

mLinis – Matlab-tehtäviä lineaarialgebrasta

1. mLi001.tex

Ohjetiedosto, poimi mukaan tehtäväpaperiin tarpeen mukaan.

Matlab-ohjeita

- Komennon suorittama tulos tulee ruudulle ENTER-painalluksen jälkeen (kuvat erilliseen ikkunaan). Jos haluat estää tulostuksen, pääätä komento puolipisteeseen. Jos myöhemmin haluat katsoa muuttujan sisällön, kirjoita sen nimi (ilman puolipistettä). Jos muuttuja on suuri matriisi, kannattaa ensin katsoa sen koko `size(A)` tai sen jotain osaa, esim. `A(1:10,1:10)`
- Edellisen komennon tulos on muuttujassa `ans`. Yleensä on suositeltavaa antaa tulokselle oma nimi tyyliin `nimi= ...`
- `format long` : Tulostetaan enemmän numeroita (n. 16). Laskutarkkuuteen tämä ei vaikuta.
`format rational` laskee rationaaliluvuilla.
`format short`: Paluu oletustulostukseen.
- Matriisin A transpoosi: `A'`
- Kokonaislukuvektori: Esim `1:10` tai `1:2:20`. Myös `linspace`. Pystyvektoriksi transponoimalla.
- `A(i,j)` A:n alkio (i,j).
`A(2,:)` A:n 2. rivi
`A(:,3)` A:n 3. sarake
`A(1:4,1:4)` osamatriisi
Matriisin osaa voi päivittää, vaikkapa:
`A(1:4,1:4)=ones(4,4)` tai
`A(2,:)=A(2,:)-2*A(:,1)` (Gaussin rivioperaatio).
- Matriisien liittäminen: Jos A:lla ja B:llä on yhtä monta riviä, ne voidaan liittää peräkkäin: `[A b]` (tai `[A, b]`)
Jos yhtä monta saraketta, niin allekkain: `[A;B]`
- Laskutoimitukset tarkoittavat matriisilaskua. Siis esim.
`A*B`, `A^p`
- Vektorien ja matriisien (samankokoisten) pisteittäinen eli alkioittainen laskenta tapahtuu lisäämällä eteen piste. Esim:
`u=[1 2 3]`, `v=[-2 -2 -2]`, `u.*v`.
Toinen operandi voi olla skalaari.
Siten esim. vektorin *u* kaikki komponentit voidaan korottaa toiseen komennolla
`u.^2`

Avainsanat:

Ohjetiedosto, Harjoitusohjeita, Lineaarialgebra, Matlabperusteet,

2. mLi002.tex
Olkoon

$$\mathbf{A} = \begin{bmatrix} 5 & -13 & -13 & -2 & 7 \\ 18 & -4 & 30 & -1 & -12 \\ -23 & 3 & 7 & 15 & 7 \\ 9 & 36 & -1 & 14 & 16 \\ 3 & 28 & 7 & 14 & 5 \end{bmatrix}$$

ja

$$\mathbf{b} = \begin{bmatrix} -196 \\ 435 \\ 11 \\ 111 \\ 195 \end{bmatrix}.$$

Ratkaise yhtälösystemi $Ax = b$ ja tarkista tulos matriisikertolaskulla.

Avainsanat:

Lineaarinen yhtälöryhmä, Matlabperusteet, Matlabalkeet, perusmatriisilaskenta

3. mlLi003.tex [myös Maple, Mathematica]

- a) Ratkaise lineaarinen yhtälöryhmä ja tarkista tulos kertolaskulla.

$$\begin{cases} 4x - 5y = 11 \\ 2x + y = 9 \end{cases}$$

Vihje: Matlab: "Matriisijako": $A \setminus b$

- b) Tiedetään, että Celsius-asteiden ja Fahrenheit-asteiden välillä on lineaarinen yhteys:

$$C = aF + b.$$

Lisäksi tiedetään, että vesi jäätyy 32 F:ssa ja -40 on sama kummassakin asteikossa. Johda kaava. Tarkoitusta kirjoittaa kertoimien a ja b määrittämiseksi lineaarinen yhtälösystemi, joka ratkaistaan Matlab:n takakenolla (\setminus).

- c) Muodosta matriisi, jonka 1. sarake on C-asteet -50 :sta 5 :n asteen välein 100 :aan ja toinen sisältää vastaavat F-asteet.

Vihje: Tarkan rationaalilukukaavan saat komentamalla `format rat`. Tee m-tiedosto kommentteineen.

Huomaa, että taulukkoa ei ole mukavaa katsoa kokonaisuuksina, esim. 10 ekaa riviä näet näin: `taulukko(1:10, :)` (eikö vain?).

Hivelevää on myös mennä "Workspace-ikkunaan" ja kaksoisklikata taulukko-ikonin.

Kokeile sen ajamista myös pdf:ksi `publish(Fahrenheit, pdf)`-komennolla (jos skripti on `Fahrenheit.m`), kunhan ensin testat sen kuntoon.

