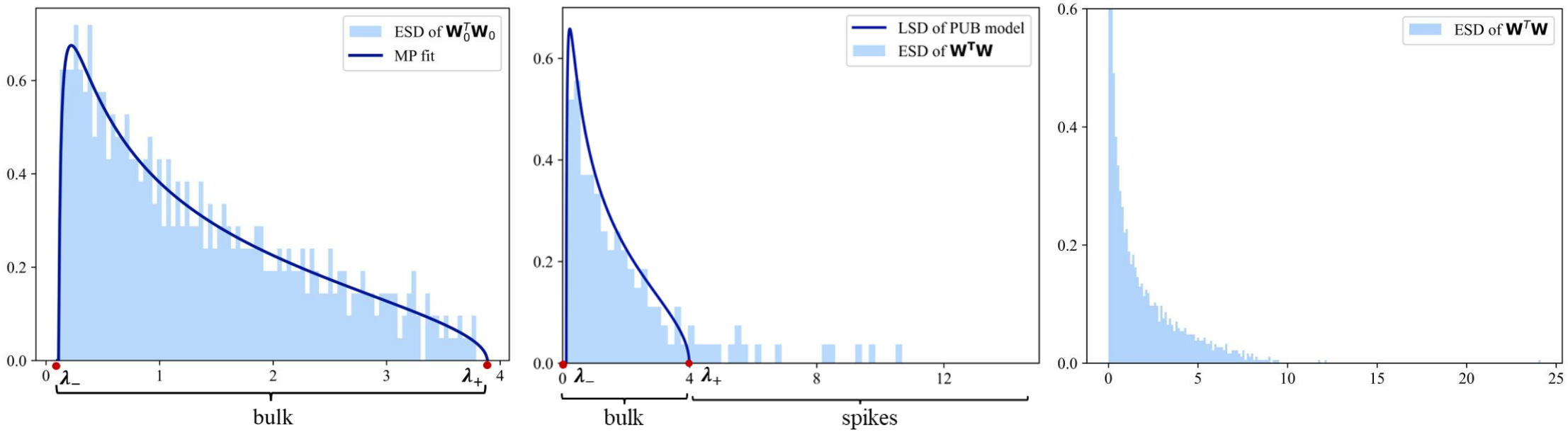
Xiaoling Wu Junpeng Zhu Zeng Li*

Department of Statistics and Data Science, Southern University of Science and Technology

Content

- Motivation
- Methodology
- Algorithms
- Experiment

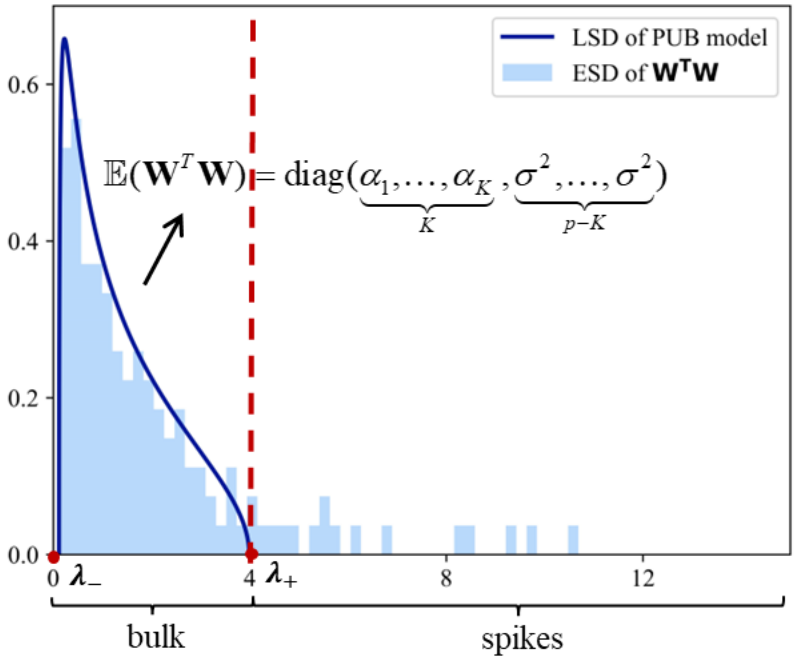
Eigenvalue phase transition of weight matrices during training



(a) Initial (b) Bulk+Spikes (c) Heavy-tailed

- **Initialization**: $\mathbb{E}\boldsymbol{W}_0^T\boldsymbol{W}_0 = \sigma_0^2\boldsymbol{I}_p$.
- **Bulk+Spikes phase**: Exist λ_{bulk} and λ_{spike}.
- **Heavy-tailed phase**: Not very common.

