Искенеева Камиля КМБО-01-20

Виноградова Арина КМБО-01-20

Грузберг Александр КМБО-01-20

Email: kamila.iskeneeva@yandex.ru

arina.airina@yandex.ru

sasha@gruzberg.ru

Telegram: @iskam17

@ari_grape

@alexgruzberg

Ускорение итеративного уточнения для разложения по сингулярным значениям

Аннотация

В работе предложены быстрые численные алгоритмы для повышения точности вычисления сингулярных векторов для вещественной матрицы. В этой статье предложен итеративный алгоритм уточнения для разложения по сингулярным значениям, запущенный с помощью пятикратного умножения матриц с высокой точностью. Алгоритм работает для задачи, в которой нет кратных и сгруппированных сингулярных значений. Кроме того, предложены четыре алгоритма, построенных с использованием высокоточных матричных умножений, два алгоритма с четырехкратным умножением и два других алгоритма с пятикратным умножением. Эти алгоритмы основаны на идее итерационного метода уточнения линейных систем со смешанной точностью. Численные эксперименты демонстрируют ускорение и квадратичную сходимость предложенных алгоритмов.

1. Введение

$$A = U\Sigma V^T \quad (1)$$

- сингулярное разложение, где $A \epsilon R^{mxn}$, $U \epsilon R^{mxm}, \Sigma \epsilon R^{mxn}, V \epsilon R^{nxn}$.

Мы предполагаем, что m ≥ n

Приближение: $\hat{u}_i \approx u_i, \hat{\sigma}_i \approx \sigma_i, \hat{v}_i \approx v_i$.

Для $k \leq n$ положим:

$$U_{1:k} := (u_1, \dots, u_k), \quad \widehat{U'}_{1:k} := (\hat{u}'_1, \dots, \hat{u}'_k) \quad \in R^{mxk} \ ,$$

$$V_{1:k} := (v_1, \dots, v_k), \quad \widehat{V'}_{1:k} := (\hat{v}'_1, \dots, \hat{v}'_k) \quad \in R^{nxk}$$

$$\hat{u}'_i \coloneqq \frac{\hat{u}_i}{||\hat{u}_i||}, \hat{v}'_i \coloneqq \frac{\hat{v}_i}{||\hat{v}_i||}$$

$$\hat{\Sigma}_k \coloneqq diag(\hat{\sigma}'_1, \dots, \hat{\sigma}'_k) \in R^{kxk}, \text{где } \hat{\sigma}'_k := (\hat{u}'_i)^T A \hat{v}'_i$$

$$R_{1:k} \coloneqq A\hat{V}'_{1:k} - \widehat{U'}_{1:k}\hat{\Sigma}'_k, \quad S_{1:k} \coloneqq A^T\hat{U}'_{1:k} - \hat{V}'_{1:k}\hat{\Sigma}'_k,$$

$$\delta_k \coloneqq \min\left|\sigma'_i - \sigma_j\right|, \text{где } 1 \leq i \leq k, k < j \leq \max(n, k+1) \quad (2)$$

Если $\delta_k > 0$, то из (2) следует, что

$$\sqrt{\left\|\sin\Theta\left(U_{1:k}, \widehat{U'}_{1:k}\right)\right\|_F^2 + \left\|\sin\Theta\left(V_{1:k}, \widehat{V'}_{1:k}\right)\right\|_F^2} \leq \frac{\sqrt{\|R_{1:k}\|_F^2 + \|S_{1:k}\|_F^2}}{\delta_k}, \quad (3)$$

где $\Theta\left(U_{1:k}, \widehat{U'}_{1:k}\right)$ и $\Theta\left(V_{1:k}, \widehat{V'}_{1:k}\right)$- матрицы канонических углов между $U_{1:k}$ и $\widehat{U'}_{1:k}$ и между $V_{1:k}$ и $\widehat{V'}_{1:k}$ соответственно, а $\|\cdot\|_F$ обозначает норму Фробениуса. Из (3) следует, что по мере уменьшения значения δ_k в (2) точность вычисленных сингулярных векторов также снижается. Следовательно, для получения достаточно точных результатов при разложении по сингулярным значениям можно использовать итерационные методы уточнения.

