SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

**Matija Belec**

**Dario Grubešić**

**Mario Hudinčec**

Metoda potencija

Seminarski Rad

Varaždin, 2015.

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

Matija Belec

Broj indeksa: 39912/11-R

Dario Grubešić

Broj indeksa: 39962/11-R

Mario Hudinčec

Broj indeksa:39976/11-R

Redoviti student

Smjer: Informacijski sustavi

Metoda potencija

Seminarski Rad

Mentor:

Damir Horvat, prof.

Varaždin, svibanj 2015.

Sadržaj

[Zadatak 1 2](#_Toc419993135)

[Zadatak 2 3](#_Toc419993136)

[Zadatak 3 5](#_Toc419993137)

[Zadatak 4 6](#_Toc419993138)

[Zadatak 5 22](#_Toc419993139)

[Zadatak 6 24](#_Toc419993140)

[Zadatak 7 25](#_Toc419993141)

[Zadatak 8 26](#_Toc419993142)

[Zadatak 9 27](#_Toc419993143)

[Zadatak 10 28](#_Toc419993144)

[Zadatak 11 29](#_Toc419993145)

[Zadatak 12 30](#_Toc419993146)

[Literatura 31](#_Toc419993147)

# Zadatak 1

*Opišite koje uvjete mora zadovoljavati kvadratna matrica da bi se na nju mogla primjeniti metoda potencija.*

Za rješavanje kvadratne matrice metodom potencija potrebno je razmotriti sljedeće pretpostavke:

1. Matrica ima linearno nezavisnih vektora

1. Postoji jedinstvena po modulu najveća svojstvena vrijednost

Kada su oba navedena uvjeta ispunjena imamo dominantnu svojstvenu vrijednost .

# Zadatak 2

*Opišite metodu potencija u slučaju da kvadratna matrica ima dominantnu jednostruku svojstvenu vrijednost. Obratite pozornost na metodu sa i bez skaliranja*

Pretpostavimo da nam matrica A zadovoljava uvjete koje smo naveli kao uvjete u zadatku 1, tada izaberemo vektor tako da ga možemo prikazati u bazi

ukoliko koristimo prvi uvjet iz zadatka 1 formiramo niz

.

.

te nakon toga podijelimo zadnju jednadžbu s i dobijemo

Kako nam k postaje sve veći tako izrazi postaju sve manji i manji, te teže ka nuli akko . Ovu tvrdnju možemo opravdati koristeći uvjet 2 koji smo postavili u zadatku jedan: . Za veliki vrijedi da je

Ova jednadžba će dati bribližnu vrijednost sve dok je nije ortogonalan s . Ako ta jednadžba daje približnu vrijednost, tada računanjem još jednog koraka poboljšat ćemo konačan rezultat, jer je

Kada jednadžbe pomnožimo sa bilo kojim vektorom koji nije ortogonalan s , tada imamo

te nakon što se sredi izraz dobijemo

Sama potencija (power) od A daje ime metodi a to je metoda potencija (Power Method).

# Zadatak 3

*Implementirajte opisane algoritme za metodu potencija( sa i bez skaliranja). Svaki od algoritama neka na ulazu ima kvadratnu matricu, vektor s kojim počinjemo iteraciju i neka ima unaprijed zadan broj iteracija.*

Programsko rješenje je rađeno u programskoj jeziku c++. Dodatna biblioteka koja je korištena za rad s matricama i vektorima je Armadillo. Programski je kod testiran na linux operacijskom sustavu.

Dva su načina na koji je kod moguće koristiti:

* Standardni unos preko tipkovnice
* Prosljeđivanje ulaznih podataka iz datoteke u program

Program izračunava rješenje za:

* Metodu potencija bez skaliranja
* Metoda potencija sa skaliranjem

U programsko je rješenje uključena i datoteka i datoteka“build.sh“ koja je korištena za kompajliranje kodova i generiranje rješenja(koja se mogu pronaći unutar direktorija “rjesenja“).

# Zadatak 4

*Primjenite implementirane algoritme iz zadatka 3 na sljedeće matrice:*

*, , , .*

*Svaki od oba algoritma primjenite sa početnim vektorom odnosno i s brojem iteracija .Primijenite neku već gotovu naredbu za traženje svojstvenih vektora i svojstvenih vrijednosti matrice (npr.,* ***Sage,Scilab,Maxima****) i usporedite rezultate svojih algoritama sa "pravim" rezultatima tih naredbi. Posebno komentirajte što se događa sa točnšću rezultata kada je broj iteracija sve veći i veći. Obratite pažnju na svojstvene vektore. Objasnite zašto je pametnije koristiti metodu potencija sa skaliranjem. Napravite metodu potencija sa i bez skaliranja ručno za sve tri matrice s pet iteracija.*

Metoda potencija sa skaliranjem za matricu :

Kao što piše u zadatku najprije treba zadanu matricu pomnožit s vektorom te potom skalirat vrijednost koju smo dobili i takve skalirane vrijednosti množiti dalje.

U ovom djelu ne trebamo računat svojstvenu vrijednost već ju možemo samo iščitati iz rješenja pete iteracije. Naime svojstvena vrijednost je broj koji dobijemo pomoću skaliranja matrice. U ovom slučaju 9,0947.

