# Numerieke Modellering en Benadering: Practicum 1

vrijdag 21 maart 2014

## Eigenwaardenproblemen

In dit practicum onderzoeken we methoden voor het bepalen van eigenwaarden van volle matrices. In heel wat toepassingen maakt men gebruik van de eigenwaarden en eigenvectoren van een volle matrix. Bijvoorbeeld, wanneer men in de statistiek de covariantiematrix van een grote dataset berekent, kunnen de eigenwaarden en eigenvectoren gebruikt worden om de dataset te reduceren zonder belangrijke statistische informatie te verliezen. In het eerste gedeelte van het practicum beschouwen we enkele theoretische eigenschappen van de methoden. In het tweede gedeelte worden de convergentie-eigenschappen van de methoden onderzocht aan de hand van MATLAB-experimenten. Ten slotte in het derde gedeelte bekijken we de Jacobi-methode in meer detail.

## 1 Theoretische eigenschappen

**Opgave 1.** Gelijktijdige iteratie of 'simultaneous iteration' past de methode van de machten toe op meerdere kolommen tegelijk. Op die manier kunnen de $d$ grootste eigenwaarden en bijhorende eigenvectoren bepaald worden. Schrijf een variant van het algoritme uit om de $d$ kleinste eigenwaarden en eigenvectoren van een niet-singuliere matrix $A$ te bepalen. Schrijf het algoritme in pseudo-code uit, en geef ook een verklaring waarom het gewenste resultaat met dit algoritme bekomen wordt.

**Opgave 2.** Beschouw de Rayleigh quotiënt iteratie.

a) Naarmate de benaderende eigenwaarde $\lambda^{(k-1)}$ dichter in de buurt komt van de exacte eigenwaarde, wordt dit stelsel meer en meer singulier. Geef een beknopte (intuïtieve) verklaring waarom het in dit geval geen numerieke problemen geeft een (bijna) singulier stelsel op te lossen.

b) Gegeven een matrix $A \in \mathbb{R}^{m \times m}$ en een vector $x \in \mathbb{R}^{m \times 1}$, met $x$ een benadering voor een eigenvector van $A$. Toon aan dat de oplossing $\rho \in \mathbb{R}$ van het minimalisatieprobleem

$$\min_{\rho \in \mathbb{R}} ||Ax - \rho x||_2$$

overeenkomt met het Rayleigh quotiënt van $x$.

**Opgave 3.** Afhankelijk van welke eigenwaarden we van een matrix wensen te berekenen, is de ene methode geschikter dan de andere. Beschouw de methoden uit lecture 27 t.e.m lecture 29 van het boek 'Numerical Linear Algebra' (L. Trefethen en D. Bau, 1997). Bepaal voor de volgende situaties welke methode het meest geschikt is, rekening houdend met de convergentiesnelheid en de rekenkost.

a) Gegeven een schatting voor de eigenvector horend bij de derde kleinste eigenwaarde van een symmetrische matrix $A \in \mathbb{R}^{m \times m}$, bepaal de bijhorende eigenwaarde.

b) Gegeven een symmetrische matrix $A \in \mathbb{R}^{m \times m}$, bepaal de eigenwaarde het dichtst gelegen bij een getal $\alpha$.

Motiveer duidelijk je keuze en geef de verwachte convergentiesnelheid aan. Geef ook een benaderende beschrijving van de rekenkost, rekening houdend met het gevraagde en met de kost van een eventuele tridiagonalisatie van de matrix.

## 2 Convergentie-experimenten

Op Toledo vind je een aantal MATLAB-programma's die je kan gebruiken voor de rest van het practicum. Plaats de bestanden in een directory en gebruik in MATLAB het commando `cd` om naar deze directory te gaan. Lees grondig de documentatie bij de MATLAB-programma's. Typ `help pract` voor meer informatie.

Je kan experimenten uitvoeren door commando's in te geven aan de MATLAB invoerregel, maar het is aangewezen om zelf m-files (scripts en functies) te schrijven. Op die manier kan je gemakkelijk wijzigingen aanbrengen en vermijd je het herhaald intypen van gelijkaardige bevelen. Met het commando `help` kan je wat meer te weten komen over een bepaald programma. Je mag gerust de code van de programma's aanpassen. Je kan figuren afdrukken met het commando `print` of door in het menu File van een venster met een grafiek de optie Print te selecteren.

Enkele nuttige commando's zijn: `help`, `lookfor`, `figure`, `subplot`, `plot`, `semilogy`, `semilogx`, `loglog`, `xlabel`, `ylabel`, `title`, `legend`, `print`, `save`, `load`. Voor nog meer nuttige commando's zie ook de MATLAB inleiding op Toledo.

In het tweede deel van het practicum beschouwen we een gegeven volle, symmetrische matrix. Een voorbeeld is te vinden in het bestand `mat1.txt`. Deze matrix kan in MATLAB ingeladen worden met behulp van het commando `load`.

**Opgave 4.** Beschouw de matrix `mat1.txt`. Reduceer deze matrix naar Hessenberg vorm en toon de structuur met behulp van `spy`. Waarom is het belangrijk om de matrix te reduceren naar Hessenberg-vorm alvorens de QR-methode toe te passen?