2. Обозначения и предыдущая работа

1. Обозначения

$(\cdot)_h$ и $(\cdot)_l$ - результаты числовой арифметики, где все операции внутри круглых скобок выполняются с большей и меньшей точностью соответственно. Мы используем следующие комбинации точности [более высокая точность и более низкая точность], [четырехкратная точность и двойная точность] и [двойная точность и одинарная точность]. Для простоты мы опустим члены, меньшие $O(m^k n^l), k + l = 3$ из подсчета операций.

2. Итеративный метод уточнения со смешанной точностью

Мы рассматриваем линейную систему $Ax = b$ для $x, b \in R^n$ и не сингулярную матрицу $A \in R^{n \times n}$. Пусть $\hat{x} \approx x$ - вычисленное решение $Ax = b$.

Существует итеративный алгоритм уточнения с арифметикой смешанной точности для повышения точности $\hat{x}$.

Алгоритм 1 Итерационное уточнение приближенного решения $\hat{x}$ линейной системы $Ax = b$. Начальное значение $\hat{x}$ получено с использованием арифметики меньшей точности.

repeat

Вычислите $r \leftarrow (b - A\hat{x})_h$

Преобразуйте значение r в значение с меньшей точностью.

Вычислите $y \leftarrow (A^{-1}r)_l$

Обновите $\hat{x}$ как $\hat{x} \leftarrow (\hat{x} + y)_{h\ or\ l}$

until точность $\hat{x}$ не станет достаточной

3. Итеративное уточнение для симметричной декомпозиции по собственным значениям

I_n - единичная матрица $n \times n$. Предполагаем, что $A = A^T \in R^{n\times n}$. Пусть $X \in R^{n\times n}$ ортогонально, $D \in R^{n\times n}$ диагонально и $A = XDX^T$. $i -$е столбцы X являются собственными векторами $x_{(i)} \in R^n$, а $i -$е диагональные элементы D являются собственными значениями $\lambda_i \in R$ для $i = 1, \dots, n$. Здесь мы предполагаем это $\lambda_i \neq \lambda_j$ для $i \neq j$. Для $\hat{X} \approx X$ мы определяем матрицу ошибок $E \in R^{n\times n}$ таким образом, что $X = \hat{X}(I_n + E)$. Алгоритм для вычисления $\tilde{E} \approx E$ и преобразования $\hat{X}$ в $\tilde{X}$ как $\tilde{X} \leftarrow \hat{X}(I_n + E)$. Алгоритм сходится квадратично при условии, что матрица ошибок E удовлетворяет следующему условию:

$$\|E\|_2 < \min\left(\frac{\min\limits_{1\le i<j\le n}|\lambda_i - \lambda_j|}{10\sqrt{n}\|A\|_2}, \frac{1}{100}\right)$$

Алгоритм 2 Уточнения приближенных собственных векторов $\hat{X} \in R^{n\times n}$ для вещественной симметричной матрицы $A \in R^{n\times n}$. Предположим $\tilde{\lambda_i} \neq \tilde{\lambda_j}$ для $i \neq j$. Общая стоимость составляет от $6n^3$ до $8n^3$ операций.

Требуется: $A \in R^{n\times n}, \hat{X} \in R^{n\times n}$

Убедитесь: $\tilde{X} \in R^{n\times n}, \tilde{D} \in R^{n\times n}$

function $[\tilde{X}, \tilde{D}] \leftarrow RefSyEv(A, \hat{X})$

$R \leftarrow \left(I_n - \hat{X}^T \hat{X}\right)_h$

$S \leftarrow \left(\hat{X}^T A \hat{X}\right)_h$

$\widetilde{\lambda_i} \leftarrow \left(\frac{s_{ii}}{1 - r_{ii}}\right)_h$ для $(1 \leq i \leq n)$

$\widetilde{D} \leftarrow diag\left(\widetilde{\lambda_1}, \dots, \widetilde{\lambda_n}\right)$

$F \leftarrow \left(S + R\widetilde{D}\right)_h$

$$\widetilde{e_{ij}} \leftarrow \begin{cases} \left(\dfrac{f_{ii}}{\widetilde{\lambda_j} - \widetilde{\lambda_i}}\right)_h & \text{для } (1 \leq i, j \leq n) \\ \left(\dfrac{r_{ii}}{2}\right)_h & \text{иначе} \end{cases}$$

$\tilde{X} \leftarrow \left(\hat{X} + \hat{X}\tilde{E}\right)_h$

end function

Здесь $S = (s_{ij})$, $R = (r_{ij})$, $F = (f_{ij})$, $\tilde{E} = (\tilde{e}_{ij})$.

