Algorytmy dekompozycji QR

Anna Szczepaniak, 210094

Praca licencjacka przygotowana pod opieką dr, mgr inż. Piotra Kowalskiego

19.09.2019

Twierdzenie (Nierówność Bessela (?, Twierdzenie 14.22))

Jeśli $(\mathbf{b}_1, \ldots, \mathbf{b}_n)$ *jest układem ortonormalnym w przestrzeni euklidesowej* $\mathbb{R}^k$*, to dla każdego wektora* $\mathbf{x}$ *z przestrzeni* $\mathbb{R}^k$ *spełniona jest nierówność*

$$\sum_{i=1}^{n} \alpha_i^2 \leqslant \|\mathbf{x}\|^2,$$

gdzie $\alpha_i = (\mathbf{x}|\mathbf{b}_i)$*. Ponadto, wektor* $\mathbf{x} - \sum_{i=1}^{n} \alpha_i \cdot \mathbf{b}_i$ *jest ortogonalny do podprzestrzeni span*$(\mathbf{b}_1, \ldots, \mathbf{b}_n)$*.*

Twierdzenie (Twierdzenie(Grama-Schmidta)(**?**))

Dla każdego układu liniowo niezależnego wektorów $(\mathbf{x}_1, \dots, \mathbf{x}_n)$ *w przestrzeni euklidesowej istnieje układ ortonormalny* $(\mathbf{b}_1, \dots, \mathbf{b}_n)$ *taki, że*

$$span(\mathbf{b}_1, \dots, \mathbf{b}_k) = span(\mathbf{x}_1, \dots, \mathbf{x}_k)$$

dla każdej liczby k *ze zbioru* $(1, \dots, n)$*.*

Twierdzenie (O rozkładzie QR)

Niech $A \in \mathbb{R}^{m \times n}$, *gdzie* $m \geqslant n$, *której kolumny są liniowo niezależne. Istnieje wtedy jedyny rozkład QR, tzn. że istnieją takie macierze* Q *i* R, *że*

$$A = QR$$

i

- *macierz* $Q \in \mathbb{R}^{m \times n}$ *jest taka, że*

  $$Q^T \cdot Q = D,$$

  gdzie $D = diag(d_1, d_2, ..., d_n)$, *oraz* $d_k > 0$ *dla* $k = 1, 2, \ldots, n$, *oraz*

- *macierz* $R \in \mathbb{R}^{n \times n}$ *jest trójkątną górną spełniającą dodatkowo warunek*

  $$r_{kk} = 1$$

  dla wszystkich $k = 1, 2, \ldots, n$.

Uwaga

Rozkładem QR nazwiemy również sytuacje, gdy

$$A = QR$$

oraz $Q^T Q = I$ *i* R *jest macierzą trójkątną górną, niekoniecznie z* 1 *na przekątnej. Powodem tego jest, że prezentowana tu postać oraz postać z twierdzenia 3 są sobie równoważne.*

Definicja (Macierz Householdera)

Macierzą Householdera H, zwaną również refleksją, nazywamy macierz postaci

$$H = I - 2 \cdot \mathbf{v} \cdot \mathbf{v}^T,$$

gdzie $\|v\|_2 = 1$.

Twierdzenie (Transformacja Householdera)

Niech $\mathbf{v} \in R^m$, *i* $\mathbf{v} \neq 0$. *Wówczas transformacją Householdera nazywamy macierz postaci:*

$$H = I - W\mathbf{v}\mathbf{v}^T,$$

gdzie

$$W = \frac{2}{\mathbf{v}^T\mathbf{v}}.$$

Macierz H jest macierzą symetryczną i ortogonalną.

