Wstęp do metod numerycznych

SVD, metody iteracyjne i metoda gradientów sprzężonych

P. F. Góra

http://th-www.if.uj.edu.pl/zfs/gora/

2011

Współczynnik uwarunkowania macierzy symetrycznej

Twierdzenie 1. *Niech* $\mathbf{A} \in \mathbb{R}^{N\times N}$ *będzie macierzą symetryczną,* $\mathbf{A} = \mathbf{A}^T$*, i niech liczby* $\{\lambda_i\}_{i=1}^N$ *będą jej wartościami własnymi. Jeżeli* $\det \mathbf{A} \neq 0$*, współczynnik uwarunkowania tej macierzy spełnia*

$$\kappa = \frac{\max\limits_i |\lambda_i|}{\min\limits_i |\lambda_i|} \,. \tag{1}$$

Dowód. W celu udowodnienia tego twierdzenia obliczmy normę macierzy $\mathbf{A}$. Ponieważ jest to macierz symetryczna i rzeczywista, jej wartości własne są rzeczywiste, natomiast jej unormowane wektory własne tworzą bazę w $\mathbb{R}^N$. Oznaczmy przez $\mathbf{y}_i$ jej i-ty wektor własny, $\mathbf{A}\mathbf{y}_i = \lambda_i \mathbf{y}_i$. Każdy

wektor $\mathbf{x} \in \mathbb{R}^N$, $\|\mathbf{x}\| = 1$, można przedstawić jako kombinację liniową

$$x = \sum_{i=1}^{N} \alpha_i \mathbf{y}_i \,, \tag{2}$$

przy czym warunek unormowania prowadzi do następujące więzu na współczynniki tej kombinacji:

$$\sum_{i=1}^{N} \alpha_i^2 = 1 \,. \tag{3}$$

Obliczmy teraz

$$\begin{aligned}
\|\mathbf{Ax}\|^2 &= \left\|\mathbf{A}\sum_{i=1}^{N}\alpha_i\mathbf{y}_i\right\|^2 = \left\|\sum_{i=1}^{N}\alpha_i\mathbf{A}\mathbf{y}_i\right\|^2 = \left\|\sum_{i=1}^{N}\alpha_i\lambda_i\mathbf{y}_i\right\|^2 \\
&\leqslant \sum_{i=1}^{N}||\alpha_i\lambda_i\mathbf{y}_i||^2 = \sum_{i=1}^{N}\alpha_i^2\lambda_i^2 \leqslant \sum_{i=1}^{N}\alpha_i^2 \max_i \lambda_i^2 \\
&= \max_i \lambda_i^2 \sum_{i=1}^{N}\alpha_i^2 = \left(\max_i |\lambda_i|\right)^2 \qquad (4)
\end{aligned}$$

Widzimy zatem, iż $\forall\, \mathbf{x} \in \mathbb{R}^N$, $\|\mathbf{x}\|^2 = 1$, zachodzi $\|\mathbf{A}\mathbf{x}\| \leqslant \max_i |\lambda_i|$, a zatem na mocy definicji normy indukowanej, $\|\mathbf{A}\| = \max_i |\lambda_i|$.

Ponieważ $\det \mathbf{A} \neq 0$, macierz $\mathbf{A}^{-1}$ istnieje, jest symetryczna i rzeczywista, a jej wartościami własnymi są liczby $\{1/\lambda_i\}_{i=1}^N$. Zupełnie analogicznie dowodzimy, iż $\|\mathbf{A}^{-1}\| = \max_i (1/|\lambda_i|) = 1/\left(\min_i |\lambda_i|\right)$, skąd natychmiast wynika teza (1). □

Singular Value Decomposition

Twierdzenie 2. *Dla każdej macierzy* $\mathbf{A} \in \mathbb{R}^{M \times N}$, $M \geqslant N$, *istnieje rozkład*

$$\mathbf{A} = \mathbf{U}\,[\mathrm{diag}(w_i)]\,\mathbf{V}^T, \tag{5}$$

gdzie $\mathbf{U} \in \mathbb{R}^{M \times N}$ *jest macierzą kolumnowo ortogonalną,* $\mathbf{V} \in \mathbb{R}^{N \times N}$ *jest macierzą ortogonalną oraz* $w_i \in \mathbb{R}$, $i = 1, \ldots, N$. *Rozkład ten nazywamy* rozkładem względem wartości osobliwych *(Singular Value Decomposition, SVD). Jeżeli* $M = N$, *macierz* $\mathbf{U}$ *jest macierzą ortogonalną.*

Jądro i zasięg operatora

Niech $\mathbf{A} \in \mathbb{R}^{M \times N}$. *Jądrem operatora* $\mathbf{A}$ nazywam

$$\text{Ker}\, \mathbf{A} = \{\mathbf{x} \in \mathbb{R}^N : \mathbf{A}\mathbf{x} = 0\}\,. \tag{6}$$

Zasięgiem operatora $\mathbf{A}$ nazywam

$$\text{Range}\, \mathbf{A} = \{\mathbf{y} \in \mathbb{R}^M : \exists \mathbf{x} \in \mathbb{R}^N : \mathbf{A}\mathbf{x} = \mathbf{y}\}\,. \tag{7}$$

Jądro i zasięg operatora są przestrzeniami liniowymi. Jeśli $M = N < \infty$, $\dim(\text{Ker}\, \mathbf{A}) + \dim(\text{Range}\, \mathbf{A}) = N$.

