Métodos numéricos - Tarea 9
Método de iteraciones en subespacios
Factorización QR

Salim Vargas Hernández

15 de octubre del 2018

1. Introducción

El método de iteraciones en subespacios permite encontrar los primeros m eigenvalores y eigenvectores de mayor magnitud en valor absoluto de A, con pequeñas modificaciones se pueden encontrar los m de magnitud más pequeña o los m más cercanos a un valor dado para los eigenvalores de A.
Este método resulta bastante útil cuando no se quiere calcular todo el espectro de A o cuando la matriz A es demasiado grande.
La factorización QR de una matriz A es una descomposición de la misma como producto de una matriz ortonormal por una triangular superior.

2. Desarrollo

2.1. Método de iteraciones en subespacios

Dada una matriz A, se quieren calcular los m mayores eigenvalores y eivenvectores en valor absoluto. El algoritmo 1 muestra el pseudocódigo del método.

2.2. Factorización QR

Dada una matriz A, se buscan matrices Q y R tales que Q sea contenga una base ortonormal de A y R sea triangular superior.
Estas matrices se pueden encontrar mediante el método de Gram-Schmidt, el cual, dada una matriz con columnas linealmente independientes, obtiene una matriz con columnas ortogonales entre si y de norma 1.
Este método permite encontrar Q, dado que es ortonormal, su transpuesta es su inversa, por lo que la matriz R se puede entontrar con la operación

$$R = Q^T A$$

El algoritmo 2 muestra el pseudocódigo del método de ortonormalización de Gram-Schmidt.

Algoritmo 1 Método de iteraciones en subespacios para cálculo de valores y vectores propios

Entrada:
Matriz simétrica A
número de eigenvalores a calcular m
tolerancia *tol*
máximo de iteraciones *maxIter*

Salida: Matriz diagonal C con los m mayores valores propios en valor absoluto
Matriz ortonormal X con los m mayores vectores propios

Inicializar X como un conjunto de vectores ortonormales.
Mientras no haya convergencia y no se alcance el máximo de iteraciones **hacer**
  $C = X^T AX$
  Obener las matrices Q y Λ de la factorización $C = Q^T \Lambda Q$ utilizando el método de Jacobi
  Si C es una matriz diagonal **entonces**
    se alzanzó la convergencia
  Si no
    $X = XQ$
    Ortonormalizar X usando el método de Gram-Schmidt
    Para cada vector ϕ de X **hacer**
      Restar a ϕ las proyecciones de los eigenvectores calculados antes que él
      Realizar 1 iteración del método de la Potencia partiendo del vector ϕ
      Actualizar ϕ en X tras este paso
    Fin Para
    Ortonormalizar X usando el método de Gram-Schmidt
  Fin Si
Fin Mientras
Devolver C, X

Algoritmo 2 Método de ortonormalización de Gram-Schmidt

Entrada:
Matriz A

Salida: Matriz ortonormal U

Para cada vector v en A **hacer**
  $u = v$
  $j = 0$
  Mientras j menor al índice de v en A **hacer**
    $u = u - \frac{< v, U_j >}{||U_j||^2} U_j$
    $j = j + 1$
  Fin Mientras
  Agregar $\frac{u}{||u||}$ a U
Fin Para
Devolver U

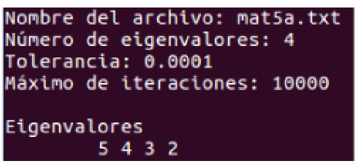
3. Resultados

3.1. Método de iteraciones en subespacios

Se probó el método de iteraciones en subespacios con las matrices `mat5a.txt`, `mat5b.txt` y `mat10.txt`. Para lograr la convergencia se utilizaron los siguientes parámetros:

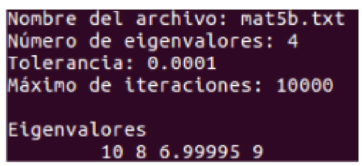
Parámetro	`mat5a.txt`	`mat5b.txt`	`mat10.txt`
tolerancia	0.0001	0.0001	0.001
maxIter	10,000	10,000	10,000
m	4	4	8

Para cada matriz se imprimían los eigenvalores obtenidos. Los resultados se muestran en la figura 1.



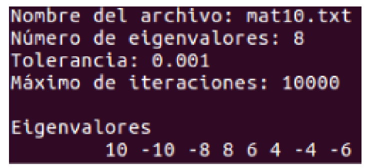
```
Nombre del archivo: mat5a.txt
Número de eigenvalores: 4
Tolerancia: 0.0001
Máximo de iteraciones: 10000

Eigenvalores
        5 4 3 2
```



```
Nombre del archivo: mat5b.txt
Número de eigenvalores: 4
Tolerancia: 0.0001
Máximo de iteraciones: 10000

Eigenvalores
        10 8 6.99995 9
```



