Rozkład SVD macierzy
Singular Value Decomposition

Karol Kuczmarz, Jakub Kuciński

16 maja 2020

1 Wprowadzenie

Definicja 1.1. Niech $A \in M_{n\times n}$. Wektor $V \in \mathbb{R}^n$ nazywamy wektorem własnym A, jeżeli

$$(\exists \lambda \in \mathbb{R})(Av = \lambda v)$$

Definicja 1.2. Wartością własną macierzy $A \in M_{m\times n}(\mathbb{R})$ nazywamy $\lambda \in \mathbb{R}$ takie, że istnieje niezerowy wektor własny dla λ (tzn. $(\exists v \in \mathbb{R}^n \setminus \{0\})(Av = \lambda v)$).

Twierdzenie 1.3 (Diagonalizacja macierzy)**.** *Jeżeli macierz $A \in M_{n\times n}(\mathbb{R})$ ma n liniowo niezależnych wektorów własnych $v_1, \ldots, v_n \in \mathbb{R}^n$ i $\lambda_1, \ldots, \lambda_n$ to odpowiadające im wartości własne, to macierz A możemy przedstawić jako*

$$A = PDP^{-1}$$

gdzie:

- $P = (v_1, \ldots, v_n)$
- $D = \begin{pmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{pmatrix}$ *diagonalna.*

Uwaga 1.4. *Nie wszystkie macierze można diagonalizować. Można przejść w liczby zespolone, ale dalej możemy mieć wielokrotne wartości własne. Uogólnienie – Twierdzenie Jordana.*

Do rozkładu SVD potrzebne jest przypomnienie własności i nazewnictwa dla macierzy symetrycznych ($A^T = A$). Będziemy posługiwać się standardowym iloczynem skalarnym w $\mathbb{R}^n$, tzn. dla $v, u \in \mathbb{R}^n$ mamy $u \circ v = \sum_{i=1}^n u_i \cdot v_i = u^T v$.

Definicja 1.5. Macierz kwadratową $A \in M_{n\times n}(\mathbb{R})$ nazywamy macierzą ortogonalną, jeżeli zachodzi $AA^T = A^T A = I$.

Uwaga 1.6. *Iloczyn macierzy ortogonalnych A i B jest macierzą ortogonalną.* $(AB)(AB)^T = ABB^T A^T = AA^T = I$*,* $(AB)^T(AB) = B^T A^A BT = BB^T = I$*.*

Definicja 1.7. Zbiór wektorów będący bazą przestrzeni $\mathbb{R}^n$ taki, że wektory są parami prostopadłe i mające długość 1 nazywamy bazą ortonormalną.

Uwaga 1.8. *Ortogonalności macierzy są równoważne warunki:*

- *kolumny A stanowią bazę ortonormalną $\mathbb{R}^n$*
- *wiersze A stanowią bazę ortonormalną $\mathbb{R}^n$*

Twierdzenie 1.9 (Twierdzenie spektralne)**.** *Każda symetryczna macierz $A \in M_{n\times n}(\mathbb{R})$ diagonalizuje się w pewnej bazie ortonormalnej i wszystkie jej wartości własne są rzeczywiste.*

Definicja 1.10. Macierz symetryczna $A \in M_{n\times n}(\mathbb{R})$ jest dodatnio określona, jeżeli dla każdego $v \in \mathbb{R}^n \setminus \{0\}$ zachodzi

$$v \circ Av > 0$$

Jeżeli nierówność jest słaba, to mówimy, że macierz jest dodatnio półokreślona.

Uwaga 1.11. *Dodatnia określoność (półokreśloność) jest równoznaczna ze wszystkimi dodatnimi (nieujemnymi) wartościami własnymi macierzy. Z twierdzenia spektralnego mamy $A = PDP^T$. Wtedy*

$$v \circ Av = v \circ PDP^T v = (P^T v) \circ D(P^T v) > 0$$

Skoro P^T jest ortogonalna, to jest macierzą zmiany bazy, czyli powyższa równość jest tożsama z

$$v \circ Dv > 0$$

$$v \circ Dv = \sum_{i=1}^{n} \lambda_i \cdot v_i^2$$

Lemat 1.12. *Dla dowolnej macierzy $A \in M_{m\times n}(\mathbb{R})$ macierze AA^T i A^TA są symetryczne i dodatnio półokreślone.*

Dowód. Macierze A^TA i AA^T są odpowiednio wymiaru $n \times n$ i $m \times m$.