4. mlLi004.tex Matlab/Maple/Mathematica

Tarkastellaan yhtälösystemejä:

$$\begin{cases} 10x_1 + 7x_2 + 8x_3 + 7x_4 = 32 \\ 7x_1 + 5x_2 + 6x_3 + 5x_4 = 23 \\ 7x_1 + 6x_2 + 10x_3 + 9x_4 = 33 \\ 7x_1 + 5x_2 + 9x_3 + 10x_4 = 31 \end{cases}$$

ja

$$\begin{cases} 2x_1 + x_2 + 5x_3 + x_4 = 9 \\ x_1 + x_2 - 3x_3 - x_4 = -5 \\ 3x_1 + 6x_2 - 2x_3 + x_4 = 8 \\ 2x_1 + 2x_2 + 2x_3 - 3x_4 = 3 \end{cases}$$

- a) Ratkaise molemmat systeemit.
- b) Muuttamalla vähän yhtälön dataa (oikeaa puolta ja/tai kerroinmatriisia), voidaan tutkia systeemin herkkyyttä pienille virheille (datassa ja pyöristyksessä).

Ratkaise 1. systeemi oikean puolen vektoreilla

$[32.1, 22.9, 32.9, 31.1]'$ ja $[32.01, 22.99, 32.99, 31.01]'$

ja 2. systeemi vektoreilla

$[9.1, -5.1, 7.9, 3.1]'$ ja $[9.01, -5.01, 7.99, 3.01]'$.

Mitä nämä pienet häiriöt vaikuttavat ratkaisuihin?

- c) Muuta kerroinmatriiseja lisäämällä matriisien kuhunkin alkioon pieni satunnaisluku $0.1 * \mathbf{rand}$. Ratkaise systeemit alkuperäisillä oikeilla puolilla. Mitä nämä muutokset vaikuttavat ratkaisuihin.
- d) Lineaarisen yhtälösystemin herkkyyttä pienille virheille sanotaan *häiriöalttiudeksi* ("ill-conditioned"). Laske kummankin matriirin häiriöalttius.

Suhteellisen virheen suurtenemisypähtälö:

$$\frac{\|\Delta x\|}{\|x\|} \leq \kappa \frac{\|\Delta b\|}{\|b\|}$$

Pahimmillaan ratkaisun suhteellinen virhe voi olla luokkaa $\kappa \times$ (datan suhteellinen virhe) ($\kappa = \mathit{cond}(A)$)

Luokittelu:

mplteht/mplLinis/mplLixx.tex, matlabteht/mlLinis/mlLixx.tex

Avainsanat:

Numeerinen lineaarialgebra, matriisit, lineaariset yhtälöryhmät, häiriöalttius

5. mLi005.tex

Olkoon

$$\mathbf{A} = \begin{bmatrix} 5 & 1 & 1 \\ 1 & 5 & 1 \\ 1 & 1 & 5 \end{bmatrix}, \mathbf{b} = \begin{bmatrix} 10 \\ 14 \\ 18 \end{bmatrix}.$$

Varmista ensin, että matriisi \mathbf{A} on kääntyvä laskemalla $\det(\mathbf{A})$.

Tutki, mitä muita keinoja on matriisin ei-singulaarisuuden tarkistamiseen. Kokeile vaikka **rank**, **rref**, **lu**, **cond**, **rcond** (katso helpillä).

(Huomaa, että “oikeissa tehtävissä” tärkeämpi käsite on “lähes singulaarisuus”, tähän \det ei ole yleispätevä työkalu.)

Ratkaise yhtälöryhmä $\mathbf{Ax} = \mathbf{b}$ käyttämällä

a) Käänteismatriisia `inv`

b) MATLABin matriisijakoa `x = A\b`.

Opetus: Huomaa, että “matriisijako” on numeerisen tarkkuuden ja laskentatehon kannalta yleensä parempi tapa (mikä ei pienissä, hyvänlaatuisissa tehtävissä tule ilmi).

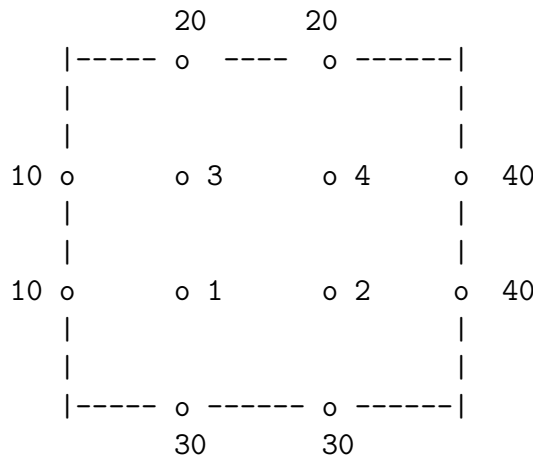
Opettajalle: Tehtävään voidaan lisätä myös \mathbf{A} -matriisin muodostaminen diagonaalittain **diag**-funktioilla (vaikkei mene hyödyn puolelle näin pienessä tehtävässä).

