Vilniaus Universitetas

2 laboratorinis

Skaitiniai metodai

Darbą atliko:

Dovydas Martinkus

Duomenų Mokslas 4 kursas 1 gr.

Vilnius, 2022

**Turinys**

[1 Užduoties ataskaita 3](#_Toc116072976)

[1.1 Zeidelio metodas 3](#_Toc116072977)

[1.2 Jungtinių gradientų metodas 4](#_Toc116072978)

[Priedas 6](#_Toc116072979)

# Užduoties ataskaita

Reikalinga 0,0001 tikslumu iteraciniais ir variaciniais metodais išspręsti tiesinę lygčių sistemą *Ax=B*, kur:

A = ,

B = .

Šiam tikslui buvo naudojami Zeidelio ir jungtinių gradientų metodai.

## Zeidelio metodas

Žemiau pateikti Zeidelio metodu gauti iteracijų rezultatai 5 skaičių po kablelio tikslumu:

1 lentelė Iteracijų rezultatai naudojant Zeidelio metodą

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *n* | *x1* | *x2* | *x3* | *x4* | *||xn-x||∞* |
| 0 | 0.00000 | 0.00000 | 0.00000 | 0.00000 | 7.70000 |
| 1 | 0.57692 | 0.58637 | 1.04419 | 0.35293 | 1.82900 |
| 2 | 0.15421 | 0.35996 | 1.15679 | 0.46462 | 0.18742 |
| 3 | 0.17313 | 0.31128 | 1.15977 | 0.47025 | 0.02983 |
| 4 | 0.18479 | 0.30734 | 1.15859 | 0.46896 | 0.00486 |
| 5 | 0.18634 | 0.30752 | 1.15838 | 0.46866 | 0.00044 |
| 6 | 0.18639 | 0.30761 | 1.15836 | 0.46864 | 0.00006 |

Konvergavimas pasiektas po 6 iteracijų. Patikrinimui gautą artinį įstatome į lygčių sistemą ir palyginame gautus rezultatus (6 skaičių po kablelio tikslumu):

2 lentelė Gauto artinio patikrinimas jį įstatant į lygčių sistemą

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | b1 | b2 | b3 | b4 |
| Gautas B | 1.200035 | 1.99998 | 2.999995 | 1.5 |
| Norimas B | 1.200000 | 2.00000 | 3.000000 | 1.5 |

Jei *||C|| < 1*, tai iteracinė seka *xk+1 = Cxk + d, k > 0,* konverguoja į *Ax = b* sprendinį kiekvienam pradiniam artiniui x0.

Zeidelio metodo atveju turime iteracinę seką *xk+1= D−1[(L + U)xk + b] = D−1[(D − A)xk + b],* vadinasi

*C = D−1(L + U).*

Apskaičiavę *||C||* reikšmę gauname *||C|| = 0.5527425 < 1,* todėl sudaryta iteracinė seka konverguoja su bet kokiu pradiniu artiniu.

## Jungtinių gradientų metodas

Prisimename, kad turime matricą:

A = .

Matome, kad matrica A yra simetrinė, be to ji ir teigiamai apibrėžta (nes ji simetrinė ir jos tikrinės reikšmės 3.85, 2.17, 1.79 ir 1.67 visos didesnės už 0).

Pagal teoremą, jeigu papildomai *p0, p1, ...., pn-1* sudaro matricos *A* atžvilgiu jungtinių vektorių sistemą, tai ne daugiau kaip per *n* iteracijų su bet kuriuo pradiniu artiniu *x0* iteraciniu metodu , gaunamas tikslus lygčių sistemos *Ax=B* sprendinys. Ši sąlyga patikrinta naudojant R ir yra tenkinama.

Žemiau pateikti jungtinių gradientų metodu gauti iteracijų rezultatai 5 skaičių po kablelio tikslumu:

3 lentelė Iteracijų rezultatai naudojant jungtinių gradientų metodą

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *n* | *x1* | *x2* | *x3* | *x4* | *||xn-x||∞* |
| 0 | 0.00000 | 0.00000 | 0.00000 | 0.00000 | 7.70000 |
| 1 | 0.33579 | 0.55965 | 0.83948 | 0.41974 | 1.37473 |
| 2 | 0.19674 | 0.30824 | 1.16337 | 0.44882 | 0.07021 |
| 3 | 0.18854 | 0.30627 | 1.15819 | 0.46903 | 0.00804 |
| 4 | 0.18637 | 0.30762 | 1.15837 | 0.46864 | 0.00000 |

Konvergavimas pasiektas po 4 iteracijų. Patikrinimui gautą artinį įstatome į lygčių sistemą ir palyginame gautus rezultatus (neapvalintus):

4 lentelė Gauto artinio patikrinimas jį įstatant į lygčių sistemą

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | b1 | b2 | b3 | b4 |
| Gautas B | 1.2 | 2 | 3 | 1.5 |
| Norimas B | 1.2 | 2 | 3 | 1.5 |

Matome, kad gautas tikslus lygčių sistemos *Ax=B* sprendinys.