Sens *SVD*

Sens *SVD* najlepiej widać w przypadku, w którym co najmniej jedna z wartości $w_i = 0$. Dla ustalenia uwagi przyjmijmy $w_1 = 0$, $w_{i\neq 1} \neq 0$.
Po pierwsze, co to jest $\mathbf{z} = [z_1, z_2, \ldots, z_n]^T = \mathbf{V}^T\mathbf{x}$? Ponieważ $\mathbf{V}$ jest macierzą ortogonalną, $\mathbf{z}$ jest rozkładem wektora $\mathbf{x}$ w bazie kolumn macierzy $\mathbf{V}$. Korzystając z (5), dostajemy

$$\mathbf{A}\mathbf{x} = \mathbf{U}\left[\mathrm{diag}(w_i)\right]\mathbf{V}^T\mathbf{x} = \mathbf{U}\left[\mathrm{diag}(0, w_2, \ldots, w_N)\right]\mathbf{z} = \mathbf{U}\begin{bmatrix} 0 \\ w_2 z_2 \\ \vdots \\ w_N z_N \end{bmatrix}. \qquad (8)$$

Wynikiem ostatniego mnożenia będzie pewien wektor z przestrzeni $\mathbb{R}^M$. Ponieważ pierwszym elementem wektora $[0, w_2 z_2, \ldots, w_N z_N]^T$ jest zero, wynik ten nie zależy od pierwszej kolumny macierzy $\mathbf{U}$. Widzimy zatem, że *kolumny macierzy $\mathbf{U}$, odpowiadające niezerowym współczynnikom w_i, stanowią bazę w zasięgu operatora $\mathbf{A}$*.
Co by zaś się stało, gdyby $\mathbf{x}$ był równoległy do wektora stanowiącego pierwszą kolumnę $\mathbf{V}$? Wówczas $\mathbf{z} = 0$, a wobec tego $\mathbf{A}\mathbf{x} = 0$. Ostatecznie więc widzimy, że *kolumny macierzy $\mathbf{V}$, odpowiadające zerowym współczynnikom w_i, stanowią bazę w jądrze operatora $\mathbf{A}$.*

SVD i odwrotność macierzy

Niech $\mathbf{A} \in \mathbb{R}^{N\times N}$. Zauważmy, że $|\det \mathbf{A}| = \prod_{i=1}^{N} w_i$, a zatem $\det \mathbf{A} = 0$ wtedy i tylko wtedy, gdy co najmniej jeden $w_i = 0$. Niech $\det \mathbf{A} \neq 0$. Wówczas równanie $\mathbf{A}\mathbf{x} = \mathbf{b}$ ma rozwiązanie postaci

$$\mathbf{x} = \mathbf{A}^{-1}\mathbf{b} = \mathbf{V}\left[\mathrm{diag}(w_i^{-1})\right]\mathbf{U}^T\mathbf{b}\,. \tag{9}$$

Niech teraz $\det \mathbf{A} = 0$. Równanie $\mathbf{A}\mathbf{x} = \mathbf{b}$ *także* ma rozwiązanie, o ile tylko $\mathbf{b} \in$ Range $\mathbf{A}$. Rozwiązanie to dane jest wzorem

$$\mathbf{x} = \tilde{\mathbf{A}}^{-1}\mathbf{b} = \mathbf{V}\left[\mathrm{diag}(\tilde{w}_i^{-1})\right]\mathbf{U}^T\mathbf{b}\,. \tag{10a}$$

gdzie

$$\tilde{w}_i^{-1} = \begin{cases} w_i^{-1} & \text{gdy } w_i \neq 0\,, \\ 0 & \text{gdy } w_i = 0\,. \end{cases} \tag{10b}$$

SVD i współczynnik uwarunkowania

Twierdzenie 3. *Jeżeli macierz* $\mathbf{A} \in \mathbb{R}^{N\times N}$ *posiada rozkład* (5) *oraz* $\det \mathbf{A} \neq 0$*, jej współczynnik uwarunkowania spełnia*

$$\kappa = \frac{\max\limits_i |w_i|}{\min\limits_i |w_i|}. \tag{11}$$

Jeśli macierz jest źle uwarunkowana, ale *formalnie* odwracalna, numeryczne rozwiązanie równania $\mathbf{Ax} = \mathbf{b}$ może być zdominowane przez wzmocniony błąd zaokrąglenia. Aby tego uniknąć, często zamiast (bezużytecznego!) rozwiązania dokładnego (9), używa się *przybliżonego* (i użytecznego!) rozwiązania w postaci (10) z następującą modyfikacją

$$\tilde{w}_i^{-1} = \begin{cases} w_i^{-1} & \text{gdy } |w_i| > \tau\,, \\ 0 & \text{gdy } |w_i| \leqslant \tau\,, \end{cases} \tag{12}$$

gdzie τ jest pewną zadaną tolerancją.

Metody iteracyjne

W metodach dokładnych otrzymane rozwiązanie jest dokładne z dokładnością do błędów zaokrąglenia, które, dodajmy, dla układów źle uwarunkowanych mogą być *znaczne*.

W metodach iteracyjnych rozwiązanie dokładne otrzymuje się, teoretycznie, w granicy nieskończenie wielu kroków — w praktyce liczymy na to, że po skończonej (i niewielkiej) ilości kroków zbliżymy się do wyniku ścisłego w granicach błędu zaokrąglenia.