6. mLi006.tex

Tasapainolämpötilajakauma metallilevyssä.

Kuva esittää metallilevyä, joka on ylä- ja alapinnoiltaan lämpöeristetty ja jonka reunojen lämpötilat on kiinnitetty. (Lämpöä virtaa vain reunojen kautta.) Tasapainolämpötilajakauma saadaan *Laplacen yhtälön* $\Delta u = 0$ ratkaisuna. Numeerinen approksimaatio voidaan laskea ns. differenssimenetelmällä: Jaetaan levy sopivilla hilaviivoilla osiin ja numeroidaan näin muodostuvat solmupisteet. Menetelmä: Kunkin hilasolmun lämpötila on naapurisolmujen lämpötilojen keskiarvo. (Johdetaan kurssin lopulla.)

Muodosta 4×4 - yhtälösystemi solmujen 1, 2, 3, 4 lämpötilojen likiarvoille u_1, u_2, u_3, u_4 . Ohje: Aloitetaan solmusta 1: $u_1 = \frac{1}{4}(30 + u_2 + u_3 + 10)$. Vastaavasti muut kolme solmua.



- (a) Ratkaise yhtälösystemi ja sijoita ratkaisulämpötilat ao. hilapisteisiin.
- (b) Muodosta 4×4 - matriisi, jossa on reunalämpötilat ja ratkaisemasi sisäpistelämpötilat sekä nurkissa lähinnä olevien kahden reunasolmun keskiarvot tähän tapaan: $U = [5 \ 20 \ 20 \ 30; 10 \ u_3 \ u_4 \ 40; 10 \ u_1 \ u_2 \ 40; 20 \ 30 \ 30 \ 35]$; Piirrä ratkaisupinnan approksimaatio: `mesh(U)` tai `surf(U)`.

Avainsanat: Lämpötilamatriisi, Laplacen yhtälön diskretointi, differenssimenetelmän alkeellisin perustehtävä, lineaarinen yhtälöryhmä.

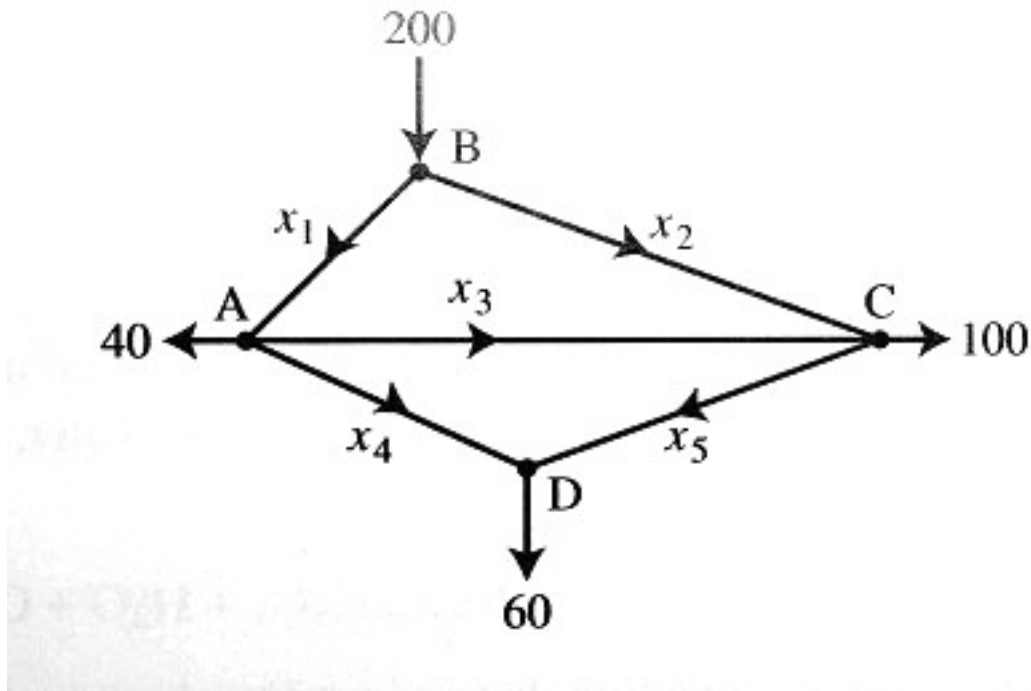
7. mLi007.tex

Oheinen kuva esittää liikenneverkkoa. Kuhunkin solmuun A,B,C,D tulevien ja siitä lähtevien ajoneuvojen lukumäärien summa pysyy samana (solmuun ei häviä eikä siinä synny ajoneuvoja). Kadut ovat yksisuuntaisia nuolien osoittamalla tavalla.