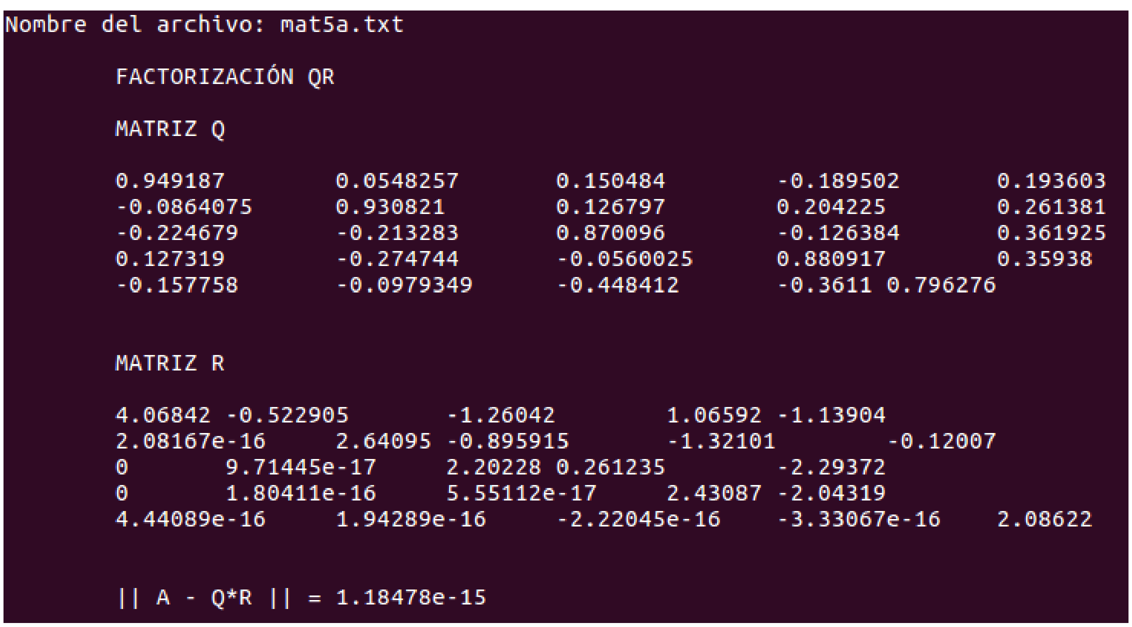
```
Nombre del archivo: mat10.txt
Número de eigenvalores: 8
Tolerancia: 0.001
Máximo de iteraciones: 10000

Eigenvalores
        10 -10 -8 8 6 4 -4 -6
```

Figura 1: Resultados del método iteraciones en el subespacio

3.2. Factorización QR

Se probó la factorización QR con las matrices `mat5a.txt`, `mat5b.txt` y `mat10.txt`.
Se imprimían en consola las matrices Q y R, así como la norma matricial de $||A-QR||$. Los resultados se muestran en las figuras 2 y 3.
En el caso de la matriz `mat10.txt`, el resultado fue $||A-QR|| = 6.46966e-15$.



```
Nombre del archivo: mat5a.txt

        FACTORIZACIÓN QR

        MATRIZ Q

        0.949187        0.0548257       0.150484        -0.189502       0.193603
        -0.0864075      0.930821        0.126797        0.204225        0.261381
        -0.224679       -0.213283       0.870096        -0.126384       0.361925
        0.127319        -0.274744       -0.0560025      0.880917        0.35938
        -0.157758       -0.0979349      -0.448412       -0.3611 0.796276


        MATRIZ R

        4.06842 -0.522905       -1.26042        1.06592 -1.13904
        2.08167e-16     2.64095 -0.895915       -1.32101        -0.12007
        0       9.71445e-17     2.20228 0.261235        -2.29372
        0       1.80411e-16     5.55112e-17     2.43087 -2.04319
        4.44089e-16     1.94289e-16     -2.22045e-16    -3.33067e-16    2.08622


        || A - Q*R || = 1.18478e-15
```

Figura 2: Factorización QR de la matriz `mat5a.txt`

```
Nombre del archivo: mat5b.txt

        FACTORIZACIÓN QR

        MATRIZ Q

        0.99392 0.0753524       0.0480536       -0.0457897      0.0451549
        -0.0708226      0.989939        0.00291097      0.119809        0.025333
        -0.0508109      -0.0169841      0.985296        0.0540112       0.152982
        0.0549948       -0.116835       -0.0614445      0.98835 0.0520913
        -0.0387397      -0.0200876      -0.15198        -0.0616083      0.985496


        MATRIZ R

        9.34396 -1.22753        -0.862899       1.0416  -0.596986
        -1.38778e-16    7.40924 -0.149968       -1.77243        -0.218319
        6.245e-17       2.94903e-17     8.2029  -0.929588       -2.31995
        -1.42247e-16    2.91434e-16     -4.16334e-17    7.49446 -0.840316
        -1.11022e-16    6.93889e-17     0       -2.77556e-16    7.10509


        || A - Q*R || = 2.28078e-15
```

Figura 3: Factorización QR de la matriz `mat5b.txt`

4. Conclusiones

El método de iteraciones en subespacios es un método eficiente para calcular algunos eigenvetores de la matriz A, resulta de gran utilidad cuando A es muy grande.

Variantes en la actualización de la matriz X por el método de la potencia pueden hacer que se encuentren los eigenvalores más pequeños en valor absoluto o bien aquellos que estén cerca de un valor en particular.

La factorización QR programada permitió encontrar las matrices Q y R de manera satisfactoria mediante el método de ortogonalización de Gram-Schmidt.