Existing method: Population Unit Bulk (PUB) model in Bulk+Spikes phase



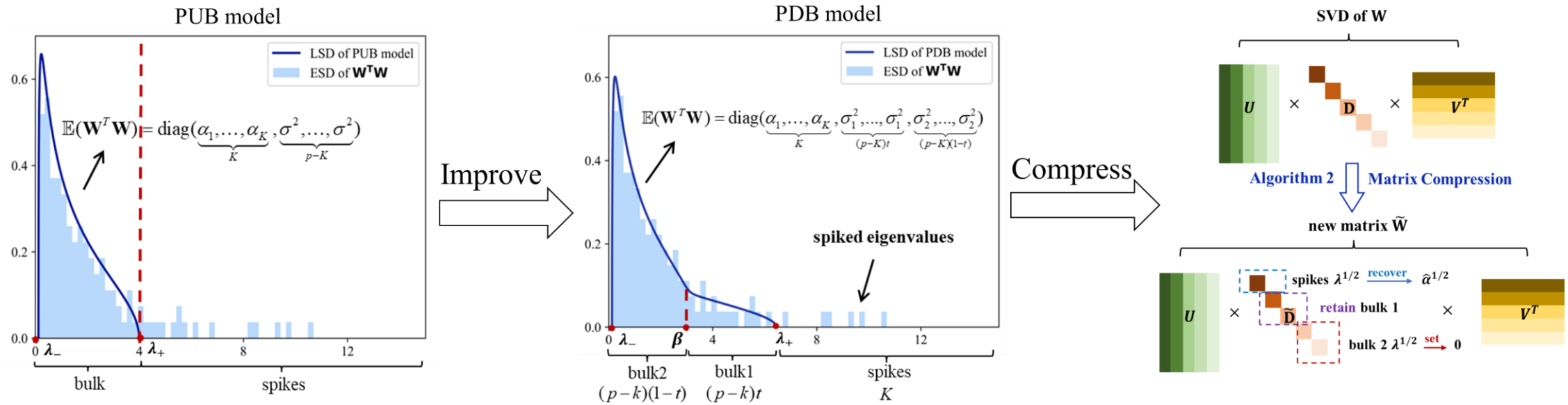
Before training

$$\mathbb{E}\mathbf{W}_0^T\mathbf{W}_0 = \sigma_0^2\mathbf{I}_p \quad \Longrightarrow \quad \text{After training}$$

$$\boldsymbol{\Sigma}_{PUB} = \mathbb{E}\mathbf{W}^T\mathbf{W} = \mathrm{diag}(\underbrace{\alpha_1,\ldots,\alpha_K}_{\substack{K \\ \text{spikes}}},\underbrace{\sigma^2,\ldots,\sigma^2}_{\substack{p-K \\ \text{bulk}}})$$

- PUB requires a homogeneous population variance.
- PUB fails to accurately capture the ESD of $\boldsymbol{W}^T\boldsymbol{W}$.