Kuvan saat mukaan tehtävääsi, kunhan kopioit

polku/img/liik.eps-tiedoston omaan img-hakemistoosi. Tässä:

polku=<http://www.math.aalto.fi/opetus/Mattie/MattieT/matlabteht/mLLinis/>



- Muodosta yhtälösystemi tuntemattomien ajoneuvomäärien x_1, \dots, x_5 suhteen.
- Määritä systeemin yleinen ratkaisu.
- Jos x_4 :llä merkitty katuosuus suljetaan, niin mikä on yleinen ratkaisu?
- Määritä kohdan (c) tilanteessa pienin x_1 :n ja suurin x_3 :n arvo (jotta yhdensuuntaisuutta osoittavia liikennemerkkejä ei tarvitse kääntää).

Huom! Porrasmuotoon saattamisessa saat halutessasi käyttää Matlab/Octave-funktiota `rref` (kts. `help rref`).

Avainsanat: Liikenneverkko, lineaarinen yhtälöryhmä, (reduoitu)porrasmuoto, rref.

8. mLi008.tex

1. Muodosta 5×5 -yksikkömatriisi I . (`help eye`)
2. Muodosta matriisi E_1 , jossa on vaihdettu I :n rivit 2 ja 5.
3. Muodosta matriisi E_2 , joka saadaan kertomalla I :n 4. rivi luvulla 4.
4. Muodosta matriisi E_3 , joka saadaan I :stä *Gaussin rivioperaatiolla*:

$$r_4 \leftarrow r_4 + 4r_1,$$

missä r_i tarkoittaa matriisin riviä numero i .

5. Muodosta matriisit E_1^{-1} , E_2^{-1} , E_3^{-1} käyttäen komentoa `inv` ja selitä, mitä rivioperaatiota ne vastaavat.

Avainsanat: Alkeismatriisit, LU-hajotelma, Gaussin rivioperaatio, käänteismatriisi.

9. mLi009.tex (KP3-ii, 2008, harj1)

Olkoon

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \\ a_{3,1} & a_{3,2} & a_{3,3} \end{bmatrix}$$

ja olkoot

$$E_0 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad E_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 3 \end{bmatrix}, \quad E_2 = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Muodosta matriisitulot E_0A , E_1A , AE_1 ja E_2A ja selvitä, mitä nämä operaatiot tekevät matriisin A riveille/sarakkeille.

Avainsanat: Alkeismatriisit, LU-hajotelma, Gaussin rivioperaatio

Vihje: Käsineläskä ja ajattelutehtävä, tarkistukseen voit hyödyntää Matlabin `syms`-komentoa tai voit tehdä symboliset matriisioperaatiot Maplella/Mathematicalla.

10. mLi010.tex

Harjoituksen (KP3-II/s. 2006) ohjetta:

Seuraavissa tehtävissä voitaisiin johonkin johtopäätökseen päästä determinantin avulla. Näissä harjoituksissa ei kelpuuteta tällaisia ratkaisuja, vaan harjoitellaan johtopäätösten tekoa rivioperaatioiden seurauksena.

Osa tehtävistä on käsinlaskuun tarkoitettu, mutta niiden yhteydessä voidaan harjoitella samalla Matlab/Octave/Scilab-työskentelyä (Kts. MattieO).

Annettuna on 3×3 -systeemin liitännäismatriisi

$$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 4 & 5 & 6 & 7 \\ 6 & 7 & 8 & 9 \end{bmatrix}.$$

Muodosta rivioperaatioilla porrasmuoto "ref" — "row echelon form". Merkitse tukisarakeet ja tukialkioiden paikat. Jatka sitten rivioperaatioita alhaalta ylöspäin päästäksesi redusoituun porrasmuotoon "rref".

Avainsanat: Lineaarinen yhtälöryhmä, Gaussin rivioperaatio, ref, rref, (redusoitu) porrasmuoto, row echelon form.

11. mLi013.tex

Piirretään toisen asteen pintoja. Tätä varten tulee pinnat esittää parametrimuodossa

$$x_1 = x_1(u, v), x_2 = x_2(u, v), x_3 = x_3(u, v),$$

missä muuttujat u ja v saava arvoja jostain sopivasta alueesta. Esimerkiksi parametrisointi

$$\begin{cases} z_1 = r_1 \sin u \cos v, \\ z_2 = r_2 \sin u \sin v \\ z_3 = r_3 \cos u, \end{cases} \quad \begin{cases} 0 \leq u \leq \pi \\ 0 \leq v \leq 2\pi \end{cases}$$

MATLAB ilmoittaa \mathbb{R}^2 :n muuttujat tietyllä tavalla organisoituihin matriiseihin seuraavasti:

```
[U,V] = meshgrid(linspace(0,pi,21),linspace(0,2*pi,21));
```

Nyt ellipsoidin $r_1 = r_2 = r_3$ parametrisointi tehdään seuraavasti

```
Z1 = sin(U).*cos(V);  
Z2 = sin(U).*sin(V);  
Z3 = cos(U);
```

Kuvan tästä saa piirrettyä komennolla `surf(Z1,Z2,Z3)`. Kokeile miten pinta muuttuu, kun asetat kertoimiksi r_k eri arvoja. Tutustu myös komentoon `axis`.

