

## ICTCG 法の実行時間予測モデルに対する説明可能な AI の適用

中谷 崇真†, 河合 直聡‡, 片桐 孝洋‡, 星野 哲也‡, 永井 亨‡

名古屋大学情報学部コンピュータ科学科†

名古屋大学 情報基盤センター‡

## 1. はじめに

人工知能 (AI) が出力する答えを検証無しに利用することは、不適切な出力の導出時にそれを許容してしまうという点で危険性がある。そのため現状の AI では、導出した出力を人間が検証する必要がある。しかし AI モデルから得られる全出力は膨大であり、それらをすべて人間が確認することは現実的でない。よって AI 出力の検証工程の自動化や、検証工数の削減手法は必須である。そのため、AI モデルの構築コストを減らす研究開発が進められている。

この一方で青木ら[1]は、数値計算分野における性能チューニング工数の削減を目的とした、ソフトウェア自動チューニング (Software Auto-tuning, AT) 技術 [2] へ AI を適用した場合における、AI が出力する予測時間に関する説明性について検証を行い、有用性を示した。

本研究では、典型的な数値計算アプリケーションとして閾値付き不完全コレスキー分解前処理付き共役勾配法 (ICTCG 法) を取り上げる。その性能チューニングに AI を適用を行い、得られる AI モデルに対して XAI による検証をすることを目的とする。

## 2. 説明可能な AI (XAI)

## 2.1 概要

AI 出力の説明性を高めることで信頼性の高い AI とするツールは、説明可能な AI (Explainable AI, XAI) と呼ばれ、いくつかのツールが提案されている。そのツールの中には、オープンソース化され、使用が容易なものも存在する。以下では本論文で使用する XAI ツールである SHAP について説明する。

## 2.2 SHAP

SHAP (SHapley Additive exPlanations) [3] は協力ゲーム理論のシャープレイ値 (Shapley Value) を機械学習に応用したものである。そのため、採用されている評価基準に理論的な妥当性がある。厳密なシャ-

ープレイ値の計算は計算量が高いことから、近似的に算出する手法が研究されている。

SHAP は各特徴量が出力に与えた影響を説明として提示するため、全体的な傾向を説明するのに使われる、大域説明ツールである。

## 3. 閾値付き不完全コレスキー分解前処理付き共役勾配法 (ICTCG 法)

## 3.1 概要

不完全コレスキー分解 (Incomplete Cholesky Decomposition, IC) は、疎な正定値対称行列  $A$  を係数とする連立一次方程式  $Ax = b$  ( $A \in \mathbb{R}^{n \times n}$ ,  $x, b \in \mathbb{R}^n$ ) を解く反復法の 1 つである共役勾配法 (Conjugate Gradient (CG) Method) の前処理として活用される。

行列  $A$  に対して、IC を用いて式(1)の分解をすることを考える。

$$A = U^t D U + R \quad (U, D \in \mathbb{R}^{n \times n}) \quad \dots(1)$$

ここで、 $U$ : 上三角行列、 $D$ : 対角行列、 $R$ :  $A$  と IC 分解後の  $U^t D U$  との差の行列、とする。このとき、行列の要素値が 0 であった  $A$  の要素が、分解行列  $U$  において非ゼロ要素となる場合がある。これを fill-in と呼ぶ。

## 3.2 閾値付き IC 分解前処理とアルゴリズム

閾値付き IC 分解 (Incomplete Cholesky Decomposition with Threshold, ICT)[4] とは、式(1)の行列分解過程で、分解行列  $U$  の非ゼロ要素について、fill-in レベルを持つようにしたものである。fill-in レベルとは、非ゼロ要素をどこまで許容するかのレベルである。

本研究では、我々が開発した実装[5]を利用する。この実装では、最大 fill-in レベル ( $m$ )、および閾値 ( $t$ ) を設定する。これらは、fill-in のレベルが  $m$  以下、値が  $t$  以上の場合、fill-in の発生を許容することを示す。

以上のアルゴリズムの概略を、図-1 に記載する。ここで、 $a_{i,j}$ :  $A$  の  $i, j$  要素、 $d_{i,i}$ :  $D$  の  $i, i$  要素、 $u_{i,j}$ :  $U$  の  $i, j$  要素、 $f_{i,j}$ :  $u_{i,j}$  の fill-in レベル、 $t$ : 閾値、 $m$ : 最大 fill-in レベル、である。

Application of Explainable AI to predictive model for execution time of ICTCG

† Takamasa Nakaya, Department of Computer Science, School of Informatics, Nagoya University

‡ Masatoshi Kawai, Takahiro Katagiri, Tetsuya Hoshino, Toru Nagai, Information Technology Center, Nagoya University

$$d_{i,j} = a_{i,j} - \sum_{k=1}^{i-1} u_{i,k} d_{i,k} u_{k,j}$$

$$f_{i,j} = \begin{cases} 0, & a_{i,j} \neq 0 \\ f_{i,k} + f_{k,i} + 1, & \text{else} \end{cases}$$

$$u_{i,j} = \begin{cases} d_{i,j}^{-1} (a_{i,j} - \sum_{k=1}^{i-1} u_{i,k} d_{i,k} u_{k,j}), & f_{i,j} \leq m \wedge |u_{i,j}| \geq t \\ 0, & \text{else} \end{cases}$$