Алгоритм для уменьшения требуемого параметра-математической точности алгоритма 2. Он основан на итеративном уточнении для решения линейных систем с использованием арифметики смешанной точности. Из $R = I_n - \hat{X}^T \hat{X}$ и $S = \hat{X}^T A \hat{X}$ в алгоритме 2, F в строке 6 удовлетворяет

$$F = S + R\widetilde{D} = \hat{X}^T A \hat{X} + (I_n - \hat{X}^T \hat{X})\widetilde{D} = \hat{X}^T (A\hat{X} - \hat{X}\widetilde{D}) + \widetilde{D}$$

Поскольку на диагональную часть F ссылки нет, ее можно вычислить следующим образом

$$F := \hat{X}^T W \quad (5)$$

где $W \coloneqq A\hat{X} - \hat{X}\widetilde{D}$. Условие сходимости такое же, как (4) для алгоритма 2.

Алгоритм 3 Уточнение приближенных собственных векторов $\hat{X} \in R^{n\times n}$ для вещественной симметричной матрицы $A \in R^{n\times n}$. Предположим $\widetilde{\lambda_i} \neq \widetilde{\lambda_j}$ для $i \neq j$. Общая стоимость составляет $6n^3$ операций.

Требуется: $A \in R^{n\times n}, \hat{X} = \left(x_{(1)}, \dots, x_{(n)}\right) \in R^{n\times n}$

Убедитесь: $\tilde{X} \in R^{n\times n}, \widetilde{D} \in R^{n\times n}$

function $[\tilde{X}, \widetilde{D}] \leftarrow RefSyEv2(A, \hat{X})$

$P \leftarrow (A\hat{X})_h$

$r_{ii} \leftarrow (1 - \widehat{x_{(i)}}^T \widehat{x_{(i)}})$ для $(1 \le i \le n)$

$s_{ii} \leftarrow \left(\widehat{x_{(i)}}^T p_{(i)}\right)_h$ для $(1 \le i \le n)$

$\widetilde{\lambda_i} \leftarrow \left(\frac{s_{ii}}{1 - r_{ii}}\right)_h$ для $(1 \le i \le n)$

$\widetilde{D} \leftarrow diag(\widetilde{\lambda_1}, \dots, \widetilde{\lambda_n})$

$W \leftarrow (P - \hat{X}\widetilde{D})_h$

$F \leftarrow (\hat{X}^T W)_l$

$$\widetilde{e_{ij}} \leftarrow \begin{cases} \left(\dfrac{f_{ii}}{\widetilde{\lambda_j} - \widetilde{\lambda_i}}\right)_h & i \ne j \\ \left(\dfrac{r_{ii}}{2}\right)_h & \text{иначе} \end{cases} \quad \text{для } (1 \le i, j \le n)$$

$\tilde{X} \leftarrow \left(\hat{X} + (\hat{X}\tilde{E})_l\right)_h$

end function

4. Итеративное уточнение для разложения по сингулярным значениям

Здесь мы вводим итеративное уточнение для полного разложения по сингулярным значениям. Далее мы предполагаем, что

$$\sigma_1 > \sigma_2 > \cdots > \sigma_n$$

и приближение $\widetilde{\sigma_i}$ для σ_i удовлетворяет $\widetilde{\sigma_i} \ne \widetilde{\sigma_j}$ для $i \ne j$. Пусть $\widehat{U} \approx U$ и $\widehat{V} \approx V$ для U и V в (1). Определите матрицы ошибок $F \in R^{m\times m}$ и $G \in R^{n\times n}$ такими

