Verification of the Effectiveness of Deep Learning in Preprocessing Parameter Estimation for the Conjugate Gradient Method

Takamasa Nakaya[1], Takahiro KATAGIRI[2], Tetsuya HOSHINO[2], Daichi MUKUNOKI[2], Masatoshi KAWAI[3]
1:Graduate School of Informatics, Nagoya University, 2:Information Technology Center, Nagoya University, 3:Cyberscience Center, Tohoku University

Background and research objectives

- Increased value of optimal parameter settings through numerical simulation
  - Utilizing deep learning for automatic parameter optimization
- Related studies:
  - Direct derivation of preprocessing matrices using deep learning[1]
  - Prediction of single-precision and double-precision switching
- The investigations using deep learning models could not be confirmed.
  - Selection of storage precision **considering mixed precision**
  - **the impact of selection performance** based on model prediction capability

✓ Deep learning-based automatic parameter tuning [2] was implemented.
✓ Target problem: ICTCG method

A) Effectiveness for model parameter selection:
  - Storage accuracy selection including mixed precision

B) Model improvement and effective usage conditions:
  - Impact of input parameters and model performance

ICTCG method

- A method that improves the effectiveness of IC decomposition by introducing parameters such as threshold and max fill-in level, used as a preprocessing step for the CG method.
- An example of the decomposition for $\boldsymbol{A} \approx U^t DU$, $\boldsymbol{A}, \boldsymbol{D}, \boldsymbol{U} = \mathbb{R}^{n\times m}$ is shown .

$\boldsymbol{a_{i,j}}$: the (i, j)−th element of A, $\boldsymbol{d_{i,i}}$: the (i, i)−th element of D, $\boldsymbol{u_{i,j}}$: the (i, j)−th element of U, $\boldsymbol{f_{i,j}}$: fill-in level of $u_{i,j}$, $\boldsymbol{t}$:threshold, $\boldsymbol{m}$:max fill-in level

$$d_{i,i} = a_{i,i} - \sum_{k=1}^{i-1} u_{k,i} d_{k,k} u_{k,j}, \quad f_{i,j} = \begin{cases} 0, & a_{i,j} \neq 0 \\ \min\limits_{k=1\ldots i-1} \left(f_{i,k} + f_{k,i} + 1\right), & else \end{cases}$$

$$u_{i,j} = \begin{cases} d_{i,i}^{-1}\left(a_{i,j} - \sum\limits_{k=1}^{i-1} u_{k,i} d_{k,k} u_{k,j}\right), & f_{i,j} \le m \wedge |u_{i,j}| \ge t \\ 0, & else \end{cases}$$

Evaluation Settings and method

- λ: Thermal diffusion coefficient of the problem.
  ✓ The larger the value, the longer the computation time required.

Training data: Used for model building **Test data**: Used for model evaluation

– Data is acquired by performing actual calculations using the Flow Type I.
– Test data was configured in the λ range falls below, within, or above the training data.