Tablica rješenja iteracija za matricusa skaliranjem:

|  |  |
| --- | --- |
| Broj iteracije | Vrijednost svojstvenog vektora |
| 1 | (0.545454545454545, 1) |
| 2 | (0.635416666666667, 1) |
| 3 | (0.614074914869467, 1) |
| 4 | (0.618946314603929, 1) |
| 5 | (0.617824316142881, 1) |
| 6 | (0.618082205882576, 1) |
| 7 | (0.618022902122329, 1) |
| 8 | (0.618036537995637, 1) |
| 9 | (0.618033402583806, 1) |
| 10 | (0.618034123531438, 1) |
| 11 | (0.618033957758581, 1) |
| 12 | (0.618033995875957, 1) |
| 13 | (0.618033987111347, 1) |
| 14 | (0.618033989126658, 1) |
| 15 | (0.618033988663263, 1) |
| 16 | (0.618033988769815, 1 ) |
| 17 | (0.618033988745315, 1) |
| 18 | (0.618033988750948, 1) |
| 19 | (0.618033988749653, 1) |
| 20 | (0.618033988749951, 1) |
|  | 9.09016994374948 |

Metoda potencija bez skaliranja za matricu :

Potrebno je prvo pomnožiti matricu sa vektorom te nakon toga dobivene vektore dalje množiti s matricom:

Dobiveni rezultat je svojstveni vektor nakon pete iteracije pa svojstvena vrijednost iznosi:

Tablica rješenja iteracija za matricubez skaliranja:

|  |  |
| --- | --- |
| Broj iteracije | Vrijednost svojstvenog vektora |
| 1 | (6, 11) |
| 2 | (61, 96) |
| 3 | (541, 881) |
| 4 | (4946, 7991) |
| 5 | (44901, 72676) |
| 6 | (408281, 660561) |
| 7 | (3711086, 6004771) |
| 8 | (33734941, 54584056) |
| 9 | (306655221, 496179041) |
| 10 | (2787550426, 4510350351) |
| 11 | (25339302181, 40999854236) |
| 12 | (230338573361, 372695636321) |
| 13 | (2093816754966, 3387866684731) |
| 14 | (19033150178621, 30796283883216) |
| 15 | (173014569594701, 279943454192401) |
| 16 | (1.57273184055671e+15, 2.54473357312791e+15) |
| 17 | (1.42963997061963e+16, 2.3132060641551e+16) |
| 18 | (1.29956702913951e+17, 2.10274362380287e+17) |
| 19 | (1.18132851481539e+18, 1.91142968885148e+18) |
| 20 | (1.07384769590728e+19, 1.73752207071858e+19) |
|  | 9.09016994374947 |

Metoda potencija sa skaliranjem za matricu :

Kao i u prethodnim dijelu možemo iščitat svojstvenu vrijednost. U ovom slučaju

7,16241.

Tablica rješenja iteracija za matricusa skaliranjem:

|  |  |
| --- | --- |
| Broj iteracije | Vrijednost svojstvenog vektora |
| 1 | (0.75, 1) |
| 2 | (0.724137931034483, 1) |
| 3 | (0.721153846153846, 1) |
| 4 | (0.720805369127517, 1) |
| 5 | (0.720764617691154, 1) |
| 6 | (0.720759851378931, 1) |
| 7 | (0.720759293897592, 1) |
| 8 | (0.720759228692854, 1) |
| 9 | (0.720759221066305, 1) |
| 10 | (0.72075922017428, 1) |
| 11 | (0.720759220069946, 1) |
| 12 | (0.720759220057743, 1) |
| 13 | (0.720759220056316, 1) |
| 14 | (0.720759220056149, 1) |
| 15 | (0.720759220056129, 1) |
| 16 | (0.720759220056127, 1) |
| 17 | (0.720759220056126, 1) |
| 18 | (0.720759220056126, 1) |
| 19 | (0.720759220056126, 1) |
| 20 | (0.720759220056126, 1) |
|  | 7.16227766016838 |

Metoda potencija bez skaliranja za matricu :

Dobiveni rezultat je svojstveni vektor nakon pete iteracije pa svojstvena vrijednost iznosi:

Tablica rješenja iteracija za matricubez skaliranja:

|  |  |
| --- | --- |
| Broj iteracije | Vrijednost svojstvenog vektora |
| 1 | (6, 8) |
| 2 | (42, 58) |
| 3 | (300, 416) |
| 4 | (2148, 2980) |
| 5 | (15384, 21344) |
| 6 | (110184, 152872) |
| 7 | (789168, 1094912) |
| 8 | (5652240, 7842064) |
| 9 | (40482912, 56167040) |
| 10 | (289949856, 402283936) |
| 11 | (2076701376, 2881269248) |
| 12 | (14873911872, 20636450368) |
| 13 | (106531086720, 147803987456) |
| 14 | (763005222528, 1058613197440) |
| 15 | (5464855259904, 7582081654784) |
| 16 | (39140810744064, 54304974053632) |
| 17 | (280337354393088, 388947302500352) |
| 18 | (2.00785397068032e+15, 2.78574857568102e+15) |
| 19 | (1.4380807639084e+16, 1.99523047904461e+16) |
| 20 | (1.0299933728859e+17, 1.42903946869482e+17) |
|  | 7.16227766016838 |

Metoda potencija sa skaliranjem za matricu :

Također možemo iščitat svojstvenu vrijednost. U ovom slučaju 5,00212 .