$$U = \widehat{U}(I_m + F) \text{ и } V = \widehat{V}(I_n + G)$$

соответственно. Алгоритм, основанный на той же идее, что и алгоритм 2. Он вычисляет $\tilde{F} \approx F$ и $\tilde{G} \approx G$ и обновляет $\widehat{U}$ и $\widehat{V}$ до U и V как $\widetilde{U} \leftarrow \widehat{U}(I_n + F)$ и $\tilde{V} \leftarrow \widehat{V}(I_n + G)$) соответственно. Этот процесс показан в алгоритме 4. Обратите внимание, что для $T, \widehat{U}, \widetilde{U}, R \in R^{m\times m}$ у нас есть

$$T = (T_1 T_2), \widehat{U} = (\widehat{U}_1 \widehat{U}_2), \widetilde{U} = (\widetilde{U}_1\ \widetilde{U}_2), R = \begin{pmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{pmatrix}$$

с $T_1, \widehat{U}_1,\ \widetilde{U}_1 \in R^{m\times n}$, $T_2, \widehat{U}_2,\ \widetilde{U}_2 \in R^{m\times(m-n)}$, $R_{11} \in R^{n\times n}$, $R_{12}, {R_{21}}^T \in R^{n\times(m-n)}$, $R_{22} \in R^{(m-n)\times(m-n)}$.

Алгоритм 4 Уточнения приближенных сингулярных векторов $\widehat{U} \in R^{m\times m}$ и $\widehat{V} \in R^{n\times n}$ для вещественной матрицы $A \in R^{m\times n}$. Предположим $\widetilde{\sigma_i} \neq \widetilde{\sigma_j}$ для $i \neq j$. Общая стоимость составляет от $3m^3 + 2m^2n + 2mn^2 + 3n^3$ до $4m^3 + 2m^2n + 2mn^2 + 4n^3$ операций.

Требуется: $A \in R^{m\times n}, \widehat{U} \in R^{m\times m}, \widehat{V} \in R^{n\times n}$

Убедитесь: $\widetilde{U} \in R^{m\times m}, \widetilde{\Sigma} \in R^{m\times n}, \widetilde{V} \in R^{n\times n}$

function$[\widetilde{U}, \widetilde{\Sigma}, \widetilde{V}] \leftarrow RefSVD(A, \widehat{U}, \widehat{V})$

$$R \leftarrow \left(I_m - \widehat{U}^T\widehat{U}\right)_h$$

$$S \leftarrow \left(I_n - \widehat{V}^T\widehat{V}\right)_h$$

$$T \leftarrow \left(\widehat{U}^T A\widehat{V}\right)_h$$

$$\widetilde{\sigma_i} \leftarrow \left(\frac{t_{ii}}{1 - \frac{r_{ii} + s_{ii}}{2}}\right)_h \text{ для } (1 \le i \le n)$$

$$\widetilde{\Sigma}_n \leftarrow diag(\widetilde{\sigma_1}, \dots, \widetilde{\sigma_n});\ \widetilde{\Sigma} \leftarrow \left(\widetilde{\Sigma}_n, O_{n,m-n}\right)^T$$

$$C_\alpha \leftarrow \left(T_1 + R_{11}\,\widetilde{\Sigma}_n\right)_h;\ C_\beta \leftarrow \left({T_1}^T + S\,\widetilde{\Sigma}_n\right)_h$$

$$D \leftarrow \left(\widetilde{\Sigma}_n C_\alpha + C_\beta \widetilde{\Sigma}_n\right)_h$$

$$E \leftarrow \left(C_\alpha \widetilde{\Sigma}_n + \widetilde{\Sigma}_n C_\beta\right)_h$$

$$\widetilde{g_{ij}} \leftarrow \begin{cases} \left(\dfrac{d_{ij}}{\widetilde{\sigma_j}^2 - \widetilde{\sigma_i}^2}\right)_h & i \neq j \\ \left(\dfrac{s_{ii}}{2}\right)_h & \text{иначе} \end{cases} \quad \text{для } (1 \le i, j \le n)$$

$$\widetilde{f_{ij}} \leftarrow \begin{cases} \left(\dfrac{e_{ij}}{\widetilde{\sigma_j}^2 - \widetilde{\sigma_i}^2}\right)_h & (i \neq j, i, j \le n) \\ \left(-\dfrac{t_{ji}}{\widetilde{\sigma_i}}\right)_h & (i \le n < j) \\ \left(r_{ij} - \widetilde{f_{ji}}\right)_h & (j \le n < i) \\ \left(\dfrac{r_{ij}}{2}\right)_h & \text{иначе} \end{cases} \quad \text{для } (1 \le i, j \le n)$$

$$\widetilde{U} \leftarrow \left(\widehat{U} + \widehat{U}\widetilde{F}\right)_h;\ \widetilde{V} \leftarrow \left(\widehat{V} + \widehat{V}\widetilde{G}\right)_h$$

end function

3. Алгоритм быстрого итеративного уточнения разложения матрицы по сингулярным значениям