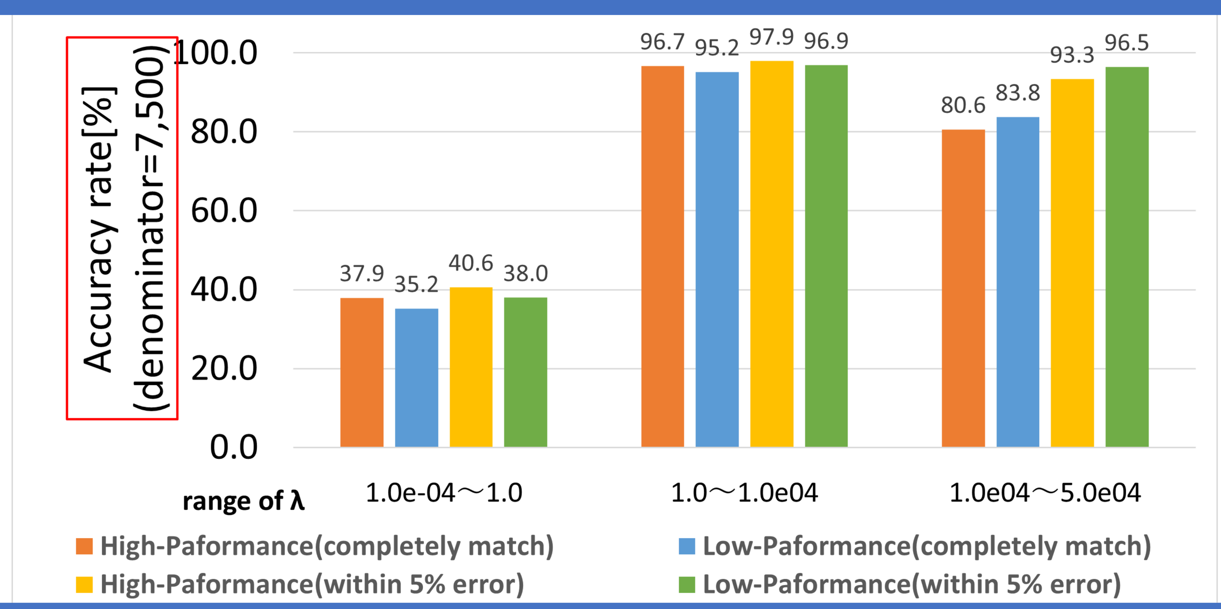
	Training Data	Test Data
Storage Precision of Matrices and Vectors	dd(only double), ss(only single), sd(Matrices：single, Vectors:double)	
Queue size	Square matrices of 128 × 128 × 128 for both rows and columns	
Condition number λ	1.0～1.0e5 ※**1**	1.0e-5～1.0, 1.0～1.0e5, 1.0e5～5.0e5 ※**2**
Threshold	1.0e-5～1.0 ※**1**	1.0e-5～1.0 ※**2**
Max fill-in level	0, 1, 2	
Convergence condition	Relative residuals are 1.0e-07 or less for both double and single	

- We constructed models using combinations of three batch sizes and four epoch counts and selected two of these combinations for use.
  ✓ High-performance model: Loss function is **minimized**
  ✓ Low-performance model: Loss function is **maximized**
- The model was generated using Keras on TensorFlow (ver. 2.4.1) by combining CNN and MLP.
- **Evaluation method**

I. Compare execution times for model output and actual calculations, obtain predicted and actual storage accuracy
II. Evaluate the model's storage accuracy selection performance based on prediction accuracy.

※**1**:100 values on a base-10 logarithmically spaced scale in λ (threshold) range
※**2**:50 values on a base-10 logarithmically spaced scale in each λ (threshold) range

