ТТ-разложение для компактного представления тензоров

Родоманов А. О.

1 октября 2013 г.

Обзор

Тензоры. Основные форматы их представления

ТТ-разложение. Основные понятия

Алгоритм TT-SVD

Алгоритм ТТ-округления

Операции над тензорами в ТТ-формате

Крестовые алгоритмы ТТ-интерполяции

Примеры

QTT-формат

Что такое тензор

Мы будем понимать *тензор* как многомерный массив

$$\mathbf{A} = [A(i_1, \ldots, i_d)],$$

где

$$i_k = 1, \ldots, n_k \quad (k = 1, \ldots, d).$$

Терминология:

- *размерность* (*порядок*) тензора $= d$;
- *размер* тензора $= n_1 \times n_2 \times \ldots \times n_d$;
- *размерности* (*моды*) тензора = числа $n_1, n_2, \ldots, n_d$.

Проклятие размерности

Число элементов $= n^d$.
При $n = 2, d = 100$

$$2^{100} > 10^{30} \quad (\approx 10^{18} \text{ Пб памяти}).$$

Число элементов растет экспоненциально при росте $d \Rightarrow$ работать с тензорами при помощи стандартных средств невозможно.

Что делать

1. Выделить более узкий, специальный класс тензоров;
2. разработать формат представления тензоров из этого класса;
3. разработать эффективные методы выполнения базовых операций над тензорами (сложение, свертка и пр.).

Каноническое представление

$$A(i_1, i_2, \ldots, i_d) = \sum_{\alpha=1}^{R} U_1(i_1, \alpha) U_2(i_2, \alpha) \ldots U_d(i_d, \alpha).$$

Наименьшее возможное число R называется (*каноническим*) *рангом* тензора $\mathbf{A}$.

Проблемы:

- вычисление ранга R является NP-полной задачей;
- нахождение канонического представления является некорректно поставленной задачей (по Адамару);
- не существует хорошо работающих алгоритмов.

Разложение Таккера

$$A(i_1, i_2, \ldots, i_d) =$$
$$= \sum_{\alpha_1, \alpha_2, \ldots, \alpha_d} G(\alpha_1, \alpha_2, \ldots, \alpha_d) U_1(i_1, \alpha_1) U_2(i_2, \alpha_2) \ldots U_d(i_d, \alpha_d).$$

Проблемы:

- не лишено проклятия размерности.

Матрицы развертки. Определение

С каждым тензором **A** связаны $d-1$ так называемых *матриц развертки*

$$A_k = [A(i_1 i_2 \dots i_k; i_{k+1} \dots i_d)],$$

где

$$A(i_1 i_2 \dots i_k; i_{k+1} \dots i_d) = A(i_1, i_2, \dots i_d).$$

Здесь $i_1 i_2 \dots i_k$ и $i_{k+1} \dots i_d$ являются строчными и столбцовыми (мульти)индексами; A_k являются матрицами размера $M_k \times N_k$, где $M_k = \prod\limits_{s=1}^{k} n_s$, $N_k = \prod\limits_{s=k+1}^{d} n_s$.

Матрицы развертки. Пример

Рассмотрим 3-мерный тензор $\mathbf{A} = [A(i,j,k)]$, заданный своими элементами:

$$\begin{aligned} A(1,1,1) &= 111, \quad & A(2,1,1) &= 211,\\ A(1,2,1) &= 121, \quad & A(2,2,1) &= 221,\\ A(1,1,2) &= 112, \quad & A(2,1,2) &= 212,\\ A(1,2,2) &= 122, \quad & A(2,2,2) &= 222. \end{aligned}$$

Тогда

$$A_1 = [A(i;jk)] = \begin{bmatrix} 111 & 121 & 112 & 122 \\ 211 & 221 & 212 & 222 \end{bmatrix},$$

$$A_2 = [A(ij;k)] = \begin{bmatrix} 111 & 112 \\ 211 & 212 \\ 121 & 122 \\ 221 & 222 \end{bmatrix}.$$

Мотивация ТТ-разложения

$$A(i_1 i_2; i_3 i_4 i_5 i_6) = \sum_{\alpha_2} U(i_1 i_2; \alpha_2) V(i_3 i_4 i_5 i_6; \alpha_2)$$

Слева 6-мерный тензор, а справа 3-мерный и 5-мерный.
Размерность уменьшилась!
Далее рекурсивно.