Proposed method: Population Double Bulk (PDB) model in Bulk+Spikes phase



Before training

$$\mathbb{E}\,\mathbf{W}_0^T\mathbf{W}_0 = \sigma_0^2\mathbf{I}_p$$

PUB model

PDB model

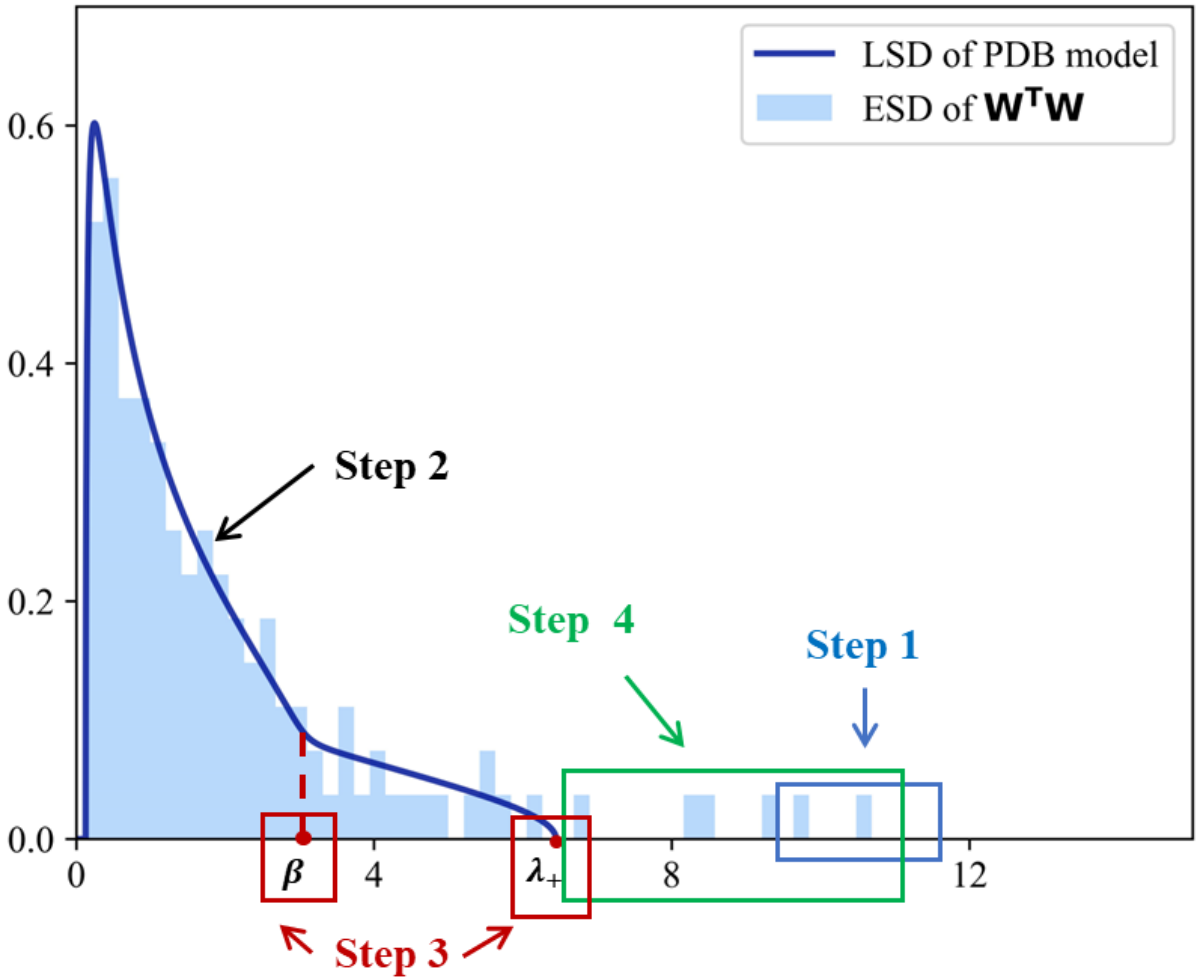
After training

$$\mathbb{E}(\mathbf{W}^T\mathbf{W}) = \mathrm{diag}(\underbrace{\alpha_1,\ldots,\alpha_K}_{K\text{ spikes}},\underbrace{\sigma^2,\ldots,\sigma^2}_{p-K\text{ bulk}})$$

$$\mathbb{E}(\mathbf{W}^T\mathbf{W}) = \mathrm{diag}(\underbrace{\alpha_1,\ldots,\alpha_K}_{K\text{ spikes}},\underbrace{\sigma_1^2,\ldots,\sigma_1^2}_{(p-K)t\text{ bulk 1}},\underbrace{\sigma_2^2,\ldots,\sigma_2^2}_{(p-K)(1-t)\text{ bulk 2}})$$

Population Double Bulk Least Squares (PDBLS) algorithm for model estimation

Estimate $\Theta_{bulk} = \{\sigma_1^2, \sigma_2^2, t\}$, $\Theta_{spike} = \{K, \alpha_1, \ldots, \alpha_K\}$ and $\Theta_{bound} = \{\lambda_+, \beta\}$.