Evaluation results



- **Expected Results**

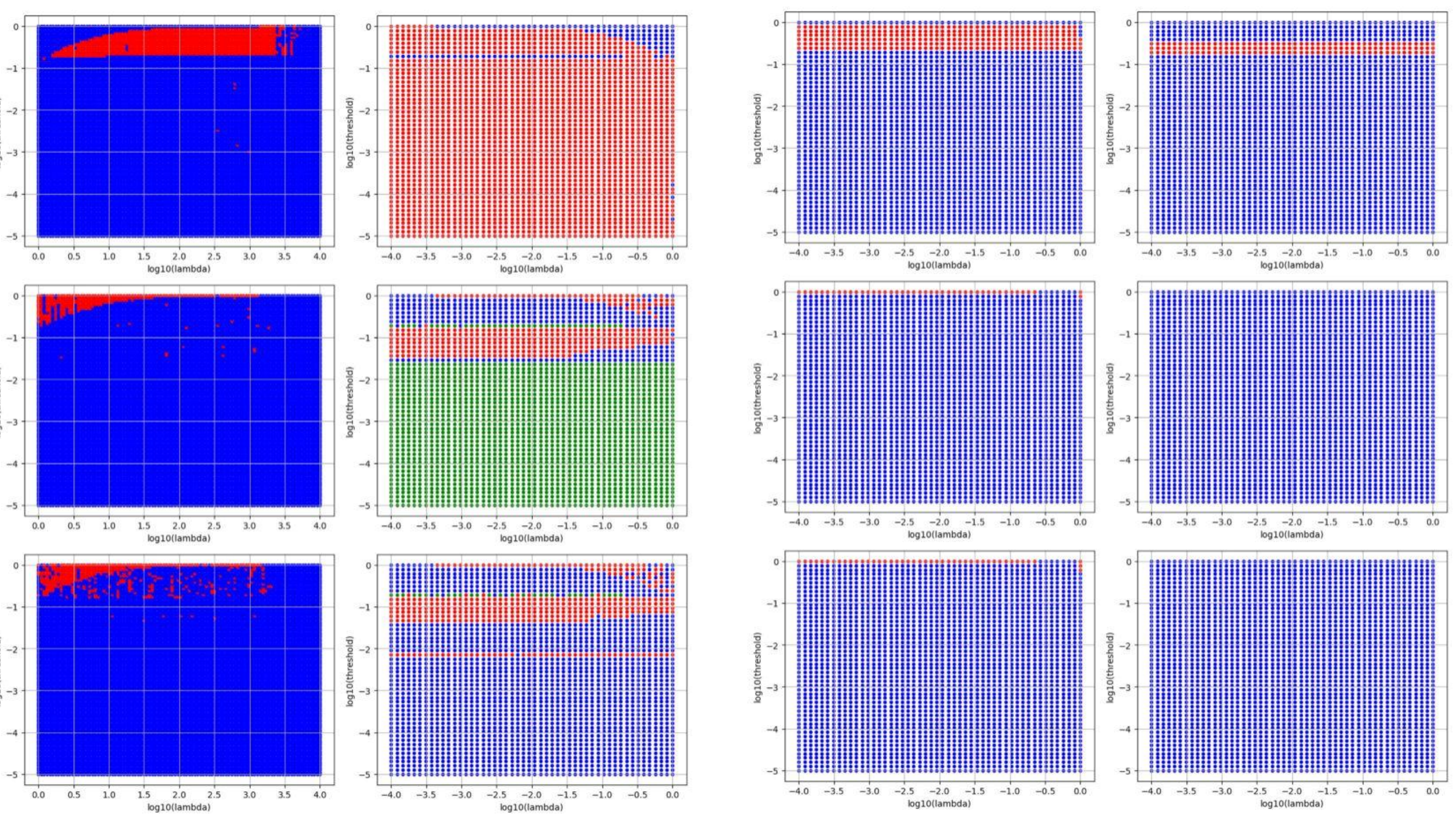
a. High accuracy is achieved only for λ within the training range.
b. Between the high-performance model and the low-performance model, the high-performance model achieves higher accuracy.

- **Evaluation Results**

A) The accuracy rate clearly decreased **only** within the λ range below the learning range.
B) When λ was larger than the learning range, **the low-performance model demonstrated better prediction performance**.

Analysis

A) The results of selecting storage precision for training data and λ values within the range 1.0e-04 to 1.0 are shown.