Rozpatrzmy układ równań:

$$a_{11}x_1 + a_{12}x_2 + a_{13}x_3 = b_1 \quad (13a)$$
$$a_{21}x_1 + a_{22}x_2 + a_{23}x_3 = b_2 \quad (13b)$$
$$a_{31}x_1 + a_{32}x_2 + a_{33}x_3 = b_3 \quad (13c)$$

Przepiszmy ten układ w postaci

$$x_1 = (b_1 - a_{12}x_2 - a_{13}x_3)/a_{11} \quad (14a)$$
$$x_2 = (b_2 - a_{21}x_1 - a_{23}x_3)/a_{22} \quad (14b)$$
$$x_3 = (b_3 - a_{31}x_1 - a_{32}x_2)/a_{33} \quad (14c)$$

Gdyby po prawej stronie (14) były “stare” elementy x_j, a po lewej “nowe”, dostalibyśmy metodę iteracyjną

$$x_i^{(k+1)} = \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k)} - \sum_{j=i+1}^{N} a_{ij} x_j^{(k)} \right) \Big/ a_{ii} \qquad (15)$$

Górny indeks $x^{(k)}$ oznacza, ze jest to przybliżenie w k-tym kroku. Jest to tak zwana *metoda Jacobiego*.

Zauważmy, że w metodzie (15) nie wykorzystuje się najnowszych przybliżeń: Powiedzmy, obliczając $x_2^{(k+1)}$ korzystamy z $x_1^{(k)}$, mimo iż znane jest już wówczas $x_1^{(k+1)}$. (Za to metodę tę łatwo można zrównoleglić.) Sugeruje to następujące ulepszenie:

$$x_i^{(k+1)} = \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{N} a_{ij} x_j^{(k)} \right) \Big/ a_{ii} \qquad (16)$$

Jest to tak zwana metoda Gaussa-Seidela.

Jeżeli macierz $\mathbf{A} = \{a_{ij}\}$ jest rzadka, obie te metody iteracyjne będą efektywne *tylko i wyłącznie* wówczas, gdy we wzorach (15), (16) uwzględni się ich strukturę, to jest uniknie redundantnych mnożeń przez zera.

Trochę teorii

Metody Jacobiego i Gaussa-Seidela należą do ogólnej kategorii

$$\mathbf{M}\mathbf{x}^{(k+1)} = \mathbf{N}\mathbf{x}^{(k)} + \mathbf{b} \tag{17}$$

gdzie $\mathbf{A} = \mathbf{M} - \mathbf{N}$ jest *podziałem* (*splitting*) macierzy. Dla metody Jacobiego $\mathbf{M} = \mathbf{D}$ (część diagonalna), $\mathbf{N} = -(\mathbf{L} + \mathbf{U})$ (części pod- i ponaddiagonalne, bez przekątnej). Dla metody Gaussa-Seidela $\mathbf{M} = \mathbf{D} + \mathbf{L}$, $\mathbf{N} = -\mathbf{U}$. Rozwiązanie równania $\mathbf{A}\mathbf{x} = \mathbf{b}$ jest punktem stałym iteracji (17).

Definicja *Promieniem spektralnym* (diagonalizowalnej) macierzy $\mathbf{G}$ nazywam

$$\rho(\mathbf{G}) = \max\{|\lambda| : \exists \mathbf{y} \neq 0 : \mathbf{G}\mathbf{y} = \lambda\mathbf{y}\} \tag{18}$$

Twierdzenie 4. *Iteracja* (17) *jest zbieżna jeśli* $\det \mathbf{M} \neq 0$ *oraz* $\rho(\mathbf{M}^{-1}\mathbf{N}) < 1$.

Dowód. Przy tych założeniach iteracja (17) jest odwzorowaniem zwężającym. □

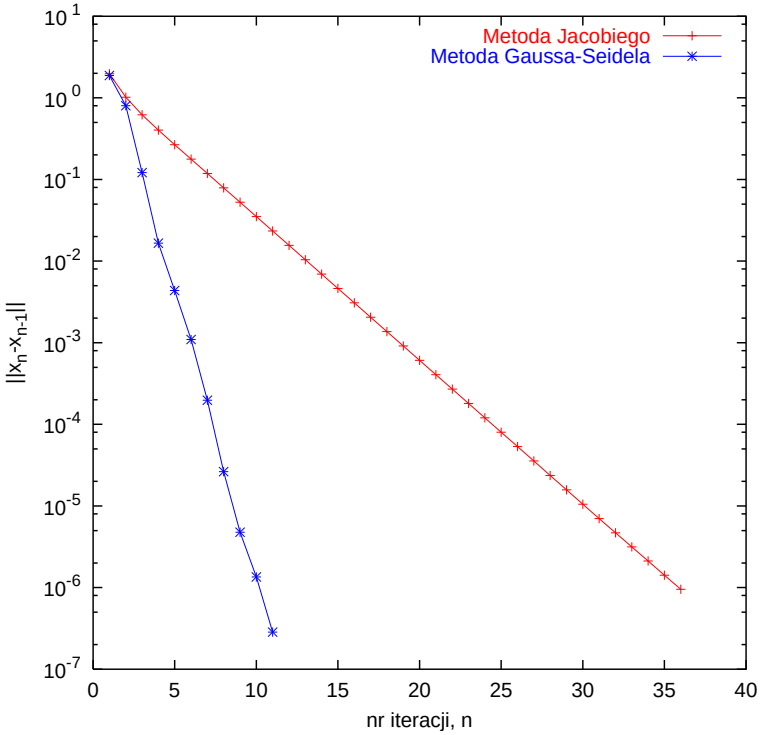
Twierdzenie 5. *Metoda Jacobiego jest zbieżna jeśli macierz* $\mathbf{A}$ *jest silnie diagonalnie dominująca.*

Twierdzenie 6. *Metoda Gaussa-Seidela jest zbieżna jeśli macierz* $\mathbf{A}$ *jest symetryczna i dodatnio określona.*

Przykład

Rozwiązujemy układ równań:

$$\begin{aligned} 3x &+ y &+ z &= 1 \\ x &+ 3y &+ z &= 1 \\ x &+ y &+ 3z &= 1 \end{aligned}$$