Напомним, что разложением матрицы $A \in \mathbb{R}^{m\times n}$ по сингулярным значениям (SVD) называется такое соотношение

$$A = U\Sigma V^T, \qquad (1)$$

где $U \in \mathbb{R}^{m\times m}, V \in \mathbb{R}^{n\times n}$ – ортогональные матрицы, составленные из левых сингулярных векторов $u_{(i)} \in \mathbb{R}^m, i = 1, \ldots, m$ и правых сингулярных векторов $v_{(i)} \in \mathbb{R}^n, i = 1, \ldots, n$ соответственно, а $\Sigma \in \mathbb{R}^{m\times n}$ – прямоугольная диагональная матрица, составленная из сингулярных значений $\sigma_i \in \mathbb{R}$, $i = 1, \ldots, n$ (полагается, что $m \geq n$).

Авторы Yuki Uchino, Takeshi Terao и Katsuhisa Ozaki предлагают быстрый (по сравнению с иными описанными в [1]) алгоритм для полного разложения матрицы по сингулярным значениям. Здесь и далее полагается, что сингулярные значения удовлетворяют следующему соотношению:

$$\sigma_1 > \sigma_2 > \sigma_3 > \cdots > \sigma_n,$$

а их приближённые значения $\tilde{\sigma}_i$ таковы, что $\tilde{\sigma}_i \neq \tilde{\sigma}_j$ для $i \neq j$.

На вход алгоритма поступает матрица $A \in \mathbb{R}^{m\times n}$ и матрицы приближённых сингулярных векторов $\widehat{U} \in \mathbb{R}^{m\times m}, \widehat{V} \in \mathbb{R}^{n\times n}$. На выходе ожидается получение матриц уточнённых сингулярных векторов $\widetilde{U} \in \mathbb{R}^{m\times m}, \widetilde{V} \in \mathbb{R}^{n\times n}$ и матрицы уточнённых сингулярных значений $\widetilde{\Sigma} \in \mathbb{R}^{m\times n}$. Опишем алгоритм пошагово:

1. $P \leftarrow \left(A\widehat{V}\right)_h$ //символ $(\cdot)_h$ означает вычисление с двойной точностью;
2. $Q \leftarrow \left(A^T\widehat{U}_1\right)_h$ //$\widehat{U}_1 \in \mathbb{R}^{m\times n}$ – левый блок блочной матрицы $\left(\widehat{U}_1\widehat{U}_2\right) = \widehat{U}$;
3. $r_{ii} \leftarrow \left(1 - \left(\hat{u}_{(i)}\right)^T \hat{u}_{(i)}\right)_h, for\ (1 \leq i \leq n)$;
4. $s_{ii} \leftarrow \left(1 - \left(\hat{v}_{(i)}\right)^T \hat{v}_{(i)}\right)_h, for\ (1 \leq i \leq n)$;
5. $t_{ii} \leftarrow \left(\left(\hat{u}_{(i)}\right)^T \hat{p}_{(i)}\right)_h, for\ (1 \leq i \leq n)$ // $P = \left(p_{(1)}, p_{(2)}, \ldots, p_{(n)}\right)$;
6. $\tilde{\sigma}_i \leftarrow \left(\frac{t_{ii}}{1-(r_{ii}+s_{ii})/2}\right)_h, for\ (1 \leq i \leq n)$;
7. $\widetilde{\Sigma}_n \leftarrow diag(\tilde{\sigma}_1, \tilde{\sigma}_2, \ldots, \tilde{\sigma}_n)$;
8. $\widetilde{\Sigma} \leftarrow \left(\widetilde{\Sigma}_n, O_{n,m-n}\right)^T$ //$O_{n,m-n} \in \mathbb{R}^{n\times(m-n)}$ – матрица нулей;