Tablica rješenja iteracija za matricusa skaliranjem:

|  |  |
| --- | --- |
| Broj iteracije | Vrijednost svojstvenog vektora |
| 1 | (1, 0.571428571428571, 0.285714285714286) |
| 2 | (1, 0.513513513513514, 0.108108108108108) |
| 3 | (1, 0.502673796791444, 0.0427807486631016) |
| 4 | (1, 0.500533617929562, 0.0170757737459979) |
| 5 | (1, 0.500106678045658, 0.00682739492212503) |
| 6 | (1, 0.500021333788454, 0.00273072492213167) |
| 7 | (1, 0.500004266684871, 0.00109227132702433) |
| 8 | (1, 0.500000853334062, 0.000436907039494007) |
| 9 | (1, 0.500000170666696, 0.000174762696492834) |
| 10 | (1, 0.500000034133334, 6.99050690527597e-05) |
| 11 | (1, 0.500000006826667, 2.79620268575541e-05) |
| 12 | (1, 0.500000001365333, 1.11848106819377e-05) |
| 13 | (1, 0.500000000273067, 4.47392426788835e-06) |
| 14 | (1, 0.500000000054613, 1.7895697067644e-06) |
| 15 | (1, 0.500000000010923, 7.15827882674486e-07) |
| 16 | (1, 0.500000000002184, 2.86331153067292e-07) |
| 17 | (1, 0.500000000000437, 1.14532461226717e-07) |
| 18 | (1, 0.500000000000087, 4.58129844906707e-08) |
| 19 | (1, 0.500000000000017, 1.8325193796267e-08) |
| 20 | (1, 0.500000000000004, 7.33007751850669e-09) |
|  | 5.00000000000001 |

Metoda potencija bez skaliranja za matricu :

Dobiveni rezultat je svojstveni vektor nakon pete iteracije pa svojstvena vrijednost iznosi:

Tablica rješenja iteracija za matricu bez skaliranja:

|  |  |
| --- | --- |
| Broj iteracije | Vrijednost svojstvenog vektora |
| 1 | (7, 4, 2) |
| 2 | (37, 19, 4) |
| 3 | (187, 94, 8) |
| 4 | (937, 469, 16) |
| 5 | (4687, 2344, 32) |
| 6 | (23437, 11719, 64) |
| 7 | (117187, 58594, 128) |
| 8 | (585937, 292969, 256) |
| 9 | (2929687, 1464844, 512) |
| 10 | (14648437, 7324219, 1024) |
| 11 | (73242187, 36621094, 2048) |
| 12 | (366210937, 183105469, 4096) |
| 13 | (1831054687, 915527344, 8192) |
| 14 | (9155273437, 4577636719, 16384) |
| 15 | (45776367187, 22888183594, 32768) |
| 16 | (228881835937, 114440917969, 65536) |
| 17 | (1144409179687, 572204589844, 131072) |
| 18 | (5722045898437, 2861022949219, 262144) |
| 19 | (28610229492187, 14305114746094, 524288) |
| 20 | (143051147460937, 71525573730469, 1048576) |
|  | 5.00000000000001 |

Također ćemo prikazati rezultate za četvrtu matricu mada za nju nemamo ručno rađen dio.

Tablica rješenja iteracija za matricusa skaliranjem:

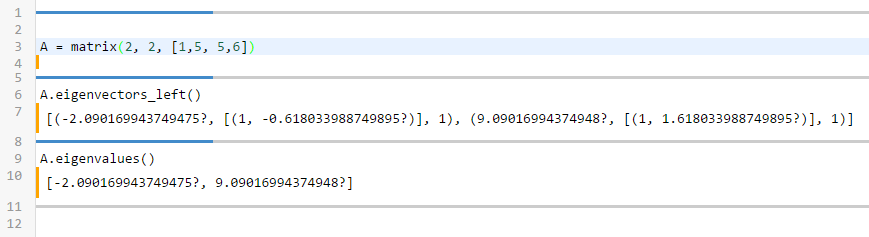
|  |  |
| --- | --- |
| Broj iteracije | Vrijednost svojstvenog vektora |
| 1 | (0.946717940668655, -0.187229634280333, 1) |
| 2 | (1, -0.284693934940124, 0.750204431702937) |
| 3 | (1, -0.280316909527727, 0.691579127515324) |
| 4 | (1, -0.278449839753962, 0.681798721602405) |
| 5 | (1, -0.27807707560487, 0.680263790837866) |
| 6 | (1, -0.278014626639038, 0.680031950137736) |
| 7 | (1, -0.278004907329185, 0.679997705209548) |
| 8 | (1, -0.278003449110532, 0.679992711151462) |
| 9 | (1, -0.278003234603477, 0.67999198819099) |
| 10 | (1, -0.278003203396325, 0.679991883979314) |
| 11 | (1, -0.278003198884976, 0.679991868995284) |
| 12 | (1, -0.278003198235217, 0.679991866843993) |
| 13 | (1, -0.278003198141837, 0.679991866535396) |
| 14 | (1, -0.278003198128434, 0.679991866491152) |
| 15 | (1, -0.278003198126512, 0.679991866484811) |
| 16 | (1, -0.278003198126236, 0.679991866483902) |
| 17 | (1, -0.278003198126197, 0.679991866483772) |
| 18 | (1, -0.278003198126191, 0.679991866483753) |
| 19 | (1, -0.27800319812619, 0.67999186648375) |
| 20 | (1, -0.27800319812619, 0.67999186648375) |
|  | 5.01457295922353 |

