Sprawozdanie

P3.20 z analizy numerycznej
Obliczanie A^{-1} za pomocą rozkładu QR macierzy

Artur Derechowski
Mateusz Markiewicz

2 stycznia 2019

Spis treści

Wstęp

Macierz odwrotną do A, czyli taką A^{-1}, że $AA^{-1} = A^{-1}A = I$ można otrzymać na różne sposoby. W naszej pracowni przedstawiamy dwa z nich, czyli rozkład QR i rozkład LU oraz porównanie tych dwóch metod.

Cel zadania

Zadanie zostało podzielone na dwie części. Wyznaczamy rozkład macierzy $A = QR$, gdzie Q jest macierzą ortogonalną, a R macierzą górnotrójkątną. Następnie wyznaczamy również rozkład macierzy $A = LU$, gdzie L i U są odpowiednio macierzami dolno i górnotrójkątnymi.
Następnie wykorzystujemy oba te rozkłady do policzenia macierzy odwrotnej do A i badamy je pod względem dokładności.

Streszczenie sprawozdania

Opis teoretyczny problemu

Wprowadzenie do zagadnienia

?

Rozkład QR

Wstęp

Daną macierz A można jednoznacznie rozłożyć na iloczyn macierzy Q i R, takich, że $A = QR$. Dodatkowo, macierz Q jest ortogonalna, czyli jest odwrotna do swojej transpozycji, spełnia równanie $QQ^T = I$. Macierz R jest natomiast górnotrójkątna.
Dzięki temu można łatwo rozwiązać układ równań $Ax = b$:

$$\begin{aligned} Ax &= b \\ QRx &= b \\ Rx &= Q^T b \end{aligned}$$

Gdy układ równań jest trójkątny, kolejne zmienne można wyznaczyć podstawieniami w sumarycznym czasie $O(n^2)$.

Podobnie, mając macierz $A = QR$, można wyliczyć odwrotność macierzy A, co jest naszym zadaniem.

$$\begin{aligned} AA^{-1} &= I \\ QRA^{-1} &= I \\ A^{-1} &= R^{-1}Q^T \end{aligned}$$

Przekształcenia Householdera

Aby uzyskać macierz górnotrójkątną R, w każdym kroku algorytmu będziemy zerować dolną część jednej kolumny macierzy A. Używamy do tego przekształceń Householdera, które jest odbiciem, czyli przekształceniem ortogonalnym. Przykładowo, po zastosowaniu jednego przekształcenia Householdera, macierz H_1A będzie wyglądała następująco:

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ 0 & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & a_{n2} & \dots & a_{nn} \end{bmatrix}$$

Aby uzyskać takie przekształcenie mnożymy macierz A przez macierz Householdera postaci

$$H = I - 2v * v^T$$

gdzie v jest wektorem jednostkowym.

Przebieg algorytmu

Do rozkładu QR stosuje się przekształcenia Householdera na wektorze

$$u = x - \|x\|e_1$$

gdzie x jest pierwszą kolumną macierzy A, a $e_1 = [1, 0, ..., 0]^T$.

Gdy normalizujemy wektor u, macierz przekształcenia H dana jest wzorem:

$$H = I - 2\frac{u}{\|u\|}\frac{u^T}{\|u^T\|} = I - 2\frac{uu^T}{u^Tu}$$

Można pokazać, że po tym przekształceniu macierz $H_1 A$ będzie miała wyzerowaną pierwszą dolną kolumnę poza najwyższym polem, czyli będzie odpowiedniej postaci do dalszego przekształcania na macierz górnotrójkątną.

W kolejnych krokach algorytmu stosujemy przekształcenia Householdera na macierzy A bez lewej kolumny i górnego wiersza. Wtedy uzyskana macierz będzie jednak mniejsza od macierzy A, więc aby nie zmieniać "lewych" kolumn, które zostały wcześniej dobrze dopasowane, dopełniamy macierz H_i macierzą identycznościową. Wtedy macierz, przez którą w każdym kroku mnożymy dotychczas uzyskaną macierz wygląda następująco:

$$H_k := \begin{bmatrix} I_{n-k} & 0 \\ 0 & H_k \end{bmatrix}$$

Stosując $n-1$ kolejnych przekształceń Householdera H_i otrzymujemy wynikowo macierz górnotrójkątną R, a także ortogonalną macierz Q, czyli rozkład, którego szukamy w następujący sposób:

$$R = H_{n-1} ... H_2 H_1 A$$
$$Q = H_1 H_2 ... H_{n-1}$$

Wynikowe macierze w rozkładzie QR można więc przedstawić jako iloczyn macierzy Householdera.

Można zobaczyć, że mając rozkład macierzy na iloczyn QR spełniający wyżej wymienione własności, macierz odwrotną A^{-1} można wyliczyć jako $A^{-1} = R^{-1}Q^T$. Następnym krokiem algorytmu jest więc odwrócenie macierzy górnotrójkątnej R. Można to zrobić w następujący sposób:

```
for i := 1 to n do
    A_ii = 1/A_ii
    for i := n − 1 step −1 to 1 do
        for j := n step −1 to i + 1 do
            s := 0
            for k := i + 1 to j do
                s := s + A_ik * A_kj
                A_ij = −A_ii * s
```

Ten algorytm, zastosowany do macierzy górnotrójkątnej, wylicza jej odwrotność w czasie $O(n^3)$.