SOR

Jeśli $\rho(\mathbf{M}^{-1}\mathbf{N})$ w metodzie Gaussa-Seidela jest bliskie jedności, zbieżność
metody jest bardzo wolna. Można próbować ją poprawić:

$$x_i^{(k+1)} = w\left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^{N} a_{ij}x_j^{(k)}\right)\Big/ a_{ii} + (1-w)x_i^{(k)}, \tag{19}$$

gdzie $w \in \mathbb{R}$ jest *parametrem relaksacji*. Metoda ta zwana jest *succesive over-relaxation*, SOR. W postaci macierzowej

$$\mathbf{M}_w\mathbf{x}^{(k+1)} = \mathbf{N}_w\mathbf{x}^{(k)} + w\mathbf{b} \tag{20}$$

$\mathbf{M}_w = \mathbf{D} + w\mathbf{L}$, $\mathbf{N}_w = (1-w)\mathbf{D} - w\mathbf{U}$. *Teoretycznie* należy dobrać takie w, aby zminimalizować $\rho(\mathbf{M}_w^{-1}\mathbf{N}_w)$.

Metoda gradientów sprzężonych — motywacja

Rozważmy funcję $f : \mathbb{R}^N \to \mathbb{R}$

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{A}\mathbf{x} - \mathbf{b}^T\mathbf{x} + c\,, \tag{21}$$

gdzie $\mathbf{x}, \mathbf{b} \in \mathbb{R}^N$, $c \in \mathbb{R}$, $\mathbf{A} = \mathbf{A}^T \in \mathbb{R}^{N\times N}$ jest *symetryczna i dodatnio określona*. Przy tych założeniach, funkcja (21) ma dokładnie jedno minimum, będące zarazem minimum globalnym. Szukanie minimów dodatnio określonych form kwadratowych jest (względnie) łatwe i z praktycznego punktu widzenia ważne. Minimum to leży w punkcie spełniającym

$$\nabla f = 0\,. \tag{22}$$

Obliczmy

$$\frac{\partial f}{\partial x_i} = \frac{1}{2}\frac{\partial}{\partial x_i}\sum_{j,k} A_{jk}x_j x_k - \frac{\partial}{\partial x_i}\sum_j b_j x_j + \underbrace{\frac{\partial c}{\partial x_i}}_{0}$$

$$= \frac{1}{2}\sum_{j,k} A_{jk}\left(\underbrace{\frac{\partial x_j}{\partial x_i}}_{\delta_{ij}} x_k + x_j \underbrace{\frac{\partial x_k}{\partial x_i}}_{\delta_{ik}}\right) - \sum_j b_j \underbrace{\frac{\partial x_j}{\partial x_i}}_{\delta_{ij}}$$

$$= \frac{1}{2}\sum_k A_{ik}x_k + \frac{1}{2}\sum_j A_{ji}x_j - b_i = \frac{1}{2}\sum_k A_{ik}x_k + \frac{1}{2}\sum_j A_{ij}x_j - b_i$$

$$= (\mathbf{Ax} - \mathbf{b})_i \ . \tag{23}$$

Widzimy zatem, że funkcja (21) osiąga minimum w punkcie, w którym zachodzi

$$\mathbf{A}\mathbf{x} - \mathbf{b} = 0 \Leftrightarrow \mathbf{A}\mathbf{x} = \mathbf{b}. \tag{24}$$

Rozwiązywanie układu równań liniowych (24) z macierzą symetryczną, dodatnio określoną jest równoważne poszukiwaniu minimum dodatnio określonej formy kwadratowej.

Przypuśćmy, że macierz $\mathbf{A}$ jest przy tym *rzadka* i duża (lub co najmniej średnio-duża). Wówczas metoda gradientów sprzężonych jest godną uwagi metodą rozwiązywania (24)

Metoda gradientów sprzężonych, *Conjugate Gradients*, CG

$\mathbf{A} \in \mathbb{R}^{N\times N}$ symetryczna, dodatnio określona, x_1 — początkowe przybliżenie rozwiązania równania (24), $0 < \varepsilon \ll 1$.

$$
\begin{aligned}
&\mathbf{r}_1 = \mathbf{b} - \mathbf{A}\mathbf{x}_1,\ \mathbf{p}_1 = \mathbf{r}_1 \\
&\textbf{while } \|\mathbf{r}_k\| > \varepsilon \\
&\qquad \alpha_k = \frac{\mathbf{r}_k^T \mathbf{r}_k}{\mathbf{p}_k^T \mathbf{A}\mathbf{p}_k} \\
&\qquad \mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{A}\mathbf{p}_k \\
&\qquad \beta_k = \frac{\mathbf{r}_{k+1}^T \mathbf{r}_{k+1}}{\mathbf{r}_k^T \mathbf{r}_k} \\
&\qquad \mathbf{p}_{k+1} = \mathbf{r}_{k+1} + \beta_k \mathbf{p}_k \\
&\qquad \mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k \\
&\textbf{end}
\end{aligned}
\tag{25}
$$

Wówczas zachodzą twierdzenia:

Twierdzenie 7. *Ciągi wektorów* $\{\mathbf{r}_k\}$, $\{\mathbf{p}_k\}$ *spełniają następujące zależności:*

$$\mathbf{r}_i^T \mathbf{r}_j = 0\,, \quad i > j\,, \tag{26a}$$

$$\mathbf{r}_i^T \mathbf{p}_j = 0\,, \quad i > j\,, \tag{26b}$$

$$\mathbf{p}_i^T \mathbf{A} \mathbf{p}_j = 0\,, \quad i > j\,. \tag{26c}$$

Twierdzenie 8. *Jeżeli* $\mathbf{r}_M = 0$, *to* $\mathbf{x}_M$ *jest ścisłym rozwiązaniem równania* (24).