Najpierw pokażemy symetryczność.

$$(AA^T)^T = (A^T)^T A^T = AA^T$$

$$(A^TA)^T = A^T(A^T)^T = A^TA$$

Pokażemy dodatnią półokreśloność A^TA, dowód dla macierzy AA^T jest analogiczny. Weźmy dowolny wektor $v \in \mathbb{R}^n \setminus \{0\}$.

$$v \circ A^TAv = Av \circ Av \geq 0$$

z definicji iloczynu skalarnego. □

2 Główne twierdzenie

Zanim podamy twierdzenie najpierw udowodnimy pomocniczy lemat.

Lemat 2.1. *Dla $A \in M_{m\times n}(\mathbb{R})$ zachodzi $rank(A^TA) \leq \min(m, n)$.*

Dowód. Wiemy, że dla przekształcenia liniowego o macierzy B mamy $rank(B) = ImF_B$, czyli rząd macierzy B jest równy wymiarowi obrazu przekształcenia zadanego tą macierzą.

Przekształcenie zadane macierzą A^TA jest pomiędzy następującymi przestrzeniami: $F_{A^TA} : \mathbb{R}^n \to \mathbb{R}^m \to \mathbb{R}^n$. Stąd $rank(A^TA) = ImF_{A^TA} \leq \min(m, n)$. □

Twierdzenie 2.2 (Rozkład SVD)**.** *Każdą macierz $A \in M_{m\times n}(\mathbb{R})$ można przedstawić w postaci iloczynu $A = U\Sigma V^T$, gdzie:*

- *$U \in M_{m\times m}(\mathbb{R})$ jest macierzą ortogonalną*
- *$\Sigma \in M_{m\times n}(\mathbb{R})$ jest macierzą postaci $\begin{pmatrix} D & 0 \end{pmatrix}$ lub $\begin{pmatrix} D \\ 0 \end{pmatrix}$ dla D diagonalnej z nieujemnymi wyrazami na przekątnej uporządkowanymi nierosnąco*

- $V \in M_{n \times n}(\mathbb{R})$ *jest macierzą ortogonalną.*

Dowód. Wiemy, że macierz A^TA jest symetryczna i dodatnio półokreślona. Z twierdzenia spektralnego macierz A^TA możemy zdiagonalizować w bazie ortonormalnej, czyli $A^TA = V\bar{D}V^T$, gdzie V jest ortogonalna a $\bar{D}$ jest diagonalna.

Możemy bez straty ogólności założyć, że macierz $\bar{D} = \begin{pmatrix} D & 0 \\ 0 & 0 \end{pmatrix}$, gdzie D jest diagonalna z dodatnimi uporządkowanymi nierosnąco wyrazami na przekątnej. $D \in M_{l \times l}(\mathbb{R})$ dla $l \leq \min(m, n)$, ponieważ $l = rank(A^TA)$ (później *Lemat 2.1*).

Niech $V = \begin{pmatrix} V_1 & V_2 \end{pmatrix}$, gdzie V_1 to macierz wektorów odpowiadających niezerowym wartościom własnym natomiast V_2 to macierz wektorów odpowiadających zerowym wartościom własnym.

Mamy

$$A^TA = V\bar{D}V^T$$

$$V^TA^TAV = \bar{D} = \begin{pmatrix} D & 0 \\ 0 & 0 \end{pmatrix}$$

$$\begin{pmatrix} V_1^T \\ V_2^T \end{pmatrix} A^TA \begin{pmatrix} V_1 & V_2 \end{pmatrix} = \begin{pmatrix} V_1^TA^TA \\ V_2^TA^TA \end{pmatrix} \begin{pmatrix} V_1 & V_2 \end{pmatrix} = \begin{pmatrix} V_1^TA^TAV_1 & V_1^TA^TAV_2 \\ V_2^TA^TAV_1 & V_2^TA^TAV_2 \end{pmatrix}$$

Zatem $D = V_1^TA^TAV_1$.

