

Алгоритм итеративного уточнения для SVD, основанный на матричном умножении

Описание алгоритма

Данный алгоритм выполняет SVD разложение для матрицы $A \in \mathbb{R}^{m \times n}$, где $m \geq n$. В случае $m < n$ рассматривается A^T . Разложение имеет вид: $A = U\Sigma V^T$, где $U \in \mathbb{R}^{m \times m}$ и $V \in \mathbb{R}^{n \times n}$ – ортогональные матрицы, а $\Sigma \in \mathbb{R}^{m \times n}$ – диагональная матрица с элементами на диагонали λ_i , $0 < i \leq n$, такими что $\lambda_1 > \lambda_2 > \dots > \lambda_n > 0$, λ_i – сингулярные числа матрицы, а столбцы матриц U и V – левые и правые сингулярные векторы соответственно.

Идея алгоритма строится на использовании известных равенств и свойств матриц (ортогональность U и V и диагонализированность Σ):

$$U^T U = I_m, \quad (1)$$

$$V^T V = I_n, \quad (2)$$

$$U^T A V = \Sigma, \quad (3)$$

где I_m и I_n – единичные матрицы размеров m и n соответственно.

Пусть матрица $\hat{U} \in \mathbb{R}^{m \times m}$ и $\hat{V} \in \mathbb{R}^{n \times n}$ – приближение матриц U и V , такое, что $U = \hat{U}(I_m + F)$ и $V = \hat{V}(I_n + G)$, а $F \in \mathbb{R}^{m \times m}$ и $G \in \mathbb{R}^{n \times n}$ – корректирующие матрицы. Тогда свойство ортогональности (1) и (2) можно записать как:

$$(I_m + F^T) \hat{U}^T \hat{U} (I_m + F) = I_m,$$

$$(I_n + G^T) \hat{V}^T \hat{V} (I_n + G) = I_n,$$

откуда следует, что

$$\hat{U}^T \hat{U} = (I_m + F^T)^{-1} (I_m + F)^{-1}, \quad (4)$$

$$\hat{V}^T \hat{V} = (I_n + G^T)^{-1} (I_n + G)^{-1}. \quad (5)$$

Определим ε как $\varepsilon := \max(\varepsilon_F, \varepsilon_G)$, где $\varepsilon_F = \|F\|$, $\varepsilon_G = \|G\|$. Под нормой матрицы в данном случае подразумевается спектральная норма матрицы, т.е. норма, которая соответствует наибольшему сингулярному значению матрицы.

Предположим, что $\varepsilon < 1$. Тогда для $I_m + F$ и для $I_n + G$ справедливо:

$$(I_m + F)^{-1} = I_m - F + \Delta_F, \Delta_F := \sum_{k=2}^{\infty} (-F)^k, \|\Delta_F\| \leq \frac{\varepsilon_F}{1 - \varepsilon_F},$$

$$(I_n + G)^{-1} = I_n - G + \Delta_G, \Delta_G := \sum_{k=2}^{\infty} (-G)^k, \|\Delta_G\| \leq \frac{\varepsilon_G}{1 - \varepsilon_G}.$$

Подставляя данные уравнения в формулы (4) и (5), получим:

$$F + F^T = I_m - \hat{U}^T \hat{U} + \Delta_1, \quad \Delta_1 := \Delta_F + \Delta_F^T + (F - \Delta_F)^T (F - \Delta_F), \quad (6)$$

$$G + G^T = I_n - \hat{V}^T \hat{V} + \Delta_2, \quad \Delta_2 := \Delta_G + \Delta_G^T + (G - \Delta_G)^T (G - \Delta_G), \quad (7)$$

Также, подставляя $U = \hat{U}(I_m + F)$ и $V = \hat{V}(I_n + G)$ в (3), получим:

$$\Sigma - F^T \Sigma - \Sigma G = \hat{U}^T A \hat{V} + \Delta_3, \Delta_3 := -\Sigma \Delta_G - \Delta_F^T \Sigma - (A - \Delta_F)^T \Sigma (G - \Delta_G). \quad (8)$$