Definicja (Macierz Givensa)

Niech $i, j \in 1, \ldots, n$ *i* $\theta \in \mathbb{R}$. *Macierz* $R(i,j,\theta) \in \mathbb{R}^{n \times n}$ *zdefiniowana następująco:*

$$R(i,j,\theta) = \begin{bmatrix} 1 & 0 & \cdots & 0 & \cdots & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 & 0 \\ 0 & 0 & \ddots & 0 & \cdots & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \cdots & \cos\theta & \cdots & -\sin\theta & \cdots & 0 & 0 \\ 0 & 0 & \cdots & \vdots & \ddots & \vdots & \cdots & 0 & 0 \\ \vdots & \vdots & \cdots & \sin\theta & \cdots & \cos\theta & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & \cdots & 0 & \ddots & 0 & 0 \\ \vdots & \vdots & \cdots & \cdots & \cdots & \cdots & \cdots & 1 & 0 \\ 0 & 0 & \cdots & 0 & \cdots & 0 & \cdots & 0 & 1 \end{bmatrix}$$

nazywamy macierzą rotacji Givensa.

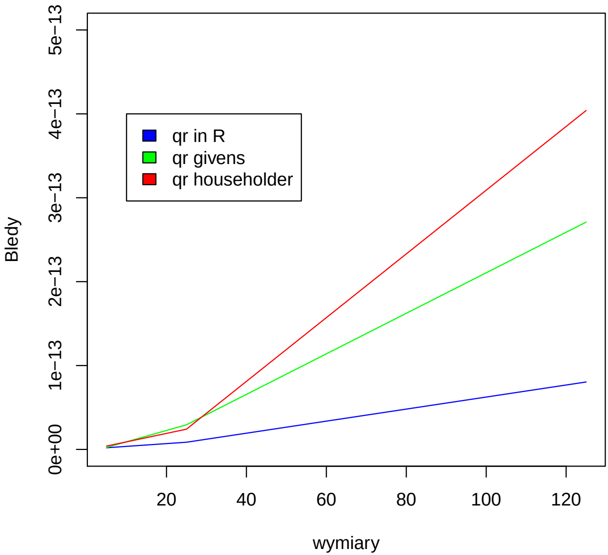
```
> house = function(x) {
+   norm_x = sqrt(t(x)\cdot x)
+   if (x[1] > 0) {
+     x[1] = x[1] + norm_x;
+   } else {
+     x[1] = x[1] - norm_x;
+   }
+   return(x)
+ }
```

```
> qr_householer = function(A)
+ {
+  R = A
+  dimm = dim(A)
+  n = dimm[2]
+   m = dimm[1]
+   Q = diag(n)
+   for (i in 1:min(m-1,n)){
+     u = house(R[i:m,i])
+     den =  (t(u) %*% u)[1,1]
+     P = diag(n+1-i) - 2/den * ( u%*% t(u)  )
+     R[i:m,i:n] = P %*% R[i:m,i:n]
+     if (i>1)
+       Q[i:n,1:(i-1)] = P %*% Q[i:n,1:(i-1)]
+     Q[i:n,i:n] = P %*% Q[i:n,i:n]
+   }
+   return(list(R=R, Q=t(Q)))
+ }
```

```
> givens = function(i,j,A){
+   dimm = dim(A)
+   n = dimm[1]
+   G = diag(n)
+   p = sqrt((A[i,i])^2 + (A[j,i])^2)
+   G[i,i] = A[i,i]/ p
+   G[i,j] = (A[j,i])/p
+   G[j,i] = -(A[j,i])/p
+   G[j,j] = A[i,i]/ p
+
+ return(G)
+
+ }
```

```
> qr_givens = function(A){
+   R = A
+
+   n = dim(A)[2]
+   m = dim(A)[1]
+   Q = diag(n)
+   for (i in 1:(n-1)) {
+     for (j in (i+1):m) {
+       G = givens(i,j,R)
+       R = G %*% R
+       Q = Q %*% t(G)
+     }
+   }
+   return(list(R=R, Q=Q))
+ }
```

Rozmiar macierzy nie wpływa istotnie na jakość naszych samodzielnie opisanych funkcji w stosunku do gotowej funkcji w R. Są one mniej dokładne niż wbudowany, gotowy algorytm dekompozycji QR w R. Ponadto algorytm metodą rotacji Givensa nie zawsze jest jakościowo lepszy od algorytmu metodą odbić Householdera. Zależy to od wymiarów oraz od losowości rozkładanej macierzy.


qr in R
qr givens
qr householder
Bledy
wymiary
0e+00
1e−13
2e−13
3e−13
4e−13
5e−13
20
40
60
80
100
120