9. $C_\gamma \leftarrow \left(P - \widehat{U}_1 \tilde{\Sigma}_n\right)_h$;

10. $C_\delta \leftarrow \left(Q - \hat{V} \tilde{\Sigma}_n\right)_h$;

11. $C_\alpha \leftarrow \left({\widehat{U}_1}^T C_\gamma\right)_l$ //символ $(\cdot)_l$ означает вычисление с одинарной точностью;

12. $C_\beta \leftarrow \left(\hat{V}^T C_\delta\right)_l$;

13. $D \leftarrow \left(\tilde{\Sigma}_n C_\alpha + C_\beta \tilde{\Sigma}_n\right)_h$ // $D = \left(d_{ij}\right)$;

14. $E \leftarrow \left(C_\alpha \tilde{\Sigma}_n + \tilde{\Sigma}_n C_\beta\right)_h$ // $E = \left(e_{ij}\right)$;

15. $\tilde{g}_{ij} \leftarrow \begin{cases} \left(\frac{d_{ij}}{{\tilde{\sigma}_j}^2 - {\tilde{\sigma}_i}^2}\right)_h, (i \neq j) \\ \left(\frac{s_{ii}}{2}\right)_h, (otherwise) \end{cases}, for\ (1 \leq i, j \leq n)$;

16. $\tilde{f}_{ij} \leftarrow \begin{cases} \left(\frac{e_{ij}}{{\tilde{\sigma}_j}^2 - {\tilde{\sigma}_i}^2}\right)_h, (i \neq j) \\ \left(\frac{r_{ii}}{2}\right)_h, (otherwise) \end{cases}, for\ (1 \leq i, j \leq n)$ // $\tilde{F}_{11} = \left(\tilde{f}_{ij}\right)$;

17. $\tilde{F}_{12} \leftarrow \left({\tilde{\Sigma}_n}^{-1} P^T \widehat{U}_2\right)_h$;

18. $\tilde{F}_{21} \leftarrow \left(\left({\widehat{U}_2}^T C_\gamma\right)_l {\tilde{\Sigma}_n}^{-1}\right)_h$;

19. $\tilde{F}_{22} \leftarrow \left(\frac{1}{2}\left(I_{m-n} - {\widehat{U}_2}^T \widehat{U}_2\right)\right)_h$ //$I_{m-n} \in \mathbb{R}^{(m-n)\times(m-n)}$ – единичная матрица;

20. $\widetilde{U} \leftarrow \left(\widehat{U} + \left(\widehat{U}\tilde{F}\right)_l\right)_h$ //$\tilde{F} = \begin{pmatrix} \tilde{F}_{11} & \tilde{F}_{12} \\ \tilde{F}_{21} & \tilde{F}_{22} \end{pmatrix}$ – блочная матрица, где $\tilde{F}_{11} \in \mathbb{R}^{n\times n}, \tilde{F}_{12} \in \mathbb{R}^{n\times(m-n)}, \tilde{F}_{21} \in \mathbb{R}^{(m-n)\times n}, \tilde{F}_{22} \in \mathbb{R}^{(m-n)\times(m-n)}$;

21. $\tilde{V} \leftarrow \left(\hat{V} + \left(\hat{V}\tilde{G}\right)_l\right)_h$ //$\tilde{G} = (\tilde{g}_{ij})$.

Общая «стоимость» алгоритма составляет $3m^3 + 2m^2n + 3mn^2 + 4n^3$ операций в лучшем случае и $4m^3 + 4mn^2 + 4n^3$ – в худшем, где

- $2n^3$: $\hat{V}^T C_\delta$;
- $2mn^2$: $A\hat{V}, A^T\widehat{U}_1, {\widehat{U}_1}^T C_\gamma$;
- $2mn(m-n)$: $P^T\widehat{U}_2, {\widehat{U}_2}^T C_\gamma$;
- $m(m-n)^2$ или $2m(m-n)^2$: ${\widehat{U}_2}^T \widehat{U}_2$;
- $2m^3$: $\widehat{U}\tilde{F}$;
- $2n^3$: $\hat{V}\tilde{G}$.