Здесь

$$\begin{aligned} \|\Delta_1\| &\leq \frac{(3 - 2\varepsilon_F)\varepsilon_F^2}{(1 - \varepsilon_F)^2} \leq \chi(\varepsilon)\varepsilon^2, \\ \|\Delta_2\| &\leq \frac{(3 - 2\varepsilon_G)\varepsilon_G^2}{(1 - \varepsilon_G)^2} \leq \chi(\varepsilon)\varepsilon^2, \\ \|\Delta_3\| &\leq \frac{\varepsilon_F^2 + \varepsilon_G^2 + (1 - \varepsilon_F - \varepsilon_G)\varepsilon_F \varepsilon_G}{(1 - \varepsilon_F)(1 - \varepsilon_G)} \|\Sigma\| \leq \chi(\varepsilon)\varepsilon^2 \|A\|, \end{aligned}$$

где

$$\chi(\varepsilon) := \frac{3 - 2\varepsilon}{(1 - \varepsilon)^2}.$$

Пренебрегая членами второго порядка Δ_1 , Δ_2 и Δ_3 , можно составить систему матричных уравнений из (6), (7) и (8), где $\tilde{F} = (\tilde{f}_{ij}) \in \mathbb{R}^{m \times m}$, $\tilde{G} = (\tilde{g}_{ij}) \in \mathbb{R}^{n \times n}$, $\tilde{\Sigma} = \text{diag}(\tilde{\lambda}_i) \in \mathbb{R}^{n \times m}$:

$$\begin{cases} \tilde{F} + \tilde{F}^T = R, R := I_m - \hat{U}^T \hat{U} \\ \tilde{G} + \tilde{G}^T = S, S := I_n - \hat{V}^T \hat{V} \\ \tilde{\Sigma} - \tilde{F}^T \tilde{\Sigma} - \tilde{\Sigma} \tilde{G} = T, T := \hat{U}^T A \hat{V} \end{cases} \Leftrightarrow \begin{cases} \tilde{f}_{ij} + \tilde{f}_{ji} = r_{ij}, & 1 \leq i, j \leq m \\ \tilde{g}_{ij} + \tilde{g}_{ji} = s_{ij}, & 1 \leq i, j \leq n \\ \tilde{\Sigma}_{ij} - \tilde{\lambda}_j \tilde{f}_{ji} - \tilde{\lambda}_i \tilde{g}_{ij} = t_{ij}, & 1 \leq i \leq m, 1 \leq j \leq n \end{cases}. \quad (9)$$

Для нахождения искомого разложения осталось лишь решить систему (9) для \tilde{F} , \tilde{G} и $\tilde{\Sigma}$. Для более оптимального счета, разобьем матрицы \tilde{F} , R , $\tilde{\Sigma}$ и T следующим образом:

$$\begin{aligned} \tilde{F} &= \begin{bmatrix} \tilde{F}_{11} & \tilde{F}_{12} \\ \tilde{F}_{21} & \tilde{F}_{22} \end{bmatrix}, \text{ где } \tilde{F}_{11} \in \mathbb{R}^{n \times n}, \tilde{F}_{12} \in \mathbb{R}^{n \times (m-n)}, \tilde{F}_{21} \in \mathbb{R}^{(m-n) \times n}, \tilde{F}_{22} \in \mathbb{R}^{(m-n) \times (m-n)}, \\ R &= \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix}, \text{ где } R_{11} \in \mathbb{R}^{n \times n}, R_{12} \in \mathbb{R}^{n \times (m-n)}, R_{21} \in \mathbb{R}^{(m-n) \times n}, R_{22} \in \mathbb{R}^{(m-n) \times (m-n)}, \\ \tilde{\Sigma} &= \begin{bmatrix} \tilde{\Sigma}_n \\ O \end{bmatrix}, \text{ где } \tilde{\Sigma}_n \in \mathbb{R}^{n \times n}, \text{ а } O \in \mathbb{R}^{(m-n) \times n} - \text{ нулевая матрица,} \\ T &= \begin{bmatrix} T_1 \\ T_2 \end{bmatrix}, \quad \text{ где } T_1 \in \mathbb{R}^{n \times n}, \quad T_2 \in \mathbb{R}^{(m-n) \times n}. \end{aligned}$$