Niech $U_1 = AV_1D^{-\frac{1}{2}}$ (odwracamy i pierwiastkujemy wyrazy z przekątnej macierzy). Zauważmy, że

$$U_1D^{\frac{1}{2}}V_1^T = AV_1D^{-\frac{1}{2}}D^{\frac{1}{2}}V_1^T = AV_1V_1^T = A$$

Pokażemy, że wektory macierzy U_1 są ortogonalne i długości 1. Rozważmy iloczyn $U_1^TU_1$.

$$U_1^TU_1 = (D^{-\frac{1}{2}})^TV_1^TA^TAV^1D^{-\frac{1}{2}}$$

Mamy $D = V_1^TA^TAV_1$. Stąd

$$U_1^TU_1 = D^{-\frac{1}{2}}DD^{-\frac{1}{2}} = I$$

Macierz U_1 możemy też zapisać jako $U_1 = \begin{pmatrix} u_1 & u_2 & \dots & u_l \end{pmatrix}$. Wtedy

$$U_1^TU_1 = \begin{pmatrix} u_1^T \\ \vdots \\ u_l^T \end{pmatrix} \begin{pmatrix} u_1 & \dots & u_l \end{pmatrix} = \begin{pmatrix} < u_1, u_1 > & \dots & < u_1, u_n > \\ \vdots & \ddots & \vdots \\ < u_n, u_1 > & \dots & < u_n, u_n > \end{pmatrix} = \begin{pmatrix} 1 & & \\ & \ddots & \\ & & 1 \end{pmatrix}$$

Czyli wektory $u_1, \dots, u_n$ są parami prostopadłe i długości 1, zatem możemy uzupełnić je do bazy ortonormalnej przestrzeni $\mathbb{R}^m$ wektorami będącymi kolumnami macierzy U_2. Wtedy macierz $U = \begin{pmatrix} U_1 & U_2 \end{pmatrix}$ jest macierzą ortogonalną.

Niech $\Sigma = \begin{pmatrix} D^{\frac{1}{2}} & 0 \\ 0 & 0 \end{pmatrix}$. Uzupełniamy macierz D zerami z prawej i z dołu tak, by była wymiaru $m \times n$. Wtedy

$$U\Sigma V^T = \begin{pmatrix} U_1 & U_2 \end{pmatrix} \begin{pmatrix} D^{\frac{1}{2}} & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} V_1^T \\ V_2^T \end{pmatrix} = \begin{pmatrix} U_1D^{\frac{1}{2}} & 0 \end{pmatrix} \begin{pmatrix} V_1^T \\ V_2^T \end{pmatrix} = U_1D^{\frac{1}{2}}V_1^T = A$$

□

Uwaga 2.3. *Macierz U jest ortogonalną macierzą wektorów własnych macierzy AA^T.*

Dowód. Wystarczy pokazać, że macierz U_1 jest macierzą wektorów własnych AA^T, ponieważ U_2 dobieramy dowolnie tak, by stworzyć bazę ortonormalną (możemy zatem wykorzystać do tego wektory własne). Już wiemy, że te wektory są wzajemnie prostopadłe i niezerowe, więc są liniowo niezależne. Weźmy dowolny wektor z macierzy U_1. Niech $u_i = Av_i\frac{1}{\lambda_i}$.

$$AA^T(Av_i)\frac{1}{\lambda_i} = \frac{1}{\lambda_i}A(A^TA)v_i$$

Wektor v_i jest wektorem własnym macierzy A^TA dla wartości własnej λ_i, zatem

$$\frac{1}{\lambda_i}A(A^TA)v_i = \frac{1}{\lambda_i}A\lambda_i v_i = \sqrt{\lambda_i}Av_i$$

Stąd wektor u_i jest wektorem własnym macierzy AA^T dla wartości własnej λ_i. $\square$

3 Numeryczne wyznaczanie rozkładu SVD

Rozkład SVD wykonujemy w dwóch krokach. Najpierw sprowadzamy macierz do postaci dwudiagonalnej, tzn. macierz ma wszystkie wyrazy zerowe oprócz przekątnej i wyrazów tuż nad przekątną

$$\begin{pmatrix} * & * & & & \\ & * & * & & \\ & & * & \ddots & \\ & & & \ddots & * \\ & & & & * \end{pmatrix}$$

Następnie dokonujemy rozkładu SVD na macierzy bidiagonalnej.