TT-разложение

$$A(i_1, i_2, \ldots, i_d) =$$
$$= \sum_{\alpha_0, \alpha_1, \ldots, \alpha_d} G_1(\alpha_0, i_1, \alpha_1) G_2(\alpha_1, i_2, \alpha_2) \ldots G_d(\alpha_{d-1}, i_d, \alpha_d),$$

где G_k — 3-мерные тензоры размеров $r_{k-1} \times n_k \times r_k$;
$r_0 = r_d = 1$ (вводится искусственно для удобства).
Терминология:

- G_k называются *TT-ядрами* тензора **A**;
- числа r_k называются *TT-рангами* тензора **A**.

Замечание

Число параметров: $O(dnr^2)$.

ТТ-разложение. Компактная запись

Вспомнив про операцию умножения матриц, ТТ-разложение можно записать более компактно:

$$A(i_1, i_2, \dots, i_d) = G_1(i_1) G_2(i_2) \dots G_d(i_d),$$

где

$$G_k(i_k) = [G_k(\alpha_{k-1}, i_k, \alpha_k)],$$

т. е. $G_k(i_k)$ являются матрицами размеров $r_{k-1} \times r_k$.

ТТ-ранги ограничены снизу

Утверждение

ТТ-ранги ограничены снизу рангами соответствующих матриц развертки:

$$r_k \geqslant \operatorname{rank} A_k.$$

Фробениусова норма

1. *Фробениусовой нормой* матрицы M размеров $m \times n$ называется число
$$\|M\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n} M_{ij}^2}.$$
2. Аналогично определяется фробениусова норма тензора $\mathbf{A}$:
$$\|\mathbf{A}\|_F = \sqrt{\sum_{i_1,\ldots,i_d} A^2(i_1,\ldots,i_d)}.$$

Ортогональные матрицы

1. Квадратная матрица Q называется *ортогональной*, если выполняется
$$QQ^T = Q^T Q = I;$$
2. Умножение матрицы на ортогональную не меняет ее фробениусовой нормы, т. е.
$$\|UAV\|_F = \|A\|_F,$$
где U и V — ортогональные матрицы.

Сингулярное разложение матрицы

Любая матрица A размеров $m \times n$ может быть представлена в виде произведения

$$A = U\Sigma V^T,$$

где

- U — ортогональная матрица размеров $m \times m$;
- Σ — диагональная матрица размеров $m \times n$ с неотрицательными числами σ_i на главной диагонали;
- V — ортогональная матрица размеров $n \times n$.

Терминология:

- σ_i (элементы главной диагонали) называются *сингулярными числами* матрицы A;
- столбцы матриц U и V называются, соответственно, *левыми* и *правыми сингулярными векторами* матрицы A.

Приближение матрицей меньшего ранга

Пусть заданную матрицу $A = U\Sigma V^T$ требуется приблизить некоторой другой матрицей B тех же размеров, но меньшего ранга k:

$$B \approx A, \quad \operatorname{rank} B = k.$$

Теорема (Эккарта-Янга)

$$\underset{B\colon \operatorname{rank} B = k}{\arg\min} \|A - B\|_F = U_k \Sigma_k V_k^T,$$

где

- Σ_k — *диагональная матрица, содержащая старшие k сингулярных чисел матрицы A на главной диагонали;*
- U_k *и* V_k — *матрицы с ортонормированной системой из k столбцов — сингулярных векторов, отвечающих старшим сингулярным числам.*

Основные теоремы

Теорема

Для любого тензора **A** *существует TT-разложение с рангами*

$$r_k = \operatorname{rank} A_k.$$

Теорема (алгоритм TT-SVD)

Для любого тензора **A** *существует TT-приближение* **T** *с заданными TT-рангами* r_k *такое, что*

$$\|\mathbf{A} - \mathbf{T}\|_F \leqslant \sqrt{\sum_{k=1}^{d-1} \varepsilon_k^2},$$

где

$$\varepsilon_k = \min_{B:\, \operatorname{rank} B \leqslant r_k} \|A_k - B\|_F.$$

Следствия

Следствие

Если тензор $\mathbf{A}$ *допускает приближение в каноническом формате с* R *слагаемыми и точностью* ε, *то существует TT-приближение с TT-рангами* $r_k \leqslant R$, *причем точность этого приближения равна* $\sqrt{d-1}\,\varepsilon$.