Тогда

$$R_{11} = \tilde{F}_{11} + \tilde{F}_{11}^T, R_{21} = \tilde{F}_{21} + \tilde{F}_{12}^T, R_{22} = \tilde{F}_{22} + \tilde{F}_{22}^T,$$

и

$$\tilde{\Sigma}_n - \tilde{F}_{11}^T \tilde{\Sigma}_n - \tilde{\Sigma}_n \tilde{G} = T_1, \tilde{F}_{12}^T \tilde{\Sigma}_n = -T_2 \Leftrightarrow \tilde{\Sigma}_n \tilde{F}_{12} = -T_2^T.$$

Рассмотрим сначала диагональные элементы \tilde{F}_{11} и \tilde{G} . Их можно найти из первого и второго уравнений системы (9):

$$\tilde{f}_{ii} = \frac{r_{ii}}{2}, \tilde{g}_{ii} = \frac{s_{ii}}{2}, 1 \leq i \leq n.$$

Далее, рассмотрим третье уравнение из системы (9). Из него следует, что:

$$(1 - \tilde{f}_{ii} - \tilde{g}_{ii})\tilde{\lambda}_i = \left(1 - \frac{r_{ii} + s_{ii}}{2}\right)\tilde{\lambda}_i = t_{ii}, 1 \leq i \leq n.$$

Тогда, если $r_{ii} + s_{ii} \neq 2, 1 \leq i < n$ (возможно, что $r_{ii} + s_{ii} = 2$, однако обычно на практике значения r_{ii} и s_{ii} сильно меньше 1 [1]) мы имеем:

$$\tilde{\lambda}_i = \frac{t_{ii}}{1 - \frac{r_{ii} + s_{ii}}{2}}, 1 \leq i \leq n. \quad (10)$$

Далее найдем недиагональные элементы \tilde{F}_{11} и \tilde{G} . Из (9) и (10) можно получить линейную систему 4×4 :

$$\tilde{f}_{ij} + \tilde{f}_{ji} = r_{ij}, \quad (11)$$

$$\tilde{g}_{ij} + \tilde{g}_{ji} = s_{ij}, \quad (12)$$

$$\tilde{\lambda}_i \tilde{f}_{ij} + \tilde{\lambda}_j \tilde{g}_{ji} = -t_{ji}, \quad (13)$$

$$\tilde{\lambda}_j \tilde{f}_{ji} + \tilde{\lambda}_i \tilde{g}_{ij} = -t_{ij}, \quad (14)$$

для $1 \leq i, j \leq n, i \neq j$. Умножение (13) и (14) на $\tilde{\lambda}_i$ и $\tilde{\lambda}_j$ соответственно нам даст следующий результат:

$$\tilde{\lambda}_i^2 \tilde{f}_{ij} + \tilde{\lambda}_i \tilde{\lambda}_j \tilde{g}_{ji} = -\tilde{\lambda}_i t_{ji}, \tilde{\lambda}_j^2 \tilde{f}_{ji} + \tilde{\lambda}_i \tilde{\lambda}_j \tilde{g}_{ij} = -\tilde{\lambda}_j t_{ij},$$

сложим эти два равенства

$$\tilde{\lambda}_i^2 \tilde{f}_{ij} + \tilde{\lambda}_j^2 \tilde{f}_{ji} + \tilde{\lambda}_i \tilde{\lambda}_j (\tilde{g}_{ji} + \tilde{g}_{ij}) = -\tilde{\lambda}_i t_{ji} - \tilde{\lambda}_j t_{ij}.$$

Подставим в полученное выражение (12):

$$\tilde{\lambda}_i^2 \tilde{f}_{ij} + \tilde{\lambda}_j^2 \tilde{f}_{ji} = -\tilde{\lambda}_i t_{ji} - \tilde{\lambda}_j t_{ij} - \tilde{\lambda}_i \tilde{\lambda}_j s_{ij}.$$