Tablica rješenja iteracija za matricubez skaliranja:

|  |  |
| --- | --- |
| Broj iteracije | Vrijednost svojstvenog vektora |
| 1 | (3.01577, -0.59642, 3.1855) |
| 2 | (17.1062516222, -4.8700460864, 12.8331857768) |
| 3 | (88.663402471123, -24.8538509689182, 61.3177585235193) |
| 4 | (447.301784987597, -124.551110351457, 304.969785175018) |
| 5 | (2245.22692453228, -624.346137243254, 1527.34657897358) |
| 6 | (11260.5473635145, -3130.59687101869, 7657.53198322912) |
| 7 | (56468.1008679458, -15698.4091488483, 38398.1790077444) |
| 8 | (283164.340683956, -78720.6633752497, 192549.6877231) |
| 9 | (1419948.92198327, -394750.393283068, 965553.890589054) |
| 10 | (7120437.95716166, -1979504.56167572, 4841840.02124817) |
| 11 | (35705955.9910711, -9926369.98476394, 24279759.7486318) |
| 12 | (179050121.650516, -49776506.4632481, 121752626.479778) |
| 13 | (897859898.557461, -249607923.28228, 610537428.307369) |
| 14 | (4502383968.60909, -1251677142.47552, 3061584478.47434) |
| 15 | (22577532901.1233, -6276626352.31883, 15352538738.0571) |
| 16 | (113216685972.021, -31474600781.4755, 76986425611.2361) |
| 17 | (567733332008.246, -157831681981.134, 386054048097.338) |
| 18 | (2846940214738.46, -791458484571.358, 1935896190387.66) |
| 19 | (14276189417353.5, -3968826315079.56, 9707692688181.8) |
| 20 | (71588993413014.2, -19901969119452.7, 48679933250608.5) |
|  | 5.01457295922353 |

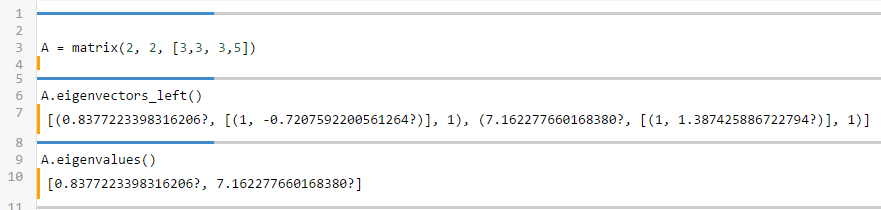
Nadalje imamo prikazane matrice u Sage-u. S naredbom *eigenvector\_left()* možemo vidjeti svojstvene vektore matrice dok s naredbom *eigenvalues()* možemo saznati svojstvenu vrijednost matrice. U nastavku su prikazi svojstvenih vrijednosti i svojstvenih vektora za matrice koje su bile zadane u zadatku 4:

Prikaz rezultata za matricu :



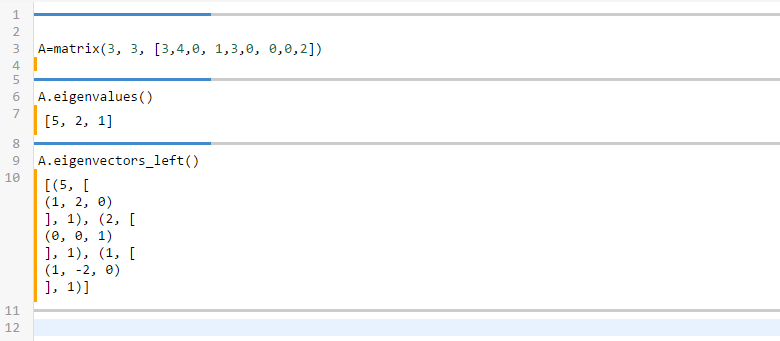
Slika :Svojstvene vrijednosti i svojstveni vektori prve matrice

Prikaz rezultata za matricu :



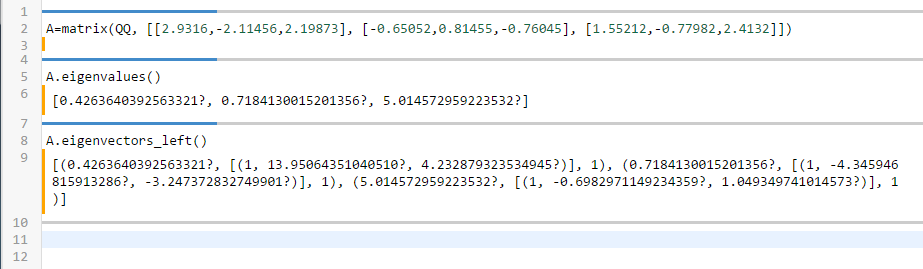
Slika :Svojstvene vrijednosti i svojstveni vektori druge matrice