Dowód. Oba (sic!) dowody przebiegają indukcyjnie. □

Ciąg $\{\mathbf{x}_k\}$ jest w gruncie rzeczy "pomocniczy", nie bierze udziału w iteracjach, służy tylko do konstruowania kolejnych przybliżeń rozwiązania.

Istotą algorytmu jest konstruowanie dwu ciągów wektorów spełniających zależności (26). Wektory $\{\mathbf{r}_k\}$ są wzajemnie prostopadłe, a zatem *w arytmetyce dokładnej* $\mathbf{r}_{N+1} = 0$, wobec czego $\mathbf{x}_{N+1}$ jest poszukiwanym ścisłym rozwiązaniem.

Zauważmy, że ponieważ $\mathbf{A}$ jest symetryczna, dodatnio określona, warunek (26c) oznacza, że wektory $\{\mathbf{p}_k\}$ są wzajemnie prostopadłe w metryce zadanej przez $\mathbf{A}$. Ten właśnie warunek nazywa się warunkiem *sprzężenia względem* $\mathbf{A}$, co daje nazwę całej metodzie.

Koszt metody

W arytmetyce dokładnej metoda zbiega się po N krokach, zatem jej koszt wynosi $O(N \cdot \text{koszt\_jednego\_kroku})$. Koszt jednego kroku zdominowany jest przez obliczanie iloczynu $\mathbf{A}\mathbf{p}_k$. Jeśli macierz $\mathbf{A}$ jest pełna, jest to $O(N^2)$, a zatem całkowity koszt wynosi $O(N^3)$, czyli tyle, ile dla metod dokładnych. Jeżeli jednak $\mathbf{A}$ jest rzadka, koszt obliczania iloczynu jest mniejszy (o ile obliczenie to jest odpowiednio zaprogramowane). Jeśli $\mathbf{A}$ jest pasmowa o szerokości pasma $M \ll N$, całkowity koszt wynosi $O(M \cdot N^2)$.

Problem!

W arytmetyce o skończonej dokładności kolejne generowane wektory nie są *ściśle* ortogonalne do swoich poprzedników — na skutek akumulującego się błędu zaokrąglenia rzut na poprzednie wektory może stać się z czasem znaczny. Powoduje to istotne spowolnienie metody.

Twierdzenie 9. *Jeżeli* $\mathbf{x}$ *jest ścisłym rozwiązaniem równania* (24)*,* $\mathbf{x}_k$ *są generowane w metodzie gradientów sprzężonych, zachodzi*

$$\|\mathbf{x}-\mathbf{x}_k\| \leqslant 2\|\mathbf{x}-\mathbf{x}_1\| \left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^{k-1}, \tag{27}$$

gdzie κ *jest współczynnikiem uwarunkowania macierzy* $\mathbf{A}$*.*

Jeżeli $\kappa \gg 1$*, zbieżność może być bardzo wolna.*

“Prewarunkowana” (*preconditioned*) metoda gradientów sprzężonych

Spróbujmy przyspieszyć zbieżność odpowiednio modyfikując równanie (24) i algorytm (25), jednak tak, aby

- nie zmienić rozwiązania,
- macierz zmodyfikowanego układu pozostała symetryczna i dodatnio określona, aby można było zastosować metodę gradientów sprzężonych,
- macierz zmodyfikowanego układu pozostała rzadka, aby jeden krok iteracji był numerycznie tani,
- macierz zmodyfikowanego układu miała niski współczynnik uwarunkowania.

Czy to się w ogóle da zrobić? Okazuje się, że tak!

Postępujemy następująco: Niech $\mathbf{C} \in \mathbb{R}^{N\times N}$ będzie odwracalną macierzą symetryczną, rzeczywistą, dodatnio określoną. Wówczas $\widetilde{\mathbf{A}} = \mathbf{C}^{-1}\mathbf{A}\mathbf{C}^{-1}$ też jest symetryczna, rzeczywista, dodatnio określona.

$$\mathbf{C}^{-1}\mathbf{A}\underbrace{\mathbf{C}^{-1}\mathbf{C}}_{\mathbb{I}}\mathbf{x} = \mathbf{C}^{-1}\mathbf{b}\,, \tag{28a}$$

$$\widetilde{\mathbf{A}}\widetilde{\mathbf{x}} = \widetilde{\mathbf{b}}\,, \tag{28b}$$

gdzie $\widetilde{\mathbf{x}} = \mathbf{C}\mathbf{x}$, $\widetilde{\mathbf{b}} = \mathbf{C}^{-1}\mathbf{b}$. Do równania (28b) stosujemy teraz metodę gradientów sprzężonych.

W każdym kroku iteracji musimy obliczyć (tyldy, bo odnosi się to do “tyldowanego” układu (28b))

$$\alpha_k = \frac{\widetilde{\mathbf{r}}_k^T \widetilde{\mathbf{r}}_k}{\widetilde{\mathbf{p}}_k^T \widetilde{\mathbf{A}} \widetilde{\mathbf{p}}_k} = \frac{\widetilde{\mathbf{r}}_k^T \widetilde{\mathbf{r}}_k}{\widetilde{\mathbf{p}}_k^T \mathbf{C}^{-1}\mathbf{A}\mathbf{C}^{-1} \widetilde{\mathbf{p}}_k}\,, \tag{29a}$$

$$\widetilde{\mathbf{r}}_{k+1} = \widetilde{\mathbf{r}}_k - \alpha_k \widetilde{\mathbf{A}} \widetilde{\mathbf{p}}_k = \widetilde{\mathbf{r}}_k - \alpha_k \mathbf{C}^{-1}\mathbf{A}\mathbf{C}^{-1} \widetilde{\mathbf{p}}_k\,, \tag{29b}$$

$$\beta_k = \frac{\widetilde{\mathbf{r}}_{k+1}^T \widetilde{\mathbf{r}}_{k+1}}{\widetilde{\mathbf{r}}_k^T \widetilde{\mathbf{r}}_k}\,, \tag{29c}$$

$$\widetilde{\mathbf{p}}_{k+1} = \widetilde{\mathbf{r}}_{k+1} + \beta_k \widetilde{\mathbf{p}}_k\,, \tag{29d}$$

$$\widetilde{\mathbf{x}}_{k+1} = \widetilde{\mathbf{x}}_k + \alpha_k \widetilde{\mathbf{p}}_k\,. \tag{29e}$$