Estimate K_0

↓

Estimate Θ_{bulk}

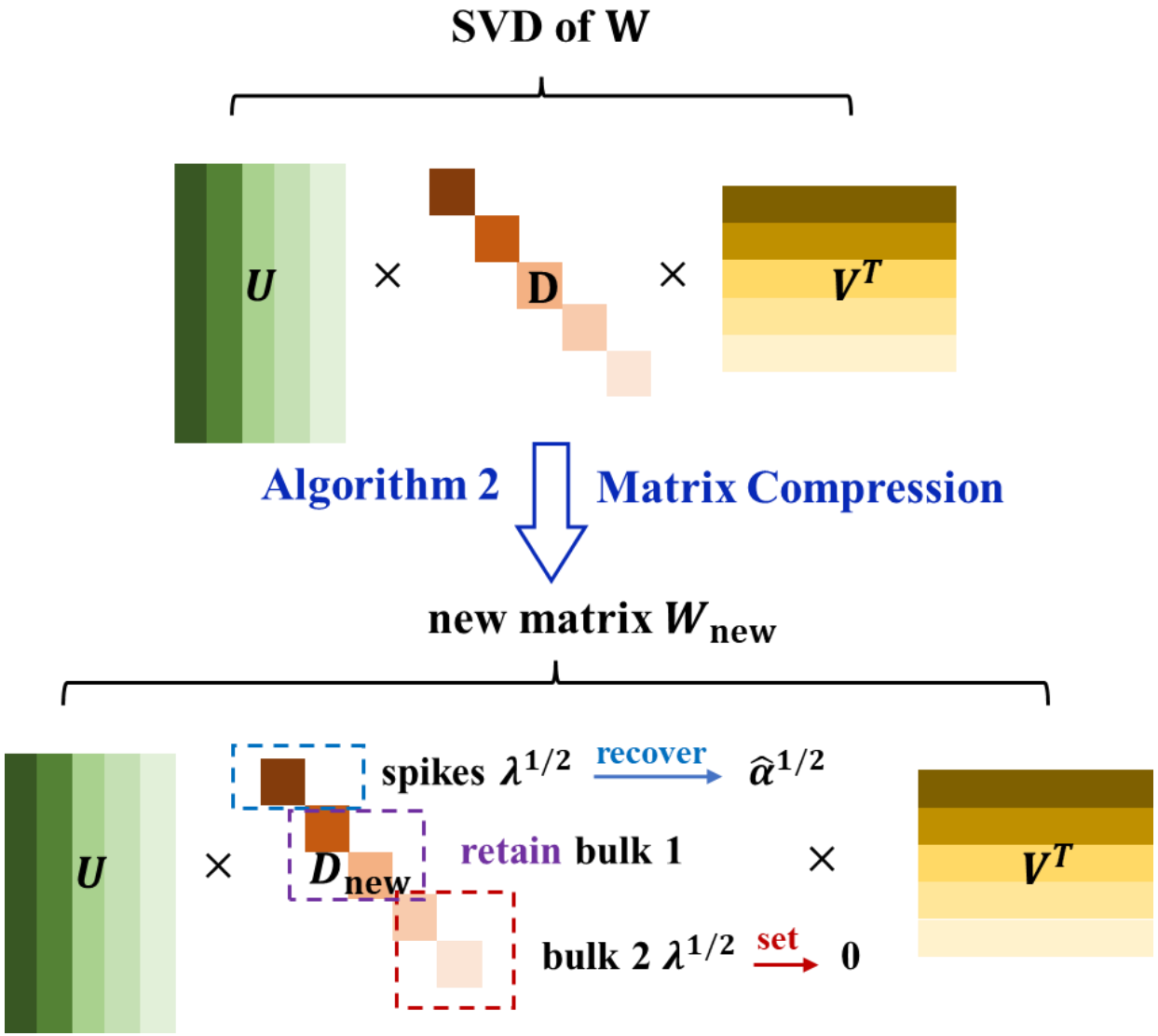
↓

Estimate Θ_{bound} and Refine K

↓

Estimate Θ_{spike}

PDB Noise-filtering algorithm for matrix compression



Experiment Design

Network architectures

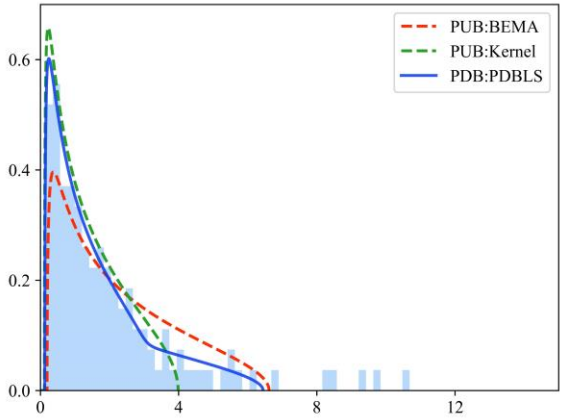
1. Fully Connected Neural Network (FCNN)
   - Dataset: MNIST
2. Convolutional Neural Network
   - Datasets: CIFAR10 and ImageNet
   - Networks: ResNet18 and VGG16
3. Large model
   - Languge
     - Datasets: RTE and SciTail
     - Networks: BERT and T5-base
   - Vision
     - Datasets: DTD and SUN397
     - Networks: ViT-L