Oba kroki będą polegały na tym, że będziemy mnożyć naszą wyjściową macierz przez macierze ortogonalne o odpowiednim wymiarze, w taki sposób, by na końcu otrzymać macierz diagonalną.

Nasza wyjściowa macierz to $A \in M_{m\times n}(\mathbb{R})$. Zakładamy, że $m \geq n$, ponieważ przeciwny przypadek możemy rozwiązać po prostu przez transpozycję macierzy A, policzenie jej rozkładu SVD i ponowną transpozycję otrzymanego wyniku.

3.1 Bidiagonalizacja

Do bidiagonalizacji będziemy wykorzystywać macierze transformacji Householdera. Wyraża się je wzorem

$$A = I - 2vv^t$$

gdzie v jest wektorem długości 1. Wtedy macierz A jest macierzą ortogonalną.

Metoda polega na tym, że po kolei będziemy zerować fragmenty odpowiednich kolumn i wierszy. W pierwszym kroku mnożymy naszą macierz przez taką macierz P_1, że

$$P_1 \cdot \begin{pmatrix} * & * & \dots & * \\ * & * & \dots & * \\ \vdots & \vdots & \ddots & \vdots \\ * & * & \dots & * \end{pmatrix} = \begin{pmatrix} * & * & \dots & * \\ 0 & * & \dots & * \\ \vdots & \vdots & \ddots & \vdots \\ 0 & * & \dots & * \end{pmatrix}$$

Następnie z prawej strony mnożymy przez macierz Q_1 taką, że

$$\begin{pmatrix} * & * & * & \dots & * \\ 0 & * & * & \dots & * \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & * & * & \dots & * \end{pmatrix} \cdot Q_1 = \begin{pmatrix} * & * & 0 & \dots & 0 \\ 0 & * & * & \dots & * \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & * & * & \dots & * \end{pmatrix}$$

Gdy chcemy wyzerować k-tą kolumnę (P_k), k-ty wiersz (Q_k), to macierze Householdera wyrażają się wzorem

$$P_k = I - 2x^{(k)}{x^{(k)}}^T$$

$$Q_k = I - 2y^{(k)}{y^{(k)}}^T$$

dla wektorów $x^{(k)}$, $y^{(k)} \in \mathbb{R}^n$.

Nie chcemy ingerować w już wyzerowane wyrazy, dlatego

$$x_i^{(k)} = 0 \text{ dla } i = 1, \ldots, k-1$$

$$y_j^{(k)} = 0 \text{ dla } j = 1, \ldots, k$$

Szczegółowe opisanie wyznaczania pozostałych wyrazów nie wnosiłoby niczego szczególnego do idei algorytmu, więc można prześledzić je w programie lub w pracy, w której została przedstawiona ta metoda.

Niech

$$P = P_n \ldots P_1$$

$$Q = Q_1 \ldots Q_n$$

Zatem $PAQ = J$, gdzie J bidiagonalna.

3.2 Rozkład SVD macierzy bidiagonalnej

Jeżeli otrzymamy rozkład SVD macierzy J, czyli $J = S\Sigma T$, to wtedy $A = (P^T S)\Sigma(TQ^T)$ będzie rozkładem macierzy A.