Równania (29) zawierają jawne odniesienia do macierzy $\mathbf{C}^{-1}$, co nie jest zbyt wygodne. Łatwo się przekonać, iż za pomocą prostych przekształceń macierz tę można „usunąć”, tak, iż pozostaje tylko jedno jej nietrywialne wystąpienie. Zdefiniujmy mianowicie

$$\widetilde{\mathbf{r}}_k = \mathbf{C}^{-1}\mathbf{r}_k\,, \quad \widetilde{\mathbf{p}}_k = \mathbf{C}\mathbf{p}_k\,, \quad \widetilde{\mathbf{x}}_k = \mathbf{C}\mathbf{x}_k\,. \tag{30}$$

W tej sytuacji $\widetilde{\mathbf{r}}_k^T\widetilde{\mathbf{r}}_k = (\mathbf{C}^{-1}\mathbf{r}_k)^T\mathbf{C}^{-1}\mathbf{r}_k = \mathbf{r}_k^T(\mathbf{C}^{-1})^T\mathbf{C}^{-1}\mathbf{r}_k = \mathbf{r}_k^T\mathbf{C}^{-1}\mathbf{C}^{-1}\mathbf{r}_k = \mathbf{r}_k^T(\mathbf{C}^{-1})^2\mathbf{r}_k$ etc.

Wówczas równania (29) przechodzą w

$$\alpha_k = \frac{\mathbf{r}_k^T \left(\mathbf{C}^{-1}\right)^2 \mathbf{r}_k}{\mathbf{p}_k^T \mathbf{A} \mathbf{p}_k}\,, \tag{31a}$$

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{A}\mathbf{p}_k\,, \tag{31b}$$

$$\beta_k = \frac{\mathbf{r}_{k+1}^T \left(\mathbf{C}^{-1}\right)^2 \mathbf{r}_{k+1}}{\mathbf{r}_k^T \left(\mathbf{C}^{-1}\right)^2 \mathbf{r}_k}\,, \tag{31c}$$

$$\mathbf{p}_{k+1} = \left(\mathbf{C}^{-1}\right)^2 \mathbf{r}_{k+1} + \beta_k \mathbf{p}_k\,, \tag{31d}$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k\,. \tag{31e}$$

W powyższych równaniach rola macierzy C sprowadza się do obliczenia — *jeden raz w każdym kroku iteracji* — wyrażenia $\left(\mathrm{C}^{-1}\right)^2 \mathbf{r}_k$, co, jak wiadomo, robi się rozwiązując odpowiedni układ równań. Zdefiniujmy

$$\mathbf{M} = \mathrm{C}^2 \,. \tag{32}$$

Macierz $\mathbf{M}$ należy rzecz jasna dobrać tak, aby równanie $\mathbf{Mz} = \mathbf{r}$ można było szybko rozwiązać.

Ostatecznie otrzymujemy następujący algorytm:

$$
\begin{aligned}
&\mathbf{r}_1 = \mathbf{b} - \mathbf{A}\mathbf{x}_1 \\
&\text{rozwiąż } \mathbf{M}\mathbf{z}_1 = \mathbf{r}_1 \\
&\mathbf{p}_1 = \mathbf{z}_1 \\
&\textbf{while } \|\mathbf{r}_k\| > \varepsilon \\
&\qquad \alpha_k = \frac{\mathbf{r}_k^T \mathbf{z}_k}{\mathbf{p}_k^T \mathbf{A}\mathbf{p}_k} \\
&\qquad \mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{A}\mathbf{p}_k \\
&\qquad \text{rozwiąż } \mathbf{M}\mathbf{z}_{k+1} = \mathbf{r}_{k+1} \\
&\qquad \beta_k = \frac{\mathbf{r}_{k+1}^T \mathbf{z}_{k+1}}{\mathbf{r}_k^T \mathbf{z}_k} \\
&\qquad \mathbf{p}_{k+1} = \mathbf{z}_{k+1} + \beta_k \mathbf{p}_k \\
&\qquad \mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k \\
&\textbf{end}
\end{aligned}
\tag{33}
$$

Incomplete Cholesky preconditioner

Niech rozkład QR macierzy $\mathbf{C}$ ma postać $\mathbf{C} = \mathbf{Q}\mathbf{H}^T$, gdzie $\mathbf{Q}$ jest macierzą ortogonalną, $\mathbf{H}^T$ jest macierzą trójkątną górną. Zauważmy, że

$$\mathbf{M} = \mathbf{C}^2 = \mathbf{C}^T\mathbf{C} = \left(\mathbf{Q}\mathbf{H}^T\right)^T \mathbf{Q}\mathbf{H}^T = \mathbf{H}\mathbf{Q}^T\mathbf{Q}\mathbf{H}^T = \mathbf{H}\mathbf{H}^T, \qquad (34)$$

a więc macierz $\mathbf{H}$ jest czynnikiem Cholesky'ego macierzy $\mathbf{M}$. Niech rozkład Cholesky'ego macierzy $\mathbf{A}$ ma postać $\mathbf{A} = \mathbf{G}\mathbf{G}^T$. *Przypuśćmy, iż* $\mathbf{H} \simeq \mathbf{G}$.