Соединим полученное выражение с (11):

$$(\tilde{\lambda}_j^2 - \tilde{\lambda}_i^2) \tilde{f}_{ij} = \tilde{\lambda}_j (t_{ij} + \tilde{\lambda}_i r_{ij}) + \tilde{\lambda}_i (t_{ji} + \tilde{\lambda}_j s_{ij}).$$

Аналогично, используя (11)-(14) мы получаем:

$$(\tilde{\lambda}_j^2 - \tilde{\lambda}_i^2) \tilde{g}_{ij} = \tilde{\lambda}_i (t_{ji} + \tilde{\lambda}_j r_{ij}) + \tilde{\lambda}_j (t_{ij} + \tilde{\lambda}_i s_{ij}).$$

Отсюда выражаем \tilde{f}_{ij} и \tilde{g}_{ij} :

$$\tilde{f}_{ij} = \frac{\alpha \tilde{\lambda}_j + \beta \tilde{\lambda}_i}{\tilde{\lambda}_j^2 - \tilde{\lambda}_i^2}, \tilde{g}_{ij} = \frac{\alpha \tilde{\lambda}_i + \beta \tilde{\lambda}_j}{\tilde{\lambda}_j^2 - \tilde{\lambda}_i^2}, \text{ при } \tilde{\lambda}_i \neq \tilde{\lambda}_j, 1 \leq i \leq n, n+1 \leq j \leq m,$$

при $\alpha := t_{ij} + \tilde{\lambda}_j r_{ij}, \beta := t_{ji} + \tilde{\lambda}_j s_{ij}$. Используя (10), можно выразить \tilde{F}_{12} :

$$\tilde{f}_{ij} = -\frac{t_{ji}}{\tilde{\lambda}_i}, \tilde{\lambda}_i \neq 0, 1 \leq i \leq n, n+1 \leq j \leq m.$$

С помощью этого равенства можно получить значения $\tilde{F}_{21} = R_{21} - \tilde{F}_{12}^T$:

$$\tilde{f}_{ij} = r_{ij} - \tilde{f}_{ji} = r_{ij} + \frac{t_{ij}}{\tilde{\lambda}_j}, \tilde{\lambda}_j \neq 0, n+1 \leq i \leq m, 1 \leq j \leq n.$$

И наконец, можно выразить \tilde{F}_{22} :

$$\tilde{f}_{ij} = \frac{r_{ij}}{2}, n+1 \leq i, j \leq m, i \neq j.$$

Из полученных равенств можно составить алгоритм, представленный в листинге 1. Это и есть алгоритм итеративного уточнения для SVD, основанный на матричном умножении.

При реализации данного алгоритма нужно учитывать, что он рассчитан на поиск разложения только в том случае, если собственные значения матрицы не совпадают друг с другом, т.е. $\tilde{\lambda}_i \neq \tilde{\lambda}_j, i \neq j$. В случае, если это не так, нужно обработать исключение, например, как в [2]. Также необходимо отметить, что для работы данного алгоритма необходима высокая арифметическая точность вычислений.

Входные данные: $A \in \mathbb{R}^{n \times m}$, где $m \geq n$, $\hat{U} \in \mathbb{R}^{m \times m}$, $\hat{V} \in \mathbb{R}^{n \times n}$

Выходные данные: $\hat{U}' \in \mathbb{R}^{m \times m}$, $\hat{V}' \in \mathbb{R}^{n \times n}$, $\hat{\Sigma}' = \text{diag}(\tilde{\lambda}_i) \in \mathbb{R}^{n \times m}$