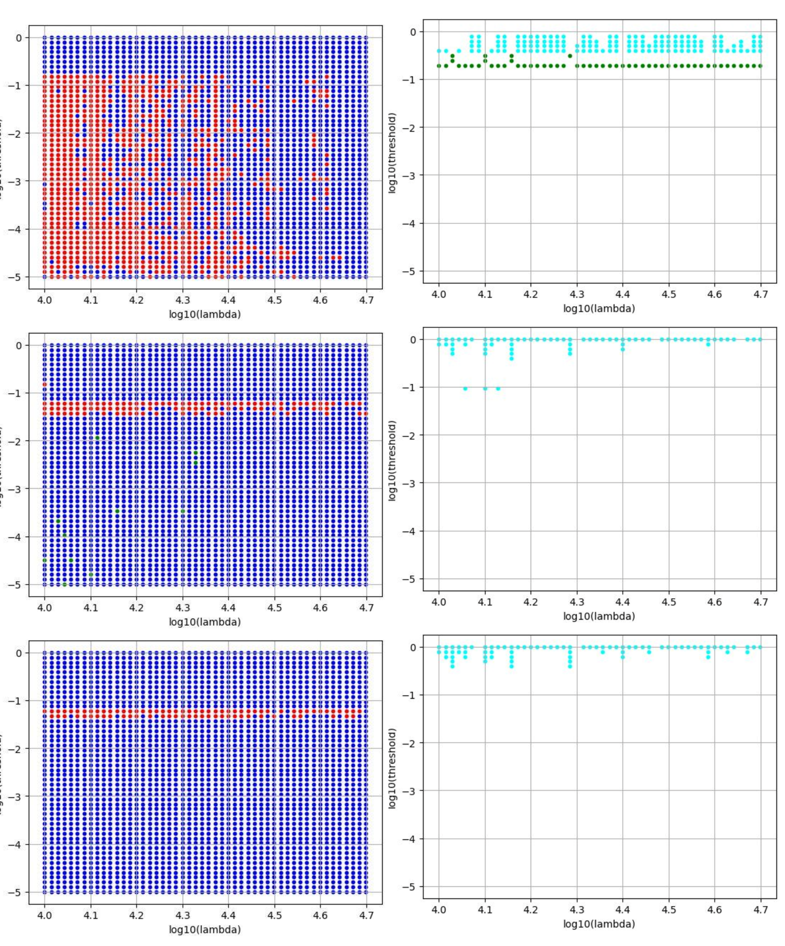


(1)Selection of storage precision in training data (2)Selection of storage precision in test data (3)High-Performance Model Prediction (4)Low-Performance Model Prediction

Horizontal axis: λ value, Vertical axis: Threshold, Top to bottom: Max fill-in levels 0, 1, 2. When λ value, threshold, and max fill-in level are fixed, the selected optimal storage accuracy is blue for **dd**, red for **ss**, and green for **sd**.

- The trends in (1) and (3), (4) are similar, while the trend in (2) differs.
  – Even across different λ ranges, the model strongly reflects the trend in the training data. This is the reason for the low accuracy rate.
- Within the range of λ from 1.0e04 to 5.0e04, **ss** was selected in most cases, showing a selection tendency similar to that observed in the learning range.
  – This explains why high accuracy is achieved when the λ range is large.

B) The results of selecting storage accuracy for actual values within the λ range of 1.0e04 to 5.0e04, along with the priority for model selection, are shown.



(5)Selection of storage precision in test data (6)Model Performance Difference

※"Completely match" > "within 5% error" > "Inappropriate" priority, with high > low colored green and high < low colored cyan.

- (6) shows that the low-performance model achieves good select at larger thresholds than the high-performance model.
- (5) indicates that **ss** tends to be selected when the threshold is large, but (1) shows that **ss** may also be selected in the training data.
- High performance may lead to overfitting, causing the model to incorrectly select **dd** instead of the intended **ss** in certain cases, which is thought to contribute to the decline in accuracy.

Summary and future work

- The proposed model and evaluation method achieved high accuracy in predicting storage precision within the learning range.
- The impact of model performance on parameter selection capability was inferred to be related to overfitting.
- **Future works**

i. Proposing method to handle a broader datasets with high predictive performance
ii. General applicability studies, such as those targeting issues beyond the ICTCG method

Acknowledgments

This research was supported by Grants-in-Aid for Scientific Research (JP23K11126, JP24K02945) from the Japan Society for the Promotion of Science (JSPS). It was also supported by the Interdisciplinary Large-Scale Information Infrastructure Joint Research Center (JHPCN) and the Innovative High-Performance Computing Infrastructure (HPCI) (Project Numbers: jh250015, jh250018).

[1]Aoki, S., Katagiri, T., Ohshima, S., Kawai, M., Nagai, T., & Hoshino, T.: Adaptation of XAI to Auto-tuning for Numerical Libraries, In 2024 IEEE 17th International Symposium on Embedded Multicore/Many-core Systems-on-Chip (MCSoC).(2024)