Wówczas

$$\widetilde{\mathbf{A}} = \mathbf{C}^{-1}\mathbf{A}\mathbf{C}^{-1} = \left(\mathbf{C}^{T}\right)^{-1}\mathbf{A}\mathbf{C}^{-1} = \left(\left(\mathbf{Q}\mathbf{H}^{T}\right)^{T}\right)^{-1}\mathbf{A}\left(\mathbf{Q}\mathbf{H}^{T}\right)^{-1} = \left(\mathbf{H}\mathbf{Q}^{T}\right)^{-1}\mathbf{A}\left(\mathbf{H}^{T}\right)^{-1}\mathbf{Q}^{T} = \mathbf{Q}\underbrace{\mathbf{H}^{-1}\mathbf{G}}_{\simeq\mathbb{I}}\underbrace{\mathbf{G}^{T}\left(\mathbf{H}^{T}\right)^{-1}}_{\simeq\mathbb{I}}\mathbf{Q}^{T} \simeq \mathbf{Q}\mathbf{Q}^{T} = \mathbb{I}\,. \tag{35}$$

Ponieważ $\widetilde{\mathbf{A}} \simeq \mathbb{I}$, współczynnik uwarunkowania tej macierzy powinien być bliski jedności.

Niepełny rozkład Cholesky'ego — algorytm w wersji *GAXPY*

for $k = 1{:}N$
  $H_{kk} = A_{kk}$
  for $j = 1{:}k{-}1$
    $H_{kk} = H_{kk} - H_{kj}^2$
  end
  $H_{kk} = \sqrt{H_{kk}}$
  for $l = k{+}1{:}N$
    $H_{lk} = A_{lk}$
    if $A_{lk} \neq 0$
      for $j = 1{:}k{-}1$
        $H_{lk} = H_{lk} - H_{lj} H_{kj}$
      end
      $H_{lk} = H_{lk}/H_{kk}$
    endif
  end
end

Uwagi

- Ponieważ $\mathbf{A}$ jest rzadka, powyższy algorytm na obliczanie przybliżonego czynnika Cholesky'ego wykonuje się szybko. Wykonuje się go *tylko raz*.
- Równanie $\mathbf{Mz} = \mathbf{r}$ rozwiązuje się szybko, gdyż znamy czynnik Cholesky'ego $\mathbf{M} = \mathbf{H}\mathbf{H}^T$.
- Obliczone $\mathbf{H}$ jest rzadkie, a zatem równanie $\mathbf{Mz} = \mathbf{r}$ rozwiązuje się szczególnie szybko.
- Mamy nadzieję, że macierz $\tilde{\mathbf{A}}$ ma współczynnik uwarunkowania bliski jedności, a zatem nie potrzeba wielu iteracji (33).

Przykład — macierz pasmowa z pustymi diagonalami

Rozważmy macierz o następującej strukturze:

$$\mathbf{A} = \begin{bmatrix} a_1 & 0 & b_3 & 0 & c_5 & 0 & 0 & 0 & 0 & 0 & \dots \\ 0 & a_2 & 0 & b_4 & 0 & c_6 & 0 & 0 & 0 & 0 & \dots \\ b_3 & 0 & a_3 & 0 & b_5 & 0 & c_7 & 0 & 0 & 0 & \dots \\ 0 & b_4 & 0 & a_4 & 0 & b_6 & 0 & c_8 & 0 & 0 & \dots \\ c_5 & 0 & b_5 & 0 & a_5 & 0 & b_7 & 0 & c_9 & 0 & \dots \\ 0 & c_6 & 0 & b_6 & 0 & a_6 & 0 & b_8 & 0 & c_{10} & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \end{bmatrix} \tag{36}$$

Macierz ta jest symetryczna, zakładamy też, że jest dodatnio określona.

Niepełny czynnik Cholesky'ego macierzy (36) ma postać

$$\mathbf{H} = \begin{bmatrix} p_1 & & & & & & \\ 0 & p_2 & & & & & \\ q_3 & 0 & p_3 & & & & \\ 0 & q_4 & 0 & p_4 & & & \\ r_5 & 0 & q_5 & 0 & p_5 & & \\ 0 & r_6 & 0 & q_6 & 0 & p_6 & \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \end{bmatrix} \tag{37}$$

(W pełnym czynniku Cholesky'ego macierzy (36) zera leżące w (37) *pomiędzy* diagonalą "p" a diagonalną "r" znikłyby — w ogólności mogłyby tam znajdować się jakieś niezerowe liczby.)

Zgodnie z podanym algorytmem, elementy ciągów $\{p_k\}$, $\{q_k\}$, $\{r_k\}$ wyliczamy z następujących wzorów:

$$
\begin{array}{rcl@{\qquad}rcl}
p_1 &=& \sqrt{a_1}, & p_2 &=& \sqrt{a_2}, \\
q_3 &=& b_3/p_1, & q_4 &=& b_4/p_2, \\
r_5 &=& c_5/p_1, & r_6 &=& c_6/p_2, \\
p_3 &=& \sqrt{a_3 - q_3^2}, & p_4 &=& \sqrt{a_4 - q_4^2}, \\
q_5 &=& (b_5 - r_5 q_3)/p_3, & q_6 &=& (b_6 - r_6 q_4)/p_4, \\
r_7 &=& c_7/p_3, & r_8 &=& c_7/p_4, \\
p_5 &=& \sqrt{a_5 - q_5^2 - r_5^2}, & p_6 &=& \sqrt{a_6 - q_5^6 - r_6^2}, \\
q_7 &=& (b_7 - r_7 q_5)/p_5, & q_8 &=& (b_8 - r_8 q_6)/p_6, \\
r_9 &=& c_9/p_5, & r_{10} &=& c_{10}/p_6, \\
p_7 &=& \sqrt{a_7 - q_7^2 - r_7^2}, & p_8 &=& \sqrt{a_8 - q_8^6 - r_8^2}, \\
q_9 &=& (b_9 - r_9 q_7)/p_7, & q_{10} &=& (b_{10} - r_{10} q_8)/p_8, \\
r_{11} &=& c_{11}/p_7, & r_{12} &=& c_{12}/p_8, \\
\ldots & \ldots & \ldots & \ldots & \ldots & \ldots
\end{array}
$$