Galiausiai lentelėje pateiktas Zeidelio, jungtinių gradientų ir naudojant R biblioteką *Rlinsolve* gautų artinių palyginimas. Kaip matome visais metodais gauti beveik identiški rezultatai:

5 lentelė Skirtingais būdais gautų artinių palyginimas

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Metodas | x1 | x2 | x3 | x4 |
| Zeidelio | 0.1863910 | 0.3076080 | 1.158364 | 0.4686360 |
| Jungtinių gradientų | 0.1863705 | 0.3076182 | 1.158366 | 0.4686376 |
| R iteracinis *lsolve.jacobi()* | 0.1863781 | 0.3075950 | 1.158385 | 0.4686426 |
| R variacinis *lsolve.cg()* | 0.1863705 | 0.3076182 | 1.158366 | 0.4686376 |

# Priedas

Žemiau pateiktas naudotas programinis kodas:

# Dovydas Martinkus

# Duomenų Mokslas 4k. 1gr.

# 2 uzduotis

###

zeidelio <- function(x0,D,L,U,B,eps) {

x <- matrix(x0)

n <- 0

repeat {

x\_n <- solve(D-L) %\*% (U %\*% matrix(x[,n+1])+B)

x <- cbind(x, x\_n)

if (norm(matrix(matrix(x[,n+2]) - matrix(x[,n+1]),

byrow=TRUE),

type = "M") > eps) {

n <- n + 1

}

else {

return(t(x))

}

}

}

jungtiniu\_gradientu <- function(x0,A,B,eps) {

x <- matrix(x0)

z <- A %\*% x - B

p <- z

dot <- t(z) %\*% z

r <- A %\*% p

tau <- c((t(z) %\*% z) / (t(r) %\*% p))

n <- 0

repeat {

if( n>0 ){

r\_n <- A %\*% matrix(p[,n+1])

r <- cbind(r, r\_n)

tau\_n <- (t(matrix(z[,n+1])) %\*% matrix(z[,n+1])) / (t(matrix(r[,n+1])) %\*% matrix(p[,n+1]))

tau <- cbind(tau, tau\_n)

}

x\_n <- matrix(x[,n+1]) - tau[n+1]\*matrix(p[,n+1])

z\_n <- matrix(z[,n+1]) - tau[n+1]\*matrix(r[,n+1])

x <- cbind(x, x\_n)

z <- cbind(z, z\_n)

if ( t(z\_n) %\*% z\_n >= eps^2 ) {

beta\_n <- c((t(z\_n) %\*% z\_n) / ( t(matrix(z[,n+1])) %\*% matrix(z[,n+1])))

p\_n <- z\_n + beta\_n\*matrix(p[,n+1])

p <- cbind(p, p\_n)

n <- n + 1

}

else {

return(list(a=t(x),b=t(p)))

}

}

}

lentele <- function(xn,A,B) {

norms <- apply(xn,1,function(x) norm(matrix(A %\*% x - B,byrow=TRUE)))

xn <- cbind(0:(nrow(xn)-1),xn,norms)

return(xn)

}

D <- matrix(c(0.18, 0.57, 0.38, 0.42,

0.57, 0.95, 0.70, 0.44,

0.38, 0.70, 0.37, 0.18,

0.42, 0.44, 0.18, 0.40),

ncol=4,nrow=4)

E <- diag(1, 4, 4)

B <- matrix(c(1.2,2,3,1.5))

A <- D + 0.1\*(16+3)\*E

U <- A

U[lower.tri(U,TRUE)] <- 0

U <- U \* -1

L <- A

L[upper.tri(L,TRUE)] <- 0

L <- L \* -1

D <- A

D[!(lower.tri(D,TRUE) & upper.tri(D,TRUE))] <- 0

D <- D

eps <- 0.0001

x0 <- rep(0,length(B))

# Zeidelio

xn\_1 <- zeidelio(x0,D,L,U,B,eps)

round(lentele(xn\_1,A,B),5)

palyginimas <- cbind(A %\*% matrix(xn\_1[nrow(xn\_1),]),B)

colnames(palyginimas) <- c("Gautas B","Norimas B")

round(t(palyginimas),6)

# Jungtiniu gradientu

result <- jungtiniu\_gradientu(x0,A,B,eps)

xn\_2 <- result[[1]]

p\_n <- result[[2]]

round(lentele(xn\_2,A,B),5)

palyginimas <- cbind(A %\*% matrix(xn\_2[nrow(xn\_2),]),B)

colnames(palyginimas) <- c("Gautas B","Norimas B")

t(palyginimas)

# patikriname ar p\_n sudaro matricos A atzvilgiu jungtiniu vektoriu sistema

for(i in 1:nrow(p\_n)) {

for(j in 1:nrow(p\_n)) {

if(i != j) {

print(t(A%\*%p\_n[i,]) %\*% p\_n[j,])

}

}

}

library(Rlinsolve)

# Artiniu palyginimas

palyginimas <- cbind(matrix(xn\_1[nrow(xn\_1),]),matrix(xn\_2[nrow(xn\_2),]),lsolve.jacobi(A,B)$x, lsolve.cg(A,B)$x)

colnames(palyginimas) <- c("Zeidelio","Jungtinių gradientų","R iteracinis metodas","R variacinis metodas")

t(palyginimas)