Prikaz rezultata za matricu :



Slika :Svojstvene vrijednosti i svojstveni vektori treće matrice

Prikaz rezultata za matricu :



Slika :Svojstvene vrijednosti i svojstveni vektori četvrte matrice

Zapravo rješenje svojstvene vrijednosti nam odgovara podijeljenim vrijednostima vektora iz iteracije 9 i 10 iz tablice kod matrica bez skaliranja. Za svojstvenu vrijednost kod skaliranja možemo uzeti broj s kojim skaliramo kao svojstvenu vrijednost jer je približno ista kao vrijednost koje smo dobili putem algoritma.

Ukoliko imamo više iteracija približavamo se vrijednosti za svojstveni vektor koje smo dobili korištenjem programa Sage. Ukoliko pogledamo rješenje za svojstvenu vrijednost koju smo dobili putem algoritma te svojstvenu vrijednost u programu Sage možemo vidjeti da se te vrijednosti u potpunosti podudaraju na decimalu.

Kod druge matrice vrijednost svojstvenog vektora lagano pada i u nekom vremenu on se počinje ponavljat za nekoliko iteracija, što znači da se vrijednost svojstvenog vektora nakon nekoliko iteracija ne mijenja puno. To vrijedni za matricu sa skaliranim vrijednostima. Također možemo zamijetiti da je svojstvena vrijednost jednaka za matrice sa i bez skaliranja. Također i za drugu matricu se svojstvena vrijednost podudara s onom vrijednosti koju smo dobili koristeći gotovu naredbu u programu Sage.

Kod treće i četvrte matrice vidimo kao i prethodno da se pojavljuju iste vrijednosti za svojstvenu vrijednosti sa i bez skaliranja.

# Zadatak 5

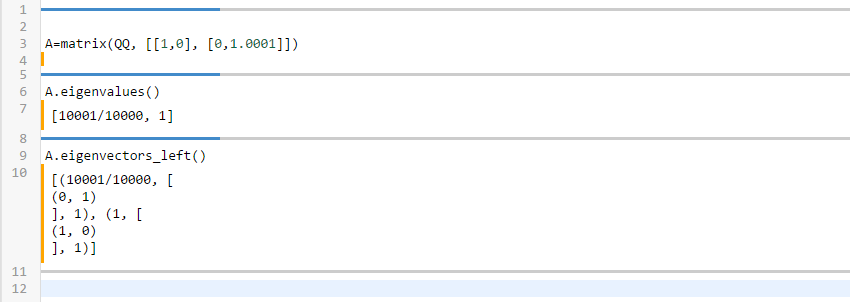
*Primijenite algoritme iz zadatka 3 na matrice*

*,*

*Primijenite neku već gotovu naredbu za traženje svojstvenih vektora i svojstvenih vrijednosti matrice (npr.,* ***Sage,Scilab, Maxima****) i usporedite rezultate svojih algoritama sa ”pravim” rezultatima tih naredbi. Objasnite zašto je kod ovih matrica potrebno puno više iteracija da bismo dobili istu točnost, nego što je to trebalo kod matrica iz zadatka 4.*

Ovaj dio je također rađen u Sage-u kao i dio u zadatku 4. prikazat ćemo svojstvene vrijednosti i svojstvene vektore matrice koje su nam zadane u zadatku 5.

Prikaz rezultata za matricu :



Slika :Svojstvene vrijednosti i svojstveni vektori prve matrice

Tablica rješenja iteracija za matricusa skaliranjem:

|  |  |
| --- | --- |
| Broj iteracije | Vrijednost svojstvenog vektora |
| 1 | (0.999900009999, 1) |
| 2 | (0.999800029996, 1) |
| 3 | (0.999700059990001, 1) |
| 4 | (0.999600099980003, 1) |
| 5 | (0.999500149965007, 1) |
| 6 | (0.999400209944012, 1) |
| 7 | (0.999300279916021, 1) |
| 8 | (0.999200359880033, 1) |
| 9 | (0.999100449835049, 1) |
| 10 | (0.999000549780071, 1) |
| 11 | (0.9989006597141, 1) |
| 12 | (0.998800779636136, 1) |
| 13 | (0.998700909545182, 1) |
| 14 | (0.998601049440238, 1) |
| 15 | (0.998501199320306, 1) |
| 16 | (0.998401359184387, 1) |
| 17 | (0.998301529031484, 1) |
| 18 | (0.998201708860598, 1) |
| 19 | (0.998101898670731, 1) |
| 20 | (0.998002098460885, 1) |
|  | 1.00005009999487 |