**Opgave 5.** In deze opgave bekijken we

- de QR-methode zonder shift,
- de QR-methode met Rayleigh quotiënt shift, en
- de QR-methode met Wilkinson shift.

Pas deze methoden toe om alle eigenwaarden van de matrix `mat1.txt` te berekenen. Illustreer grafisch en in tabelvorm de convergentie. Is de convergentie lineair, kwadratisch of kubisch? Komen de numerieke resultaten overeen met de theorie? Bespreek.

**Opgave 6.** De cursus vermeldt een verband

- tussen het QR-algoritme met Rayleigh quotiënt shift en de Rayleigh quotiënt iteratie, en

- tussen het QR-algoritme en de gelijktijdige iteratie.

Verduidelijk aan de hand van de matrix `mat1.txt` in welk opzicht het convergentiegedrag van het *QR-algoritme met Rayleigh quotiënt shift* gelijkenissen vertoont met de twee andere methoden.

**Opgave 7.** Naast (quasi)-directe methoden om eigenwaardenproblemen op te lossen, kunnen we ook iteratieve methoden beschouwen, zoals de Arnoldi methode. Iteratieve methoden zijn in het bijzonder geschikt wanneer de matrix een ijle structuur heeft. Genereer een random, ijle matrix $A \in \mathbb{R}^{1000\times 1000}$ met het commando `sprand`. Pas een aantal Arnoldi-iteratiestappen toe op deze matrix (bv. 100). Bereken per iteratiestap de Ritz waarden. Daarbij mag je gebruik maken van het ingebouwde MATLAB-commando `eig`. Maak een grafiek waarin je toont hoe de Ritz waarden per iteratiestap convergeren naar de eigenwaarden van $A$. Toon daarbij enkel de reële delen. Bespreek bondig het convergentiegedrag.

## 3 Alternatieve eigenwaardenalgoritmen

Naast de besproken methoden in lecture 27 t.e.m. lecture 29 van het boek 'Numerical Linear Algebra' (L. Trefethen en D. Bau, 1997) bestaan er nog andere belangrijke methoden voor het oplossen van eigenwaardenproblemen. Enkele voorbeelden zijn Jacobi, bisectie, verdeel- en heers en Arnoldi/Lanczos. In dit gedeelte van het practicum bekijken we de Jacobi-methode in meer detail.

De Jacobi-methode steunt op de diagonalisatie van $2 \times 2$ symmetrische matrices met behulp van een orthogonale matrix $J$,

$$J^T \begin{bmatrix} a & d \\ d & b \end{bmatrix} J = \begin{bmatrix} \neq 0 & 0 \\ 0 & \neq 0 \end{bmatrix}, \tag{1}$$

waarbij de orthogonale matrix $J$ gedefinieerd wordt als een rotatie-matrix of een reflectie-matrix. Hier beschouwen we enkel rotatie,

$$J = \begin{bmatrix} c & s \\ -s & c \end{bmatrix}. \tag{2}$$

**Opgave 8.** Bepaal de coëfficiënten $c$ en $s$ van de matrix $J$ in (2) zodat (1) geldt. Geef ook je afleiding.

**Opgave 9.** Schrijf een algoritme uit in pseudo-code om met de Jacobi-methode de eigenwaardenontbinding $VDV^T$ van een symmetrische matrix $A$ te berekenen, waarbij $D$ een diagonaalmatrix is die de eigenwaarden bevat, en $V$ een orthogonale matrix met de overeenkomstige eigenvectoren. Maak gebruik van een standaard rij per rij ordening. Elke eigenwaarde moet bepaald worden met een fout kleiner dan een opgeven tolerantie `tol`. Implementeer dit algoritme als een functie in MATLAB: `function [V,D]= jacobi(A,tol)`.

**Opgave 10.** Pas de Jacobi-methode toe om alle eigenwaarden en eigenvectoren van `mat1.txt` te berekenen. Bekijk de convergentiesnelheid. Komt de experimentele convergentiesnelheid overeen met de theorie? Wat is de rekenkost van de methode voor een willekeurige symmetrische tridiagonale matrix $A \in \mathbb{R}^{m\times m}$?

## Praktische richtlijnen

- Dit practicum wordt gemaakt in groepjes van twee.
- Het verslag kan je afgeven op mijn bureel of op het secretariaat van het departement computerwetenschappen ten laatste op **vrijdag 25 april 2014** om 15:00 uur. Je mag het ook in elektronische versie naar mij mailen.
- Gebruik figuren en tabellen om je bevindingen te verduidelijken. Gebruik logaritmische grafieken waar nodig. Kies de schaal van je figuren oordeelkundig, vooral als twee verschillende figuren vergeleken moeten worden.
- Bespreek bondig je resultaten en je aanpak in een duidelijke vergezellende tekst. Hou de lengte evenwel onder de 15 pagina's, inclusief tabellen en figuren.
- Voeg naast de pseudo-code ook je Matlab-code voor opgave 9 toe als bijlage aan je verslag.

Veel succes!

Ben Jeuris (200A 01.160)
`ben.jeuris@cs.kuleuven.be`
`http://toledo.kuleuven.be`