12. mLi014.tex

Tarkastellaan yhtälöryhmää

$$\begin{cases} 0.0001x_1 + 2x_2 = 4 \\ x_1 + x_2 = 3 \end{cases}$$

Tarkka ratkaisu on $[\frac{2}{1.9999}, \frac{3.9997}{1.9999}]^T$, joka 5:llä numerolla esitettynä on $[1.0001, 1.9999]^T$.

(a) Ratkaise yhtälösystemi niin, että suoritat laskut (järjestystä vaihtamatta) 3:lla merkitsevällä numerolla. (Laske laskimella, Matlabilla tms. ja pyöristä kunkin operaation jälkeen tulos 3:een numeroon.)

(b) Tee samoin kuin (a)-kohdassa, mutta vaihda yhtälöiden järjestys.

Selitä, miksi (a)-tapauksessa tulee suuri suhteellinen virhe (100%:n suhteellinen virhe toisessa komponentissa), kun taas (b)-tapauksessa virhe on olematon.

Avainsanat: Lineaarinen yhtälöryhmä, Gaussin rivioperaatio, numeerinen ratkaisu, numeerinen lineaarialgebra, pyöristysvirhe.

13. mlLi015.tex

Olkoon $A = \begin{bmatrix} 1 & 8 & 6 \\ -1 & -4 & 5 \\ 2 & 4 & -6 \end{bmatrix}$ ja $b = [8 \ 1 \ 4]^T$. Ratkaise yhtälö $Ax = b$ osittaistuentaa (“partial pivoting”) käyttäen.

Olkoon P permutaatiomatriisi (rinvaihtomatriisi), joka määräytyy rivien vaihdoista. Muodosta hajotelma $PA = LU$.

Matlab:lla: `help lu, [L,U,P]=lu(A)` (Tämä siis vertailun vuoksi, tarkoitus on laskea käsin.)

Avainsanat: Lineaarinen yhtälöryhmä, Gaussin rivioperaatio, LU-hajotelma, numeerinen lineaarialgebra, (osittais)tuenta, “(partial) pivoting”.

Vihje: Osittaistuenta tarkoittaa itseisarvoltaan suurimman tukiakion valitsemista pienen tukiakion aiheuttamien numeeristen ongelmien välttämiseksi. Matlab saattaa käyttää esim. ns. skaalattua osittaistuenta, jolloin rinvaihtostrategia voi olla erilainen.

14. mlLi017.tex

Muodosta “ylimääräytyvälle” yhtälöryhmälle

$$\begin{cases} x + 3y = 5 \\ x - y = 1 \\ x + y = 0 \end{cases}$$

normaaliyhtälöt ja ratkaise pienimmän neliösumman (PNS,LSQ) mielessä. Piirrä suorat ja ratkaisupiste tasoon.

Vastaustarkistuskeino: Huomaa, että Matlab:n yhtälösystemin ratkaisija: $x = A \setminus b$ on niin älykäs, että se ymmärtää ylimääräytyvässä tapauksessa suorittaa PNS-ratkaisun.

Avainsanat: Lineaarinen yhtälöryhmä, PNS,LSQ, pienimmän neliösumman menetelmä, LU-hajotelma, numeerinen lineaarialgebra, (osittais)tuenta, “(partial) pivoting”.

Vihje:

15. mLi018.tex

Huom: Käsinlasku täydennettynä pikku Matlab-osuudella.

Eräässä mittauksessa saatiin seuraava data:

xdata	1	2	3	4	5
ydata	1.8	2.7	3.4	3.8	3.9

Dataa mallinnetaan polynomilla $p(x) = c_1 x + c_2 x^2$.

(a) Muodosta PNS-tehtävän matriisi X ja vektori y siten, että tehtävä saadaan ylimääräytyväksi yhtälöryhmäksi $Xc = y$.

(b) Ratkaise kerroinvektori c . Piirrä data ja PNS-polynomi samaan kuvaan.

Vihje: (b)-kohdassa saat mieluusti käyttää Matlab:ia. Tee kuitenkin vaiheittain matriikertolaskut, transpoosit ym., lopuksi toki voit tarkistaa "takakenolla". Piirrä samaan kuvaan datapisteet ja polynomi.

Piirtäminen käy näin:

```
xd=1:5; yd=[1.8 ...]; plot(xd,yd,'x'); hold on; kertoimet=[c2 c1 0]; x=linspace(1,5); y=polyval(C, x); plot(x,y,'r'); xlim([0 6]); grid on
```

Huomaa, että *polyval* haluaa kertoimet korkeimmasta potenssista alkaen.

Vast: $p(x) = 1.76x - 0.2x^2$

Avainsanat: PNS,LSQ, pienimmän neliösumman menetelmä, käyrän sovitus, curve fitting, data fitting.