図-1 閾値付き IC 前処理法の概要

## 4. 性能評価

### 4.1 問題設定

ICTCG 法の実装での行列・ベクトルの浮動小数点数型はともに倍精度または単精度を採用する。また、 $t$ (閾値)および $m$ (最大 fill-in レベル)の 2 つを性能パラメタに設定する。この場合における、実行時間の予想を AI モデルで構築することで、AT を行う場合の検証を行った。本研究では SHAP 値が小さいほど実行時間が短く、望ましい結果になっていることを示す。

係数行列のサイズ  $64 \times 64 \times 64$ , 最大 fill-in レベルを 0, 1, 2, 閾値の範囲を 0.001~0.02 (0.001 間隔), 行列の条件数を 1.0d~9.9d (0.1d 間隔) とし教師データを取得し、回帰モデルを作成した。

### 4.2 評価環境

名古屋大学情報基盤センター設置のスーパーコンピュータ「不老」Type I サブシステム[6]を AI の教師データ取得用に利用した。また、「不老」Type II サブシステム[6]の GPU を機械学習用に利用した。

Python は ver. 3.6.13, Tensorflow は ver. 2.4.1, SHAP は ver. 0.39.0 を利用している。

### 4.3 評価結果

倍精度単精度によって生成された AI モデルへの SHAP の説明を図-2 にそれぞれ示す。

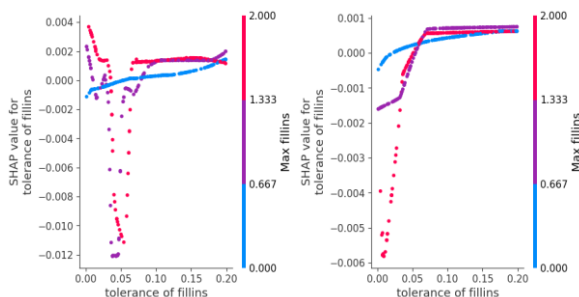


図-2 SHAP による説明結果

(問題サイズ  $262144 \times 262144$ ) . (左): 倍精度  
(右): 単精度.

図-2(左)より、倍精度演算では、最大 fill-in が 1 と 2 の場合、閾値(0 とみなす値)が約 0.05 より小さくなると急激に実行時間が短くなる要因が確認できる。しかし、約 0.03 よりさらに小さくなると、再度実行時間が長くなる要因が確認できる。これは fill-in レベル

の増加により、反復回数の削減と演算量の増加の間にトレードオフが生じる点から、アルゴリズム上の妥当性がある。

図-2(右)より、単精度では、最大 fill-in が 1 と 2 の場合、閾値が約 0.75 より小さくなると実行時間の減少の要因が確認できる。この一方で、最大 fill-in が 1 の時、閾値が約 0.05 より小さい場合に、実行時間が短くなる要因が認められない。この理由は、同一 fill-in レベルでも単精度演算になると倍精度演算に比べて、前処理の効率が悪くなることを意味しており、結果として収束性の劣化につながり、実行時間が増加するからと考えられるが、詳しい解析は今後の課題である。

## 5. まとめ

本研究では、ICTCG 法の性能パラメタチューニングの AT に AI 適用を行う場合に、XAI による妥当性の検証を行った。今回始めて行った倍精度演算と単精度演算の違いによる AI モデルの検証において、それぞれ異なる説明結果を得ることができた。倍精度計算の説明結果はアルゴリズム上の妥当性があるが、単精度演算の説明結果はさらなる検証が必要である。

## 謝辞

本研究は JSPS 科研費 JP19H0566 の助成を受けたものです。

## 参考文献

- [1] 青木将太, 片桐孝洋, 大島聡史, 永井亨, 星野哲也, 数値計算ライブラリの自動チューニングにおけるXAI適用の試み, 計算工学講演会論文集, pp. 904-907, 2023.
- [2] T. Katagiri, and D. Takahashi: Japanese Auto-tuning Research: Auto-tuning Languages and FFT, *Proc. of the IEEE*, 106 (11), pp.2056-2067, 2018.
- [3] M. T. Ribeiro, S. Singh, and C. Guestrin: Why should I trust you? : Explaining the predictions of any classifier, *Proc. of 22nd ACM SIGKDD*, pp. 1135-1144, 2016.
- [4] Yousef Saad: ILUT: A Dual Threshold Incomplete LU Factorization, 1994.
- [5] 河合直聡, 中島研吾, 前処理付きクリロフ部分空間法への低/任意精度の適用, 京都大学数理解析研究所 研究集会, 2022.
- [6] 名古屋大学情報基盤センター スーパーコンピュータ「不老」ホームページ <https://icts.nagoya-u.ac.jp/ja/sc/overview.html> [2023年12月25日閲覧]