Comparable Methods

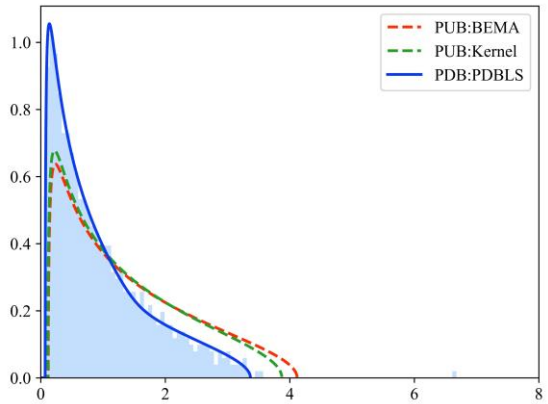
1. PUB-based
   - Bulk Eigenvalue Matching Analysis (BEMA):
   - Kernel Estimation (Kernel)
2. SVD-based
   - Sparse low rank (SLR)
   - Naïve SVD

Fitting Performance

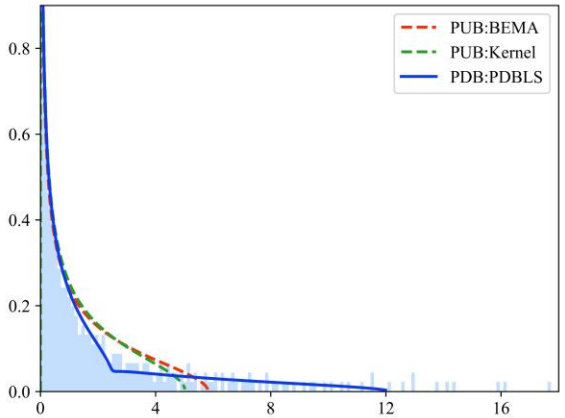
Curve fitting



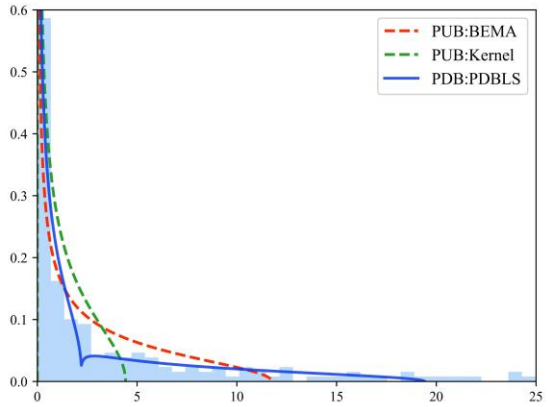
(a) FCNN on MNIST



(b) VGG16 on CIFAR10



(c) ResNet18 on ImageNet



(d) ResNet18 on CIFAR10

Moment alignment

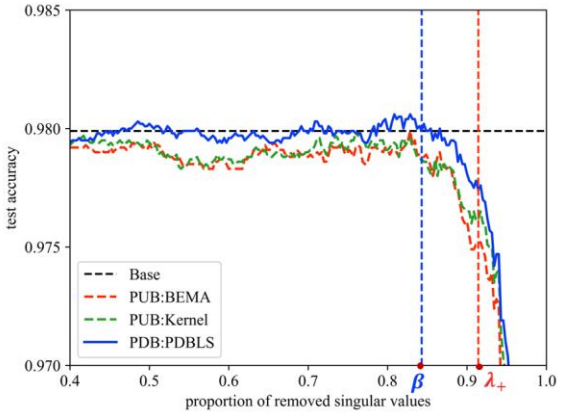
Table 3. Comparison of theoretical and empirical spectral moments for FCNN and VGG16, $\hat{\gamma}_j = \frac{1}{p}\,\mathrm{tr}\left(\mathbf{W}^T\mathbf{W}\right)^j, j = 1, 2, 3.$

		FCNN:MNIST			VGG16:CIFAR10		
Model	Method	γ_1	γ_2	γ_3	γ_1	γ_2	γ_3
PUB	BEMA	2.27	7.74	32.23	0.92	1.26	2.12
PUB	Kernel	1.37	2.81	7.04	0.90	1.22	2.03
PDB	**PDBLS**	**1.71**	**4.94**	**18.79**	**0.96**	**1.53**	**3.11**
empirical $\hat{\gamma}_j$		1.74	5.50	24.17	0.97	1.59	3.51