Cały algorytm wyznaczania macierzy odwrotnej za pomocą rozkładu QR tworzy n macierzy H_i, które można wyliczyć w czasie $O(n^2)$, następnie wykonuje operacje w czasie $O(n^3)$, więc cała złożoność algorytmu to $O(n^3)$.

Gram-Schmidt

Znaną metodą ortogonalizacji macierzy jest proces Grama-Schmidta i również w ten sposób można uzyskać rozkład QR. Nie jest to jednak zalecane, ponieważ w wyniku tych przekształceń mogą powstawać bardziej znaczące błędy numeryczne. Można to wywnioskować interpretując proces Grama-Schmidta jako odejmowanie od wektora jego rzutów na poprzednio uzyskane wektory. Wtedy, gdy dwa wektory były "prawie" liniowo zależne, odejmujemy niemal cały wektor, co powoduje brak stabilności numerycznej.

Rozkład LU

Wstęp

Rozkład LU macierzy A polega na znalezieniu macierzy dolnotrójkątnej L oraz górnotrójkątnej U, takich że ich iloczyn będzie macierzą A, czyli:

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \dots & 0 \\ l_{21} & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ l_{n1} & l_{n2} & \dots & 1 \end{bmatrix} \times \begin{bmatrix} u_{11} & u_{12} & \dots & u_{1n} \\ 0 & u_{22} & \dots & u_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & u_{nn} \end{bmatrix}$$

Metoda Doolittle'a

Metoda Doolittle'a polega na naprzemiennym wyznaczaniu kolejnych wierszy macierzy U oraz kolumn macierzy L. Szczegółowy algorytm wygląda następująco:

for $i := 1$ to n do
  $u_{ii} = a_{ii} - \sum_{k=1}^{i-1} l_{ik}u_{ki}$
  $l_{ii} = 1$
  for $j := i + 1$ to n do
    $u_{ij} = a_{ij} - \sum_{k=1}^{i-1} l_{ik}u_{kj}$
    $l_{ji} = \frac{1}{u_{ii}}(a_{ji} - \sum_{k=1}^{i-1} l_{jk}u_{ki})$

Na podstawie algorytmu widać, że rozkład macierzy rozmiaru n wymaga $O(n^3)$ operacji. Duża liczba operacji wpływa negatywnie zarówno na czas działania, jak również dokładność obliczeń.

Obliczenie macierzy odwrotnej za pomocą rozkładu LU

Dla danej macierzy kwadratowej A rozmiaru n macierz odwrotna A^{-1} to macierz kwadratowa tego samego rozmiaru spełniająca równość

$$A \times A^{-1} = Id = A^{-1} \times A$$

Powyższą równość możemy zapisać w postaci:

$$A \times \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{nn} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix}$$

stąd otrzymujemy n układów równań:

$$A \times \begin{bmatrix} x_{11} \\ x_{21} \\ \vdots \\ x_{n1} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad A \times \begin{bmatrix} x_{12} \\ x_{22} \\ \vdots \\ x_{n2} \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix} \quad \dots \quad A \times \begin{bmatrix} x_{1n} \\ x_{2n} \\ \vdots \\ x_{nn} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}$$

Rozwiązywanie układów równań za pomocą rozkładu LU

Korzystając z własności macierzy wiemy, że:

$$\begin{gathered} A \times X = B \\ (L \times U) \times X = B \\ L \times (U \times X) = B \end{gathered}$$

Stąd znajdując wektor Y taki, że $L \times Y = B$ możemy wyznaczyć wektor X z własności $U \times X = Y$. Wektory X oraz Y łatwo wyznaczyć z zależności:

$$\begin{gathered} y_1 = b_1 \\ y_i = b_i - \textstyle\sum_{j=1}^{i-1} l_{ij} y_j, \text{ dla } i = 2, 3, ..., n \\ x_n = \frac{y_n}{u_{nn}} \\ x_i = \frac{1}{u_{ii}} \Big(y_i - \textstyle\sum_{j=i+1}^{n} u_{ij} x_j \Big) \text{ dla } i = n-1, n-2, ..., 1 \end{gathered}$$

Wyznaczanie A^{-1}

Rozwiązując powyższe n układów równań otrzymamy n wektorów $X_1, X_2, ..., X_n$. Niech $X = [X_1|X_2|...|X_n]$, wtedy $X = A^{-1}$, czyli wyznaczyliśmy macierz odwrotną do macierzy A. Dla macierzy A rozmiaru $n \times n$ rozkład na macierze

L, U wymaga $O(n^3)$ operacji. Obliczenie A^{-1} wymaga rozwiązania n układów równań, a rozwiązanie każdego z tych układów wymaga $O(n^2)$ operacji, stąd cały algorytm obliczania A^{-1} za pomocą rozkładu LU jest rzędu $O(n^3)$.

?

Podsumowanie

Literatura

1) D. Kincaid, W. Cheney, Analiza numeryczna, WNT, 2005.
2) A. Schegel (aaronsc32), QR Decomposition with Householder Reflections, RPubs, 2018.
3) Mathematics Source Library C&ASM, mymathlib.com, 2004.
4) D. Bindel, Matrix Computations (CS6210), Cornell University, Sep 28 2012.