16. mLi019.tex

Määritä PNS-ratkaisu tehtävälle $Ax = b$, kun $A = \begin{bmatrix} 1 & -1 \\ 1 & 4 \\ 1 & -1 \\ 1 & 4 \end{bmatrix}$ ja

$b = \begin{bmatrix} -1 & 6 & 5 & 7 \end{bmatrix}^T$. Hyödynnä QR-hajotelmaa, jonka voit muodostaa seuraavilla komennoilla: (Huom: Yleensä ei lasketa rationaaliaritmetiikalla, mutta opettelussa voi olla hyödyksi.)

```
>> format rational
>> A=[...] % Jos kirjoitat [...], olet TONTTU!
>> [Q,R]=qr(A)
```

Vihje: Matlab muodostaa ns. täyden QR-hajotelman. Kuten huomaat, riittää ottaa Q:n kaksi ensimmäistä saraketta ja R:n 2 ensimmäistä riviä, miten nyt vain haluat. Huomaa siis, että Q on ortogonaalinen ja R on yläkolmiomatriisi.

Avainsanat: PNS,LSQ, pienimmän neliösumman menetelmä,QR-hajotelma.

17. mlLi020.tex

Suorita Matlab-komento `eigshow`. Opiskele helppiteksti ja suorita joitakin kokeiluja kuljettamalla x -vektoria läpi koko yksikköympyrän. (Tämä vain lämmittelyksi.)

Valitse erityisesti matriisit $A=[1\ 3;4\ 2]/4$, $B=[3\ 1;-2\ 4]/4$ ja $C=[2\ 4;2\ 4]/4$. Määritä kuvan perusteella kunkin ominaisarvot ja -vektorit. Saat vektorit tarkemmin komentamalla `grid on`.

Mitä, jos kuvan perusteella ominaisarvoja/vektoreita ei näyttäisi olevan?

Määritä kuvan perusteella myös ominaisvaruuden dimensio matriisin C tapauksessa.

Laske kunkin matriisin ominaisarvot ja -vektorit `eig`-komennolla. (`help eig`)

Huom: Jos näitä matriiseja ei sattuisi olemaan valmiina valikossa, voit ne sinne lisätä helppiruudun “View code for eigshow”-linkistä. Hae koodista kohta `mats = ...`. Siitä näet, miten matriiseja voi lisätä. Jos editoit koodia, tallenna se omaan hakemistoosi vaikkapa nimelle `ominashow`, ja sitten vaan `ominashow`.

Tee joitakin omia kokeiluja erilaisilla matriiseilla oman “`ominashow`”:n avulla.

Avainsanat: Ominaisarvot, ominaisvektorit, ominaisarvojen graafinen havainnollistaminen, ominaisvektorien graafinen havainnollistaminen, Matlab: `eigshow`, `eig`

Ohjeita, ominaisarvo-oppia (Liitettäväksi aiheen tehtäväpaperiin)

- **Ominaisarvo** on luku, se voi olla kompleksiluku, vaikka matriisi olisi reaalinen.
- **Ominaisvektori** on (reaalisen matriisin tapauksessa) \mathbb{R}^n :n tai \mathbb{C}^n :n vektorin mukaan, onko vastaava ominaisarvo reaalinen vai kompleksinen.
- Ominaisarvo saa aivan mainiosti olla 0, ominaisvektoriksi emme hyväksy nollavektoria.
- Ominaisarvoon λ liittyvä **ominaisavaruus** E_λ koostuu kaikista λ :aan liittyvistä ominaisvektoreista ja lisäksi nollavektorista. Tällöin kyseessä on vektori(al)avaruus, nimittäin matriisin $A - \lambda I$ nolla-avaruus, $N(A - \lambda I)$.
- Ominaisarvon λ_j **algebraallinen kertaluku** M_{λ_j} on karakteristisen polynomin $\det(A - \lambda I)$ juuren kertaluku. **Geometrisen kertaluku** m_{λ_j} on $\dim(E_{\lambda_j})$. Pätee: $m_{\lambda_j} \leq M_{\lambda_j}$
- Jos reaalilla matriisilla A on **kompleksinen** ominaisarvo $\lambda = \alpha + i\beta$, niin myös $\bar{\lambda} = \alpha - i\beta$ on A :n ominaisarvo. Jos \mathbf{v} on λ :aa vastaava ominaisvektori, niin liittolukua $\bar{\lambda}$ vastaava ominaisvektori on $\bar{\mathbf{v}}$. (Tarkoittaa vektoria, jonka koordinaatit ovat \mathbf{v} :n koordinaattien liittolukuja.)
- Jos on määrättävä diagonaalimatriisin ominaisarvot ja -vektorit, niin laskentatyötä ei jää lainkaan. Älä siis suotta ryhdy veivaamaan $\det(A - \lambda I)$:n kautta. (Koko ominaisarvohomman perustavoite on saattaa lineaarikuvauksen matriisi diagonaalimuotoon. Jos se jo on, niin mitään ei tarvitse enää tehdä, kunhan osaat siitä lukea.)
- Kolmiomatriisin (ylä- tai ala-) ominaisarvot ovat diagonaalialkiot. (Siis yleistys edelliselle, tässä tapauksessa ominaisvektoreista ei voida sanoa mitään yleistä.)
- Kun pyydetään laskemaan johonkin ominaisarvoon liittyvät ominaisvektorit, on sopivaa antaa vastaukseksi ominaisavaruuden kanta. Helpoimmin se saadaan antamalla ratkaisun vapaille muuttujille vuorollaan arvot $(1, 0, 0)$, $(0, 1, 0)$, $(0, 0, 1)$ (jos kyseessä on 3-ulotteinen ominaisavaruus). Tässä on kyse nolla-avaruuden kannan määräämistehtävästä.
- Eri ominaisarvoihin liittyvät ominaisvektorit ovat LRT.
- Diagonalisointi: Annettu A . Etsittävä, jos mahdollista, matriisit V ja D , V kääntyvä ja D diagonaalimatriisi siten, että $A = VDV^{-1}$. Jos tehtävänä on diagonalisoida A , etsitään matriisit V ja D ja perustellaan V :n kääntyvyys. (Yleensä ei vaadita V^{-1} :n laskemista ilman eri kehoitusta, tai jatkotehtävän asettamaa tarvetta.)
- Octave/Matlab-komentoa `eig` kannattaa käyttää ainakin tarkistukseen. Muoto `[V,D]=eig(A)` antaa suoraan diagonalisointimatriisit: V :n sarakkeina ominaisvektorit ja D :n diagonaalilla (samassa järjestyksessä) ominaisarvot. Jos A on diagonalisoituva, niin V :n sarakkeet ovat LRT, jolloin voidaan muodostaa V^{-1} ; Matlab/Octavella: `inv(V)`.