3.2.1 Macierze obrotu Givensa

Ustalmy, że jesteśmy w przestrzeni $\mathbb{R}^n$. Macierzą obrotu Givensa nazywamy macierz $G(i, j, \theta) \in M_{n\times n}(\mathbb{R})$ (dla $i < j$ i $\theta \in [0, 2\pi]$) w następującej postaci:

$$G(i,j,\theta) = \begin{pmatrix} 1 & \ldots & 0 & \ldots & 0 & \ldots & 0 \\ \vdots & \ddots & \vdots & & \vdots & & \vdots \\ 0 & \ldots & c & \ldots & -s & \ldots & 0 \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 0 & \ldots & s & \ldots & c & \ldots & 0 \\ \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & \ldots & 0 & \ldots & 0 & \ldots & 1 \end{pmatrix}$$

gdzie $c = \cos(\theta)$ i $s = \sin(\theta)$.

Dokładniej $G(i, j, \theta)$ jest macierzą identycznościową z wyjątkiem wyrazów

$$g_{i,i} = g_{j,j} = c$$

$$g_{i,j} = -g_{j,i} = s$$

Zauważmy, że ta macierz jest ortogonalna.

Będziemy korzystać z macierzy Givensa, gdy będziemy chcieli wyzerować konkretny wyraz w macierzy. Taki sposób jest atrakcyjny, ponieważ macierz ta jest ortogonalna oraz skoro jest z bliska macierzy identycznościowej, to w wyniku mnożenia macierzy przez macierz Givensa będziemy mogli większą część wyrazów po prostu przepisać.

Przemnożenie przez macierz obrotu Givensa z lewej strony modyfikuje tylko i-ty i j-ty wiersz, zaś z prawej – i-tą i j-tą kolumnę.

3.2.2 Algorytm

Nasza metoda będzie teoretycznie nieskończona, tzn. po wykonaniu nieskończenie wielu iteracji otrzymalibyśmy dokładny wynik, ale na komputerze możemy przerwać obliczenia, gdy otrzymamy satysfakcjonującą nas dokładność bądź gdy nie będziemy już w stanie osiągnąć lepszej (z powodu precyzji arytmetyki na liczbach zmiennoprzecinkowych). Chcemy doprowadzić macierz J do postaci diagonalnej.

Opiszemy jedną iterację algorytmu. Po każdej iteracji powinniśmy dalej mieć macierz bidiagonalną. Będziemy mnożyć macierz J naprzemiennie z lewej i prawej strony przez macierze obrotu Givensa.

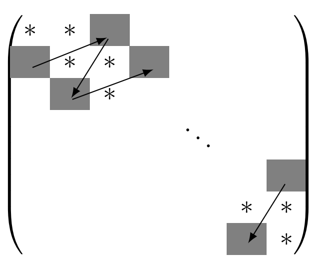
Najpierw mnożymy z prawej strony przez macierz $T_2 = G(1, 2, \theta)$. Sposób wybierania wartości θ może przyspieszać zbieżność, ale można przyjąć, że θ jest dowolny. Przemnożenie J przez T_2 generuje wartość na miejscu $(2, 1)$ macierzy.

$$\begin{pmatrix} * & * & 0 & \dots & 0 \\ 0 & * & * & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & * \end{pmatrix} \cdot T_2 = \begin{pmatrix} * & * & 0 & \dots & 0 \\ * & * & * & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & * \end{pmatrix}$$

Następnie dobieramy macierz obrotu Givensa S_2 tak, by po przemnożeniu z lewej strony wyzerować element $(2, 1)$. Jednak takie działanie zmieni wartość elementu na miejscu $(2, 4)$ i ten element wyzerujemy macierzą T_3, itd.

$$S_2 \cdot \begin{pmatrix} * & * & 0 & \dots & 0 \\ * & * & * & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & * \end{pmatrix} = \begin{pmatrix} * & * & * & \dots & 0 \\ 0 & * & * & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & * \end{pmatrix}$$

Można to zobrazować następująco:



Przy mnożeniu przez macierz obrotu Givensa kasujemy jeden wyraz spoza dwóch przekątnych, ale tworzymy następny. "Gonimy" niezerowy wyraz tak długo, aż nie znajdzie się "poza macierzą".

Zatem w jednej iteracji macierz J zamieniamy na

$$(S_n \dots S_2) J (T_2 \dots T_n)$$

Gdy wyrazy poza przekątną będą wystarczająco małe, to możemy wstawić za nie 0. Proces powtarzamy tak długo, aż zostanie nam macierz diagonalna. Musimy spamiętywać i aktualizować (poprzez mnożenie po każdej iteracji) macierze ortogonalne $S = S_n \dots S_2$ i $T = T_2 \dots T_n$.