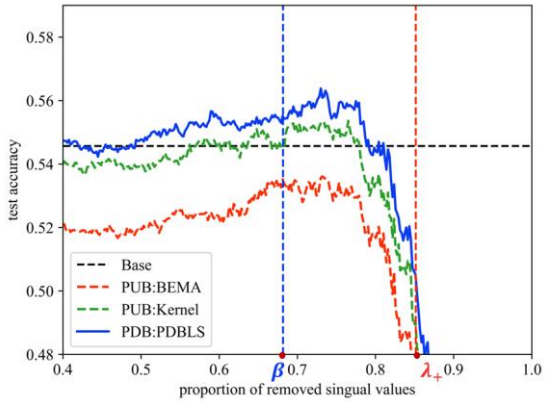
Table 5. Comparison of theoretical and empirical spectral moments for T5-base and BERT.

		T5-base: RTE			BERT: SCITAIL		
Model	Method	γ_1	γ_2	γ_3	γ_1	γ_2	γ_3
PUB	BEMA	0.67	0.90	1.51	0.59	0.69	1.02
PUB	Kernel	0.53	0.56	0.74	0.56	0.63	0.88
PDB	**PDBLS**	**0.77**	**1.55**	**4.17**	**0.67**	**1.18**	**2.78**
empirical $\hat{\gamma}_j$		0.72	1.83	5.35	0.71	1.38	3.95

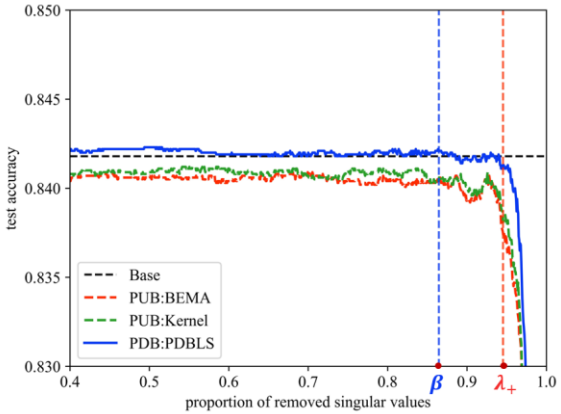
Generalization and compression performance



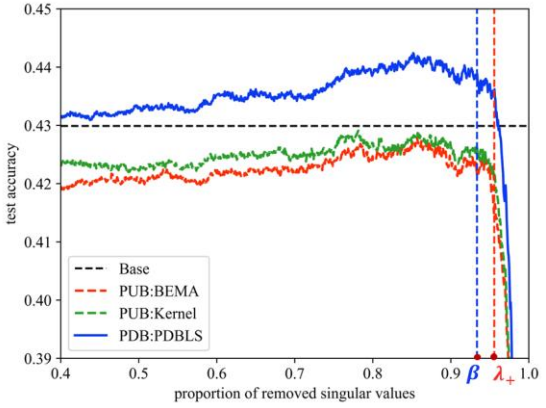
(a) FCNN



(b) FCNN with noise



(c) Vgg16



(d) Vgg16 with noise

The test accuracy obtained by training different network models of different data sets (0% noise).

Network	Datasets	**Base**	PDB	PUB	SLR	naive SVD
FCNN	MNIST	0.9799	**0.9804**	0.9791	0.9799	0.9799
ResNet18	CIFAR10	0.8349	**0.8384**	0.8338	0.8357	0.8354
VGG16	CIFAR10	0.8418	**0.8422**	0.8405	0.8415	0.8419
BERT	RTE	0.7029	**0.7319**	0.7174	0.7246	0.7029
	SciTail	0.9055	**0.9155**	0.9130	0.9008	0.9055
T5-base	RTE	0.7174	**0.7536**	0.7319	0.7174	0.7174
	SciTail	0.9025	**0.9243**	0.9167	0.9182	0.9196
VIT-L	DTD	0.7452	**0.7533**	0.7482	-	0.7405
	SUN397	0.7680	**0.7771**	0.7720	-	0.7716
Average		0.7783	**0.7891**	0.7827	0.7858	0.7789

Contributions

- **Population Double Bulk (PDB) model**: more accurately captures the empirical distribution of eigenvalues.
- **Population Double Bulk Least Squares (PDBLS) algorithm**: estimate the parameters of PDB model and establish a boundary between noise and information.
- **PDB Noise-Filtering algorithm**: compress the weight matrix by removing noisy eigenvalues and recover information of spikes.

Thanks!