- 1) Поиск R, S, T : $R \leftarrow I_m - \hat{U}^T \hat{U}$; $S \leftarrow I_n - \hat{V}^T \hat{V}$; $T \leftarrow \hat{U}^T A \hat{V}$
- 2) Расчет приближенных собственных значений: $\tilde{\lambda}_i \leftarrow t_{ii} / (1 - \frac{r_{ii} + s_{ii}}{2})$ для $i = 1, \dots, n$
- 3) Расчет диагональных элементов \tilde{F}_{11} и \tilde{G} : $\tilde{f}_{ii} \leftarrow r_{ii}/2$; $\tilde{g}_{ii} \leftarrow s_{ii}/2$ для $i = 1, \dots, n$
- 4) Расчет недиагональных элементов \tilde{F}_{11} и \tilde{G} : $\{\alpha \leftarrow t_{ij} + \tilde{\lambda}_j r_{ij}; \beta \leftarrow t_{ji} + \tilde{\lambda}_j s_{ij}; \tilde{f}_{ij} \leftarrow \frac{\alpha \tilde{\lambda}_j + \beta \tilde{\lambda}_i}{\tilde{\lambda}_j^2 - \tilde{\lambda}_i^2}; \tilde{g}_{ij} \leftarrow \frac{\alpha \tilde{\lambda}_i + \beta \tilde{\lambda}_j}{\tilde{\lambda}_j^2 - \tilde{\lambda}_i^2}\}$ для $1 \leq i, j \leq n, i \neq j$
- 5) Расчет \tilde{F}_{12} : $\tilde{f}_{ij} \leftarrow -t_{ji}/\tilde{\lambda}_i$ для $1 \leq i \leq n, n+1 \leq j \leq m$
- 6) Расчет \tilde{F}_{21} : $\tilde{f}_{ij} \leftarrow r_{ij} - \tilde{f}_{ij}$ для $n+1 \leq i \leq m, 1 \leq j \leq n$
- 7) Расчет \tilde{F}_{22} : $\tilde{f}_{ij} \leftarrow r_{ij}/2$ для $n+1 \leq i, j \leq m$
- 8) Нахождение новых значений для \hat{U} и \hat{V} : $\hat{U}' \leftarrow \hat{U} + \hat{U} \tilde{F}$; $\hat{V}' \leftarrow \hat{V} + \hat{V} \tilde{G}$

Листинг 1. Алгоритм *RefSVD* уточнения для SVD, основанный на матричном умножении.

Алгоритм имеет квадратичную сходимость. Доказательство данного факта представлено в [1].

Тестирование алгоритма

Для анализа работы реализованного алгоритма было проведено тестирование. Для тестирования было сгенерировано точное сингулярное разложение для матрицы $A = U \Sigma V^T$ размеров 4×3 , 5×5 , 6×4 , 52×50 , со случайными сингулярными значениями из интервала $[0, 10]$ или $[1, 100]$. Далее к этому разложению был добавлен шум: отклонение от начального значения на 10^{-3} , 10^{-15} , 10^{-18} , 10^{-33} . Далее к полученной матрице из зашумленных данных применялся алгоритм. Для каждого из экспериментов выводятся данные: размер матрицы A ; интервал значений сингулярных чисел; добавляемая ошибка; число итераций, а также, как результат теста, выводятся норма разницы исходной и восстановленной матриц $\|A - \hat{U} \hat{\Sigma} \hat{V}^T\|$ в пространствах L_1 и L_2 , то есть нормы $\|X\|_1 =$

$\sum_{i=1}^n \sum_{j=1}^m x_{ij}$ и $\|X\|_2 = \sqrt{\sum_{i=1}^n \sum_{j=1}^m x_{ij}^2}$ соответственно; нормы $\|\cdot\|_2$ разниц изначальных и уточненных матриц U, Σ, V ; время на выполнение алгоритма. Результаты тестирования представлены в `svd_results.csv`.

Список литературы

1. Ogita T., Aishima K. Iterative refinement for singular value decomposition based on matrix multiplication // Journal of Computational and Applied Mathematics. 2019. DOI: <https://doi.org/10.1016/j.cam.2019.112512>.
2. Ogita T., Aishima K. Iterative refinement for symmetric eigenvalue decomposition // Japan J. Indust. Appl. Math. — 2018. — DOI: <https://doi.org/10.1007/s13160-018-0310-3>.