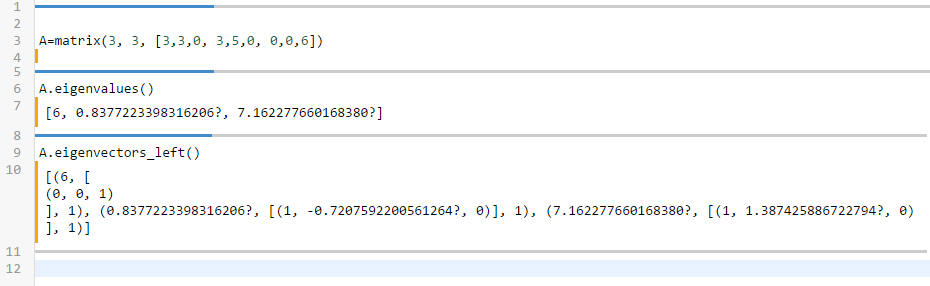
Za ovu matricu imamo situaciju da je koji smo dobili putem algoritma malo drugačiji u odnosu na rezultat u Sage-u, no ukoliko bi se ta vrijednost zaokružila dobili bi istu vrijednost i u jednom i u drugom dijelu tako da možemo zaključit da su svojstvene vrijednosti u oba rješenja jednaka. Potrebno je više iteracija da bi se dobila ista točnost jer se vrijednosti iz iteracije u iteraciju jako malo promijene. U zadatku 4 smo mogli s manje iteracija dobiti točnost jer smo zapravo s manje iteracija došli do rezultata koji je bio sličan u Sage-u.

Tablica rješenja iteracija za matricubez skaliranja:

|  |  |
| --- | --- |
| Broj iteracije | Vrijednost svojstvenog vektora |
| 1 | (1, 1.0001) |
| 2 | (1, 1.00020001) |
| 3 | (1, 1.000300030001) |
| 4 | (1, 1.000400060004) |
| 5 | (1, 1.00050010001) |
| 6 | (1, 1.00060015002) |
| 7 | (1, 1.000700210035) |
| 8 | (1, 1.00080028005601) |
| 9 | (1, 1.00090036008401) |
| 10 | (1, 1.00100045012002) |
| 11 | (1, 1.00110055016503) |
| 12 | (1, 1.00120066022005) |
| 13 | (1, 1.00130078028607) |
| 14 | (1, 1.0014009103641) |
| 15 | (1, 1.00150105045514) |
| 16 | (1, 1.00160120056018) |
| 17 | (1, 1.00170136068024) |
| 18 | (1, 1.00180153081631) |
| 19 | (1, 1.00190171096939) |
| 20 | (1, 1.00200190114048) |
|  | 1.00005009999487 |

Što se tiče dijela bez skaliranja imamo situaciju da se vrijednost vektora iz iteracije u iteraciju jako malo povećava, no taj dio nas ne zanima toliko u ovom dijelu pitanja ali smo ga ipak naveli da se vidi što se događa.

Prikaz rezultata za matricu:



Slika :Svojstvene vrijednosti i svojstveni vektori druge matrice

Tablica rješenja iteracija za matricusa skaliranjem:

|  |  |
| --- | --- |
| Broj iteracije | Vrijednost svojstvenog vektora |
| 1 | (0.75, 1, 0.75) |
| 2 | (0.724137931034483, 1, 0.620689655172414) |
| 3 | (0.721153846153846, 1, 0.519230769230769) |
| 4 | (0.720805369127517, 1, 0.434899328859061) |
| 5 | (0.720764617691154, 1, 0.36431784107946) |
| 6 | (0.720759851378931, 1, 0.305196504265006) |
| 7 | (0.720759293897592, 1, 0.255669862052841) |
| 8 | (0.720759228692854, 1, 0.21418034843888) |
| 9 | (0.720759221066305, 1, 0.17942366199109) |
| 10 | (0.72075922017428, 1, 0.150307209880735) |
| 11 | (0.720759220069946, 1, 0.12591570754862) |
| 12 | (0.720759220057743, 1, 0.105482401148573) |
| 13 | (0.720759220056316, 1, 0.0883649639011807) |
| 14 | (0.720759220056149, 1, 0.0740253043184279) |
| 15 | (0.720759220056129, 1, 0.0620126511403807) |
| 16 | (0.720759220056127, 1, 0.0519493832124816) |
| 17 | (0.720759220056126, 1, 0.0435191588575696) |
| 18 | (0.720759220056126, 1, 0.0364569715856672) |
| 19 | (0.720759220056126, 1, 0.0305408195399201) |
| 20 | (0.720759220056126, 1, 0.0255847268053571) |
|  | 7.16177718162561 |

Kod ove matrice imamo situaciju da nam vrijednosti kroz svaku iteraciju padaju tj. teže k nuli. Što se tiče rezultata u Sage-u i rezultata algoritma svojstvena vrijednost se razlikuje no to je neprimjetno malo, no također imamo situaciju da se zadnje tri iteracije podudaraju, te se također podudaraju s jednom vrijednosti dobivenom putem Sage-a. Vrijednosti bez skaliranja nam kao i kod prethodne matrice nisu toliko zanimljive.