Следствие (квазиоптимальность)

Пусть задан тензор $\mathbf{A}$ *и верхние ограничения* r_k *на TT-ранги. Тогда для* $\mathbf{A}$ *всегда существует наилучшее TT-приближение* $\mathbf{A}^{\mathbf{best}}$ *такое, что* $\operatorname{rank} A_k^{best} \leqslant r_k$. *При этом TT-приближение* $\mathbf{T}$, *вычисляемое алгоритмом TT-SVD, является квазиоптимальным:*

$$\|\mathbf{A} - \mathbf{T}\|_F \leqslant \sqrt{d-1}\|\mathbf{A} - \mathbf{A}^{\mathbf{best}}\|_F.$$

Задача ТТ-округления

Пусть уже имеется ТТ-представление

$$A(i_1, i_2, \dots, i_d) = G_1(i_1) G_2(i_2) \dots G_d(i_d),$$

однако ТТ-ядра $G_k(i_k)$ имеют неоптимальные ТТ-ранги r_k. Мы хотим найти ТТ-приближение $\mathbf{B} \approx \mathbf{A}$, которое бы имело меньшие ТТ-ранги $r'_k \leqslant r_k$.

Способ вычисления SVD

Пусть

$$A_1 = GQ,$$

где Q — ортогональная матрица.
Вычислим сокращенное сингулярное разложение

$$G = U\Sigma V^T + E, \quad \|E\|_F \leqslant \varepsilon.$$

Тогда сокращенное сингулярное разложение для A_1 будет следующим:

$$A_1 = U\Sigma \widetilde{V}^T + \widetilde{E}, \quad \|\widetilde{E}\|_F \leqslant \varepsilon,$$

где $\widetilde{V} = Q^T V$, $\widetilde{E} = EQ$.

QR-разложение

Любая матрица A размеров $m \times n$, где $m \geqslant n$, может быть представлена в виде

$$A = QR,$$

где

- Q — матрица размеров $m \times n$ с ортогональными столбцами (т. е. $Q^T Q = I$);
- R — верхнетреугольная матрица размеров $n \times n$.

RQ-разложение

Аналогично любая матрица размеров $m \times n$, где $n \geqslant m$, обладает RQ-разложением.

Ортогонализация ТТ-ядер

Алгоритм: один проход по всем ТТ-ядрам справа налево.

Лемма (об ортогональности)

Пусть «широкая» матрица Q представляется в виде

$$Q(\alpha'_1; i_2 \dots i_d) = \sum_{\alpha'_2,\dots,\alpha'_d} Q_2(\alpha'_1, i_2, \alpha'_2) \dots Q_d(\alpha'_{d-1}, i_d, \alpha'_d),$$

где ядра Q_k удовлетворяют следующим ортогональным условиям:

$$\sum_{i_k,\alpha'_k} Q_k(\alpha'_{k-1}, i_k, \alpha'_k) Q_k(\widetilde{\alpha}'_{k-1}, i_k, \alpha'_k) = \delta(\alpha'_{k-1}, \widetilde{\alpha}'_{k-1}).$$

Тогда Q имеет ортонормированную систему строк:

$$\sum_{i_2,\dots,i_d} Q(\alpha'_1; i_2 \dots i_d) Q(\widetilde{\alpha}'_1; i_2 \dots i_d) = \delta(\alpha'_1, \widetilde{\alpha}'_1).$$

Алгоритм ТТ-округления

Алгоритм: ортогонализация ТТ-ядер + SVD.
Сложность: $O(dnr^3)$, но может быть уменьшена до $O(dnr^2 + dr^4)$.

Из канонического формата в ТТ-формат

Пусть имеется каноническое представление

$$A(i_1, i_2, \dots, i_d) = \sum_{\alpha} U_1(i_1, \alpha) U_2(i_2, \alpha) \dots U_d(i_d, \alpha).$$

Зная такое представление, легко найти ТТ-разложение

$$A(i_1, i_2, \dots, i_d) = \Lambda_1(i_1) \Lambda_2(i_2) \dots \Lambda_d(i_d).$$

В роли ТТ-ядер в данном случае нужно взять

$$\Lambda_k(i_k) = \operatorname{diag} U(i_k, :), \quad k = 2, \dots, d-1,$$

$$\Lambda_1(i_1) = U(i_1, :), \quad \Lambda_d(i_d) = (U(i_d, :))^T.$$

Затем нужно применить алгоритм ТТ-округления, чтобы уменьшить ранги.

Сложение тензоров и умножение тензора на число

- Сумма тензоров $\mathbf{C} = \mathbf{A} + \mathbf{B}$:

$$C(i_1, \dots, i_d) = A(i_1, \dots, i_d) + B(i_1, \dots, i_d).$$

ТТ-ядра определяются следующим образом:

$$C_k(i_k) = \begin{bmatrix} A_k(i_k) & 0 \\ 0 & B_k(i_k) \end{bmatrix}, \quad k = 2, \dots, d-1,$$

$$C_1(i_1) = \begin{bmatrix} A_1(i_1) & B_1(i_1) \end{bmatrix}, \quad C_d(i_d) = \begin{bmatrix} A_d(i_d) \\ B_d(i_d) \end{bmatrix}.$$

ТТ-ранги удваиваются.