4 Zastosowania rozkładu SVD

4.1 Kompresja danych

Niech $A \in M_{m\times n}(\mathbb{R})$ i $A = U\Sigma V^T$ (SVD). Załóżmy, że

$$\Sigma = \begin{pmatrix} \sigma_1 & & & 0 & \dots & 0 \\ & \ddots & & \vdots & & \vdots \\ & & \sigma_r & 0 & \dots & 0 \\ 0 & \dots & 0 & 0 & \dots & 0 \\ \vdots & & \vdots & \vdots & & \vdots \\ 0 & \dots & 0 & 0 & \dots & 0 \end{pmatrix}$$

Oczywiście Σ jest macierzą prostokątną. Zapiszmy macierze U i V jako $U = \begin{pmatrix} u_1 & \dots & u_m \end{pmatrix}$ i $V = \begin{pmatrix} v_1 & \dots & v_n \end{pmatrix}$.

Wtedy

$$A = U\Sigma V^T = \begin{pmatrix} u_1 & \dots & u_m \end{pmatrix} \Sigma \begin{pmatrix} v_1^T \\ \vdots \\ v_n^T \end{pmatrix} = \sigma_1 u_1 v_1^T + \sigma_2 u_2 v_2^T + \ldots + \sigma_r u_r v_r^T$$

Skoro wartości na przekątnej macierzy Σ są uporządkowane nierosnąco oraz wektory u_i oraz v_i są długości 1, to usunięcie ostatnich niezerowych wartości szczególnych oraz odpowiadających im wektorów z macierzy U i V powinno spowodować niewielką zmianę w macierzy A, a zmniejszy to ilość pamięci potrzebnej do przechowywania danych. Jeżeli wzięlibyśmy pierwszych l wyrazów, to potrzebowalibyśmy tylko pamiętać wektory $u_1, \ldots, u_l$, $v_1, \ldots, v_l$ i wyrazy $\sigma_1, \ldots, \sigma_l$.

Dane często są przechowywane w postaci bardzo dużych prostokątnych macierzy, więc przedstawienie ich w postaci zredukowanego rozkładu SVD może przynieść dużą oszczędność pamięciową.

Literatura

[1] G. H. Golub and C. Reinsch *Handbook Series Linear Algebra. Singular Value Decomposition and Least Squares Solutions.* Numer. Math. 14, 403–420 (1970)

[2] G. Golub and W. Kahan, *Calculating the Singular Values and Pseudo-inverse of a Matrix.* J. Siam Numer. Anal., Ser. B, Vol. 2, No. 2. Printed in U.S.A.. 1965

[3] *Singular value decomposition.* Wikipedia: The Free Encyklopedia (16.05.2020) `https://en.wikipedia.org/wiki/Singular_value_decomposition`

[4] *Understanding Singular Value Decomposition and its Application in Data Science.* (16.05.2020) `https://towardsdatascience.com/understanding-singular-value-decomposition-and-its-application-in-data-science-388a54be95d`

[5] *Singular Value Decomposition as Simply as Possible* (16.05.2020) `https://gregorygundersen.com/blog/2018/12/10/svd/`

[6] A. Kiełbasiński and Hubert Schwetlick, *Numeryczna algebra liniowa.* Wydawnictwo Naukowo-Techniczne

[7] Åke Björck and Germund Dahlquist, *Numerical Methods in Scientific Computing. Volume II.*

[8] *MATH2071: LAB #9: The Singular Value Decomposition.* (16.05.2020)
http://www.math.pitt.edu/ sussmanm/2071Spring08/lab09/index.html

[9] *Wikiwand. Singular value decomposition.* (16.05.2020)
https://www.wikiwand.com/en/Singular_value_decomposition?fbclid=IwAR0_B1ohQBsQnXalAqx869TnLd2jWU2Mc2TQTwnkXkV_pMLyT3H189kIbz4#