# Zadatak 6

*Primijenite algoritme iz zadatka 3 na matricu*

*tako da u prvom slučaju uzmete početni vektor a u drugom slučaju vektor Proučite dobivene rezultate na prvih 20 iteracija. Što možete zaključiti?*

Tablica rješenja iteracija za vektor  sa skaliranjem:

|  |  |
| --- | --- |
| Broj iteracije | Vrijednost svojstvenog vektora |
| 1 | (1, 0.00510204081632648, 0.75765306122449) |
| 2 | (0.852604100343818, 0.490175239252025, 1) |
| 3 | (1, 0.255393612666684, 0.94342228796681) |
| 4 | (0.952470815576676, 0.38440060531869, 1) |
| 5 | (1, 0.327281685851616, 0.995077235549075) |
| 6 | (0.97854110629628, 0.356828205061264, 1) |
| 7 | (0.991631780746411, 0.342983299067064, 1) |
| 8 | (0.985093404940615, 0.349898389549621, 1) |
| 9 | (0.988347005257361, 0.346457329987834, 1) |
| 10 | (0.986724957325308, 0.348172833738413, 1) |
| 11 | (0.98753286608862, 0.347318376551174, 1) |
| 12 | (0.987130278162655, 0.347744159959472, 1) |
| 13 | (0.987330845241425, 0.347532037019654, 1) |
| 14 | (0.987230912421399, 0.347637727563178, 1) |
| 15 | (0.987280701252508, 0.347585070101695, 1) |
| 16 | (0.987255894607728, 0.347611306004666, 1) |
| 17 | (0.987268254024918, 0.34759823448802, 1) |
| 18 | (0.987262096147888, 0.347604747156931, 1) |
| 19 | (0.987265164198428, 0.347601502337772, 1) |
| 20 | (0.987263635595224, 0.347603119012888, 1) |
|  | 5.7851533758566 |

Tablica rješenja iteracija za vektor  sa skaliranjem:

|  |  |
| --- | --- |
| Broj iteracije | Vrijednost svojstvenog vektora |
| 1 | (-1, 0.225728155339806, -0.604368932038835) |
| 2 | (-0.739959763421017, -0.608969025901667, -1) |
| 3 | (-1, -0.173591064095121, -0.884649701547426) |
| 4 | (-0.921289566757739, -0.417378289277418, -1) |
| 5 | (-1, -0.304962650322255, -0.979040553833171) |
| 6 | (-0.970574613042596, -0.36525369529386, -1) |
| 7 | (-0.995658604532214, -0.338724466037684, -1) |
| 8 | (-0.983101503798239, -0.35200505595139, -1) |
| 9 | (-0.989342997910985, -0.34540395227989, -1) |
| 10 | (-0.986229607138213, -0.348696723992836, -1) |
| 11 | (-0.987779885323614, -0.347057125070424, -1) |
| 12 | (-0.98700725972222, -0.347874266223156, -1) |
| 13 | (-0.987392150560071, -0.347467199537364, -1) |
| 14 | (-0.987200371484247, -0.347670028145173, -1) |
| 15 | (-0.987295918572653, -0.347568976021325, -1) |
| 16 | (-0.987248313052272, -0.347619324378577, -1) |
| 17 | (-0.987272031453233, -0.347594239419617, -1) |
| 18 | (-0.987260214123314, -0.347606737616123, -1) |
| 19 | (-0.987266101887298, -0.347600510623075, -1) |
| 20 | (-0.987263168408994, -0.347603613116492, -1) |
|  | 5.78515304542514 |

Kao što se može vidjeti iz tablice vrijednost svojstvenih vektora najprije je blizu jedinice, zatim počinje padati vrijednost te nakon toga kroz par iteracija vrijednost mu stalno raste i pada. Što se pak vrijednosti za drugog vektora tiče dešava se sve isto kao i kod vektora jedina je razlika što vektor ide u negativnom smjeru dok vektor ide u pozitivan smjer. Vrijednosti bez skaliranja nam u ovom dijelu nisu zanimljivi za promatrati, pa ih nismo ni naveli no ukoliko vas zanimaju rezultati potražite ih u datoteci projektnog zadatka.

# Zadatak 7

*Navedite slučajeve kada metoda potencija neće uspješno obaviti svoj posao, tj. u kojim slučajevima će taj algoritam propasti.*

Metoda potencija ima 3 loše osobine zbog kojih i dolazi do toga da ne može obaviti svoj posao, a to su:

1. Ovakva analiza neće raditi ako je najveća po vrijednosti svojstvena vrijednost kompleksna tj. ako ima svoj konjugirano-kompleksni par

2. Ako brzina konvergencije ovisi o omjeru λ2/λ1 i sporija je što je taj omjer veći

3. Metoda konvergira uvijek samo prema najvećoj svojstvenoj vrijednosti po apsolutnoj vrijednosti.

Možemo uočiti da bi korištenjem realnog pomaka σ za matricu A mogli dobiti samo još najmanju svojstvenu vrijednost po apsolutnoj vrijednosti, no ne možemo dobiti i one između vrijednosti.

Metoda potencija strogo se pridržava određenih pretpostavki, ukoliko se ona ne pridržava tih pretpostavki zakazat će. Slučajevi kada ona zakaže su sljedeći:

1. Realna kvadratna matrica n\*n imat će n linearno nezavisnih svojstvenih vektora ako i samo ako se matrica može dijagonalizirat, no samim pogledom na matricu ne možemo znati da li se ona može dijagonalizirati.