- Умножение тензора на число.
  Одно из ТТ-ядер умножается на это число. ТТ-ранги не увеличиваются.

Многомерная свертка

Многомерной сверткой называется выражение вида

$$W = \sum_{i_1,\ldots,i_d} A(i_1,\ldots,i_d) u_1(i_1)\ldots u_d(i_d),$$

где u_k — заданные векторы длины n_k.
В этом случае

$$W = \Gamma_1 \ldots \Gamma_d,$$

где

$$\Gamma_k = \sum_{i_k} u_k(i_k) G_k(i_k).$$

Кронекерово произведение

Пусть A — матрица размеров $m \times n$, а B — матрица размеров $p \times q$. *Кронекеровым произведением* матриц A и B называется блочная матрица

$$C = A \otimes B = \begin{bmatrix} a_{11}B & \dots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \dots & a_{mn}B \end{bmatrix}$$

размеров $mp \times nq$.

Смешанное произведение

Кронекерово произведение обладает следующим полезным свойством:

$$AC \otimes BD = (A \otimes B)(C \otimes D).$$

Поэлементное произведение

Поэлементным произведением (*произведением Адамара*) двух тензоров $\mathbf{A}$ и $\mathbf{B}$ называется тензор $\mathbf{C} = \mathbf{A} \circ \mathbf{B}$, элементы которого заданы по правилу

$$C(i_1, \ldots, i_d) = A(i_1, \ldots, i_d) B(i_1, \ldots, i_d).$$

ТТ-ядра $\mathbf{C}$ можно вычислить следующим образом:

$$C_k(i_k) = A_k(i_k) \otimes B_k(i_k).$$

В результате такой операции ТТ-ранги $\mathbf{C}$ равны произведениям соответствующих ТТ-рангов.

Скалярное произведение

Скалярным произведением тензоров **A** и **B** называется выражение

$$\langle \mathbf{A}, \mathbf{B} \rangle = \sum_{i_1, \ldots, i_d} A(i_1, \ldots, i_d) B(i_1, \ldots, i_d).$$

Замечание: скалярное произведение = поэлементное произведение + многомерная свертка.

Норма

Через скалярное произведение легко вычислить фробениусову норму
$$\|\mathbf{A}\|_F = \sqrt{\langle \mathbf{A}, \mathbf{A} \rangle}.$$

Сложность операций в ТТ-формате

Операция	Сложность
Сложение тензоров	$O(dnr^2 + dr^4)$
Многомерная свертка	$O(dnr + dr^3)$
Поэлементное произведение	$O(dnr^4)$
Скалярное произведение	$O(dnr^2 + dr^4)$
Норма	$O(dnr^2 + dr^4)$

Как найти ТТ-разложение

1. Найти ТТ-представление теоретически (напрямую или, например, через каноническое разложение);
2. ТТ-интерполировать заданный тензор по нескольким известным его элементам.

Задача ТТ-интерполяции

Тензор **A** задан процедурой $A(i_1, \dots, i_d)$ вычисления его отдельного элемента.
Требуется построить ТТ-разложение

$$B(i_1, \dots, i_d) = G_1(i_1) \dots G_d(i_d),$$

задающее тензор **B** так, чтобы $\mathbf{B} \approx \mathbf{A}$.

Задача интерполяции для матриц

Пусть A является матрицей размеров $m \times n$ и ранга r.
Тогда A допускает скелетное разложение

$$A = C\widetilde{A}^{-1}R,$$

где $C = A(:, \mathcal{J})$ — некоторые r столбцов A, $R = A(\mathcal{I}, :)$ — некоторые r строк A, а $\widetilde{A} = A(\mathcal{I}, \mathcal{J})$ — невырожденная матрица на пересечении этих строк и столбцов.

Алгоритм `maxvol`

В качестве $\widetilde{A}$ следует использовать подматрицу наибольшего *объема* (т. е. модуля определителя).
На практике вместо подматрицы наибольшего объема используют квазиоптимальную подматрицу (т. е. с объемом, близким к максимальному). Такую подматрицу легко вычислить с помощью алгоритма `maxvol`.

Задача интерполяции для матриц – 2

Чтобы составить хорошее приближение, достаточно знать индексы строк или столбцов, содержащих подматрицу достаточно большого объема.

Алгоритм TT-cross

Метод интерполяции матриц нетрудно обобщить на d-мерный случай. Достаточно лишь знать $d-1$ наборов индексов столбцов, содержащих подматрицы достаточно большого объема.