19. mlLi021.tex (käsinelasku, Matlab sopii avuksi/opiksi)

Muodosta matriisin $A = \begin{bmatrix} 3 & -2 & 4 \\ -2 & 6 & 2 \\ 4 & 2 & 3 \end{bmatrix}$ ortogonaalinen diagonalisointi (tarkoittaa ortonormaalia).

Laskutyön vähentämiseksi annetaan (tai pyydetään oppilasta komentamaan):

```
>> eig(A)
ans =
    -2.00
     7.00
     7.00
```

Avainsanat: Ominaisarvot, ominaisvektorit, ortogonaalinen diagonalisointi.

Vihje: Muista, että ominaisvektorit eivät automaattisesti ole yksikkövektoreita, ja useampiker- taista ominaisarvoa vastaavat ominaisvektorit eivät automaattisesti ole ortogonaaliset.

Jos olet saanut samaan ominaisvaruuteen kuuluvat LRT ominaisvektorit v_1 ja v_2 , niin ortonor- maalin kannan saat

1) geometrisen ajattelun avulla: Muodosta v_2 :n kohtisuora projektio v_1 :llä ja vähennä se v_2 :sta. Tai

2) algebrallisesti: Määritä kerroin c siten, että $(v_1 + cv_2) \perp v_1$.

20. mlLi028.tex

Olkoon

$$\mathbf{A} = \begin{bmatrix} 5 & 1 & 1 \\ 1 & 5 & 1 \\ 1 & 1 & 5 \end{bmatrix}$$

- a) Laske matriisin \mathbf{A} diagonalisointiin tarvittavat matriisit P ja D .
- b) Varmista, että P on ortogonaalinen, ja D on diagonaalinen ja diagonaalialkiot suuruusjärjestyksessä.
- c) Osoita, että spektraalikaava

$$\lambda_1 \mathbf{u}_1 \mathbf{u}_1^T + \lambda_2 \mathbf{u}_2 \mathbf{u}_2^T + \lambda_3 \mathbf{u}_3 \mathbf{u}_3^T = A$$

pätee.

Vihje:

21. mLi029.tex

Potenssimenetelmä on eräs keino löytää itseisarvoltaan suurin ominaisarvo ja vastaava ominaisvektori. Menetelmä toimii seuraavasti:

- Valitse alkuarvaus \mathbf{b}_0 . Ainoa vaatimus on, että tällä vektorilla on nollasta poikkeava komponentti ominaisarvon suuntaan – käytännössä kannattaa valita vektori, jonka jokainen alkio on nollasta poikkeava.
- Aseta

$$\mathbf{b}_{k+1} = \frac{\mathbf{A}\mathbf{b}_k}{\|\mathbf{A}\mathbf{b}_k\|}$$

- Jatka kunnes jono (\mathbf{b}_k) suppenee. Ominaisarvo $\lambda = \|\mathbf{A}\mathbf{b}_k\|$ ja vektori $\mathbf{x} = \mathbf{b}_k$.

Toteuta menetelmä MATLABissa, ja laske matriisin `gallery(5)` suurin ominaisarvo ja vastaava ominaisvektori. Testaa tuloksen oikeellisuus.