2. Realna matrica n\*n za svojstvene vrijednosti ima kompleksne brojeve koji dolaze u konjugirano-kompleksnim parovima (|λ1| = |λ2|), te tada ne postoji dominantna po modulu najveća svojstvena vrijednost, a samo kod simetrične matrice svojstvene vrijednosti su realne i kod njih se nerijetko pojavljuju vrijednosti koje su po amplitudi jednake, ali se razlikuju u predznaku, to znači da će algoritam zakazati u slučaju kada imamo različite svojstvene vrijednosti tj. kada im se predznaci razlikuju, a po amplitudi su jednake.

3. Ako za danu matricu postoji |λ1|≈|λ2|, tada metoda sporo konvergira prema rješenju

# Zadatak 8

*Primijenite algoritme iz zadatka 3 na matricu*

*tako da za početne vektore redom uzmete*

*Proučite dobivene rezultate na prvih 20 iteracija i usporedite ih s ”pravim” rezultatima. Što se događa u ovom slučaju? Objasnite zašto u ovom slučaju ova metoda nije dobra.*

# Zadatak 9

*Objasnite nakon koliko iteracija treba zaustaviti metodu potencija da bismo dobili traženu točnost. Napišite novi algoritam koji će se zaustaviti nakon što se postigne unaprijed zadana točnost svojstvene vrijednosti. Primijenite taj algoritam na matrice iz zadatka 4.*

Metodu potencije ponavljamo toliko puta dok ne postignemo određenu točnost rješenja koju bismo mogli najbolje procijeniti kad bismo znali točnu svojstvenu vrijednost. U realnom svijetu nikad ne možemo saznati točnu svojstvenu vrijednost, zbog toga moramo iterirati metode potencija do neke određene točnosti. Kako bismo procijenili pogrešku, trebamo računati pomoću sljedećeg izraza:

Zamislimo da imamo simetričnu matricu A sa dominantnom svojstvenom vrijednosti , tada dobivamo sljedeće formule:

,

a ocjena pogreške je:

**-->** za nesimetričnu matricu.

# Zadatak 10

*Objasnite kako nastaviti dalje kada se pronađe dominantna svojstvena vrijednost. Kako pronaći sljedeću dominantnu svojstvenu vrijednost? Radi jednostavnosti ograničite se samo na simetrične matrice. Implementirajte opisani algoritam. Primijenite taj algoritam na matrice*

*, ,*

*Usporedite dobivene rezultate s ”pravim” vrijednostima.*

Postoji nekoliko koraka koje trebamo napraviti kako bismo pronašli dominantnu svojstvenu vrijednost. Prvi korak kojeg trebamo napraviti je pronaći najveću svojstvenu vrijednost i vektor početne matrice koristeći metodu potencija. Nakon što ih pronađemo, potrebno je pomnožiti najveći svojstveni vektor sa transponiranim svojstvenim vektorom te najvećom svojstvenom vrijednosti: , gdje je najveća svojstvena vrijednost, Y je najveći svojstveni vektor, a je transponirani svojstveni vektor. Nakon što smo i to napravili, moramo izračunati novu matricu po formuli: , te na kraju koristeći metodu potencija ponovno izračunati svojstvenu vrijednost novodobivene matrice koja je ujedno i sljedeća dominantna svojstvena vrijednost početne matrice.

# Zadatak 11

*Da li se opisana metoda potencija može primijeniti na realne matrice s kompleksnim svojstvenim vrijednostima? Objasnite. Primijenite tu metodu na matricu*

*tako da za početne vektore uzmete i Usporedite dobivene rezultate s ”pravim” rezultatima. Što se događa? Objasnite!*

# Zadatak 12

*Opišite metodu potencija u slučaju da kvadratna matrica ima dominantnu višestruku svojstvenu vrijednost.*

# Literatura

[1] John H. Mathews; Kutris D. Fink; - Numerical methods using matlab- fourth edition

[2] Lab15- Power Method and Dominant Eigenvalues

[3] M. Jelić; D.Panić – Metoda potencija za rješavanje najveće svojstvene vrijednosti; 2009

[4] Numerical calculator od eigenvalues;- Power method

[5] Belac,Butković,Crnac- Metoda potencija- rad od prošle godine korišten za usporedbu rezultata.

[6] <http://mathfaculty.fullerton.edu/mathews/n2003/PowerMethodMod.html>

[7] <http://zoro.ee.ncku.edu.tw/na/res/09-power_method.pdf>

[8] <http://www.mathos.unios.hr/~scitowsk/MP/Power.pdf>

[9] <http://web.math.pmf.unizg.hr/~nela/nmfmpredavanja/nmfm_svojstvene_vrijednosti.pdf>

[10] <http://student.fizika.org/~hmikulec/num/metoda_potenciranja.htm>

[11] <http://ergodic.ugr.es/cphys/LECCIONES/FORTRAN/power_method.pdf>

Korišteni alati :

* Sage- Datoteke vezane uz Sage nalaze se u mapi seminarske dokumentacije pod nazivom Sage
* C++
* Microsoft Office Word