Алгоритмы DMRG-cross и AMEn-cross

Недостаток предыдущего алгоритма: требуется явно указать все ТТ-ранги;

- если ранги недооценить, то получится слишком большая погрешность;
- если же ранги переоценить, то алгоритм будет долго работать.

1. Алгоритм `DMRG-cross` не требует задания ТТ-рангов.
2. Алгоритм `AMEn-cross` является «ускоренной версией» алгоритма `DMRG-cross`.

Пример крестовой TT-интерполяции

Тензор Гильберта:

$$A(i_1, i_2, \ldots, i_d) = \frac{1}{i_1 + i_2 + \ldots + i_d}.$$

Используется `TT-cross`.

r_{max}	Время	Число итераций	Относительная точность
2	1.37	5	1.897278e+00
3	4.22	7	5.949094e-02
4	7.19	7	2.226874e-02
5	15.42	9	2.706828e-03
6	21.82	9	1.782433e-04
7	29.62	9	2.151107e-05
8	38.12	9	4.650634e-06
9	48.97	9	5.233465e-07
10	59.14	9	6.552869e-08
11	72.14	9	7.915633e-09
12	75.27	8	2.814507e-09

[3, И. В. Оселедец, Е. Е. Тыртышников]

Вычисление d-мерных интегралов

$$I(d) = \int\limits_{[0,1]^d} \sin(x_1 + x_2 + \ldots + x_d) dx_1 dx_2 \ldots dx_d =$$

$$= \operatorname{Im}\left(\left(\frac{e^i - 1}{i}\right)^d\right).$$

Используется квадратура Чебышева с $n = 11$ узлами + `TT-cross` с $r_{max} = 2$.

d	$I(d)$	Относительная точность	Время
10	-6.299353e-01	1.409952e-15	0.14
100	-3.926795e-03	2.915654e-13	0.77
500	-7.287664e-10	2.370536e-12	4.64
1 000	-2.637513e-19	3.482065e-11	11.70
2 000	2.628834e-37	8.905594e-12	33.05
4 000	9.400335e-74	2.284085e-10	105.49

[3, И. В. Оселедец, Е. Е. Тыртышников]

Вычисление d-мерных интегралов – 2

$$I(d) = \int\limits_{[0,1]^d} \sqrt{x_1^2 + x_2^2 + \ldots + x_d^2}\, dx_1 dx_2 \ldots dx_d.$$

Выбирается $d = 100$. Эталон: квадратура Чебышева с $n = 41$ узлами + TT-cross с $r_{max} = 32$.
Таблица для $n = 11$ узлов:

r_{max}	Относительная точность	Время
2	1.747414e-01	1.76
4	2.823821e-03	11.52
8	4.178328e-05	42.76
10	3.875489e-07	66.28
12	2.560370e-07	94.39
14	4.922604e-08	127.60
16	9.789895e-10	167.02
18	1.166096e-10	211.09
20	2.706435e-11	260.13

[3, И. В. Оселедец, Е. Е. Тыртышников]

Квантизация (QTT-представление)

Почти все операции зависят линейно от d $\Rightarrow$ для матриц и векторов можно получать логарифмическую сложность от размера:

- $a(i) \to A(i_1, i_2, \ldots, i_n)$;
- $A(i,j) \to A(i_1, i_2, \ldots, i_n, j_1, j_2, \ldots, j_n)$.

Пример

$$I = \int\limits_0^{+\infty} \frac{\sin x}{x} dx = \frac{\pi}{2}.$$

Используется квадратура прямоугольников с числом разбиений $n = 2^{80}$ + `DMRG-cross`.

Результат:

$$r_{max} = 16,$$
$$\text{точность} = 2.0403\text{e-}11,$$
$$\text{время} = 1 \text{ сек.}$$

Заключение

ТТ-формат

- лишен проблемы проклятия размерности;
- обладает набором быстрых надежно работающих алгоритмов;
- является новым направлением в вычислительной математике.

Ссылки

I. V. Oseledets.
Compact matrix form of the d-dimensional tensor decomposition.
INM RAS Preprint, 2009-01.

I. V. Oseledets.
Tensor-train decomposition.
SIAM, 2011.

I. V. Oseledets and E. E. Tyrtyshnikov.
TT-Cross approximation for multidimensional arrays.
INM RAS Preprint, 2009-05.

D. V. Savostyanov and I. V. Oseledets.
Fast adaptive interpolation of multi-dimensional arrays in tensor train format.
INM RAS Preprint, 2011-03.