22. mLi030.tex

Ominaisarvojen laskentamenetelmiä, Power method [KRE⁹] Sec. 20.8
Sovella potenssimenetelmää (3 kierrosta) matriisiin

$$A = \begin{bmatrix} 3.5 & 2.0 \\ 2.0 & 0.5 \end{bmatrix}$$

alkuarvolla $x_0 = [1, 1]^T$. Laske *Rayleigh-osamäärät* q ja virherajat.

Ratkaisu: Vastaus: $q = 4, 4.493, 4.4999; |\epsilon| \leq 1.5, 0.1849, 0.0206$

Avainsanat: Potenssimenetelmä, ominaisarvojen laskentamenetelmät, Power method.

23. mLi031.tex

Osoita, että jos x on ominaisvektori, niin $\delta = 0$ virhekaavassa (2) Theorem 1 s. 872 (KRE⁹, luvun 20.8, Power Method for Eigenvalues alkusivulla) .

Vihje: Päätelytehtävä, ohjelmistoista ei hyötyä.

Avainsanat: Potenssimenetelmä, ominaisarvojen laskentamenetelmät, Power method.

24. mLi032.tex

(a) Päättelä Gershgorinin lauseen avulla matriisiin

$$A = \begin{bmatrix} 11 & 0.4 & -0.5 \\ 0.4 & 7 & a \\ -0.5 & a & 4 \end{bmatrix}$$

ominaisarvojen likiarvot ja missä rajoissa ne ovat.

(b) Millä a :n reaaliarvoilla nähdään suoraan, että matriisi on kääntyvä. (Tarkoitus ei ole laskea determinanttia tai ominaisarvoja, korkeintaan halutessasi tarkistukseksi ja varmistukseksi Gershgorinin pätevyydelle.)

Vihje: Käsinlasku, jossa voit harjoitella Matlabin laskinkäyttöä.

Avainsanat: Gershgorinin lause , ominaisarvojen laskentamenetelmät, ominaisarvoarvio.

25. mLi040.tex

Olkoot \mathbf{A} kompleksinen $n \times n$ matriisi, ja olkoot $R_i = \sum_{j \neq i} |a_{ij}|$, eli rivin alkioiden itseisarvojen summa diagonaalia lukuunottamatta. Gershgorinin kiekkolauseen väite on, että jokainen matriisin \mathbf{A} ominaisarvo λ_i sijaitsee jossakin kiekossa $D(a_{ii}, R_i)$, (kompleksitasoon piirretty kiekko, jonka keskipiste on pisteessä a_{ii} , ja jonka säde on R_i). Totea lauseen väite kokeellisesti, kun $\mathbf{A} = 10 \cdot \text{randn}(12) + 5 \cdot \text{randn}(12) \cdot i$;

Vihje: Ympyrän, jonka keskipiste on (x, y) ja säde r saa MATLABissa piirrettyä helposti seuraavasti:

```
x = 0.4; y = -0.34
t = 0:0.02:2*pi;
plot(x+cos(t), y+sin(t));
hold on
%Yksittäinen piste piirretään seuraavasti
plot(x,y,'r.');
```

26. mLi050.tex

Gram-Schmidtin menetelmä vektorijoukon $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n\}$ ortonormalisoimiseksi toimii seuraavasti:

- Ortogonalisoidaan:

$$\mathbf{u}_1 = \mathbf{v}_1$$

$$\mathbf{u}_2 = \mathbf{v}_2 - \frac{\mathbf{u}_1 \cdot \mathbf{v}_2}{\mathbf{u}_1 \cdot \mathbf{u}_1} \mathbf{u}_1$$

⋮

$$\mathbf{u}_n = \mathbf{v}_n - \sum_{k=1}^{n-1} \frac{\mathbf{u}_k \cdot \mathbf{v}_n}{\mathbf{u}_k \cdot \mathbf{u}_k} \mathbf{u}_k$$

- Normitetaan: $\mathbf{e}_i = \frac{\mathbf{u}_i}{\|\mathbf{u}_i\|}, i = 1 \dots n$

Kirjoita MATLAB-funktio $\mathbf{B} = \text{grmsch}(\mathbf{A})$ joka hakee Gram-Schmidtin menetelmällä ortonormaalin kannan matriisin \mathbf{A} sarakeavaruudelle. Testaa ortonormaalius laskemalla $\mathbf{B}' * \mathbf{B}$.

Vinkki: Laskutoimitus $\frac{\mathbf{u}_k \cdot \mathbf{v}_n}{\mathbf{u}_k \cdot \mathbf{u}_k} \mathbf{u}_k$ vastaa toimitusta $(\mathbf{u}_k^T \mathbf{v}_n) \mathbf{u}_k$. **Lisätehtävä nopeille:** Matriisin sarakeavaruuden normalisointi ei poikkea kovin paljon QR-hajotelman tekemisestä. Jos ehdit, toteuta oma algoritmisi QR-hajotelmalle.