Macierze niesymetryczne

Jeżeli w równaniu

$$\mathbf{A}\mathbf{x} = \mathbf{b} \tag{38}$$

macierz $\mathbf{A}$ nie jest symetryczna i dodatnio określona, sytuacja się komplikuje. Zakładając, że $\det \mathbf{A} \neq 0$, równanie (38) możemy "zsymetryzować" na dwa sposoby.
CGNR:

$$\mathbf{A}^T\mathbf{A}\mathbf{x} = \mathbf{A}^T\mathbf{b}\,, \tag{39}$$

lub CGNE:

$$\begin{aligned} \mathbf{A}\mathbf{A}^T\mathbf{y} &= \mathbf{b}\,, & \text{(40a)} \\ \mathbf{x} &= \mathbf{A}^T\mathbf{y}\,. & \text{(40b)} \end{aligned}$$

Do dwu powyższych równań formalnie rzecz biorąc *można* używać metody gradientów sprzężonych. Trzeba jednak pamiętać, że nawet jeśli macierz $\mathbf{A}$ jest rzadka, macierze $\mathbf{A}^T\mathbf{A}$, $\mathbf{A}\mathbf{A}^T$ nie muszą być rzadkie, a co gorsza, ich współczynnik uwarunkowania jest kwadratem współczynnika uwarunkowania macierzy wyjściowej.

Alternatywnie, zamiast "symetryzować" macierz, można zmodyfikować algorytm, tak aby zamiast dwu, generował on *cztery* ciągi wektorów. Należy jednak pamiętać, że dla wielu typów macierzy taki algorytm bywa bardzo wolno zbieżny, a niekiedy nawet dochodzi do kompletnej stagnacji przed uzyskaniem rozwiązania:

Metoda gradientów bi-sprzężonych (*Bi-Conjugate Gradients*, Bi-CG)

$\mathbf{r}_1 = \mathbf{b} - \mathbf{A}\mathbf{x}_1$, $\mathbf{p}_1 = \mathbf{r}_1$, $\bar{\mathbf{r}}_1 \neq 0$ dowolny, $\bar{\mathbf{p}}_1 = \bar{\mathbf{r}}_1$

while $\|\mathbf{r}_k\| > \varepsilon$

$$
\begin{aligned}
\alpha_k &= \frac{\bar{\mathbf{r}}_k^T \mathbf{r}_k}{\bar{\mathbf{p}}_k^T \mathbf{A}\mathbf{p}_k} \\
\mathbf{r}_{k+1} &= \mathbf{r}_k - \alpha_k \mathbf{A}\mathbf{p}_k \\
\bar{\mathbf{r}}_{k+1} &= \bar{\mathbf{r}}_k - \alpha_k \mathbf{A}^T \bar{\mathbf{p}}_k \\
\beta_k &= \frac{\bar{\mathbf{r}}_{k+1}^T \mathbf{r}_{k+1}}{\bar{\mathbf{r}}_k^T \mathbf{r}_k} \\
\mathbf{p}_{k+1} &= \mathbf{r}_{k+1} + \beta_k \mathbf{p}_k \\
\bar{\mathbf{p}}_{k+1} &= \bar{\mathbf{r}}_{k+1} + \beta_k \bar{\mathbf{p}}_k \\
\mathbf{x}_{k+1} &= \mathbf{x}_k + \alpha_k \mathbf{p}_k
\end{aligned}
\tag{41}
$$

end

Wektory wygenerowane w algorytmie (41) spełniają następujące relacje:

$$\bar{\mathbf{r}}_i^T \mathbf{r}_j = \mathbf{r}_i^T \bar{\mathbf{r}}_j = 0\,,\ i > j\,, \tag{42a}$$
$$\bar{\mathbf{r}}_i^T \mathbf{p}_j = \mathbf{r}_i^T \bar{\mathbf{p}}_j = 0\,,\ i > j\,, \tag{42b}$$
$$\bar{\mathbf{p}}_i^T \mathbf{A}\mathbf{p}_j = \mathbf{p}_i^T \mathbf{A}^T \bar{\mathbf{p}}_j = 0\,,\ i > j\,. \tag{42c}$$

Jeżeli w algorytmie (41) weźmiemy $\bar{\mathbf{r}}_1 = \mathbf{A}\mathbf{r}_1$, we wszystkich krokach zachodzić będzie $\bar{\mathbf{r}}_k = \mathbf{A}\mathbf{r}_k$ oraz $\bar{\mathbf{p}}_k = \mathbf{A}\mathbf{p}_k$. Jest to wersja przydatna dla rozwiązywania układów równań z macierzami symetrycznymi, ale nieokreślonymi dodatnio. Jest to przy okazji szczególny wariant algorytmu GMRES (*generalised minimum residual*), formalnie odpowiadającego minimalizacj funkcjonału

$$\Phi(\mathbf{x}) = \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2\,. \tag{43}$$