Eetu Alanen

PSD-hajotelman ja -asteen laskeminen ja niiden sovellukset kvantti-informaatiossa

Tieto- ja ohjelmistotekniikan kandidaatintutkielma

23. syyskuuta 2024

Jyväskylän yliopisto

Informaatioteknologian tiedekunta

Tekijä: Eetu Alanen

Yhteystiedot: `eetu.o.a.alanen@student.jyu.fi`

Ohjaaja: Oskari Kerppo

Työn nimi: PSD-hajotelman ja -asteen laskeminen ja niiden sovellukset kvantti-informaatiossa

Title in English: Calculating PSD factroizations and PSD rank and how to use them in quantum information

Työ: Kandidaatintutkielma

Opintosuunta: Tieto- ja ohjelmistotekniikka

Sivumäärä: 29+0

Tiivistelmä: Tutkielmassa esitellään PSD-hajotelma, sekä PSD-aste, ja esitellään käytännönkohteita, johon erityisesti PSD-astetta käytetään. Tutkielmassa esitellään myös miten PSD-asteelle voi laskea ylä- ja alarajoja, sekä toteutustapoja kyseisille rajoille. Myös PSD-hajotelman arviontia käsitellään tutkielmassa.

Avainsanat: PSD-hajotelma, PSD-aste, kvantti-informaatio, yläraja, alaraja

Abstract: In this bachelor's thesis we introduce the consepts of PSD factorization and PSD rank, and show applications for especially PSD rank. We also show how we can set lower and upper bounds for PSD rank and how we can approximate PSD factorizations. It is also shown how we can implement the upper and lower bounds, and the approximation for PSD factorizations

Keywords: PSD-facotrization, PSD-rank, quantum information, upper bound, lower bound

Taulukot

Sisällys

1 Johdanto

PSD-hajotelmat ja PSD-aste ovat epänegatiivisten matriisien ominaisuus. Tässä tutkielmassa keskitytään PSD-asteeseen, jota käytetään mm. kvantti-informaatiossa, sekä kombinatorisessa optimoinnissa. PSD-hajotelmien muodostaminen on NP-täydellinen ongelma ja PSD-asteen laskeminen $\exists\mathbb{R}$-täydellinen onglema. Koska emme pysty muodostamaan tarkkaa PSD-hajotelmaa, joudumme arvioimaan sitä numeerisilla menetelmillä. Samoin emme voi laskea matriisin PSD-astetta, mutta voimme asettaa sille ylä- ja alarajoja käyttäen erilaisia numeerisia menetelmiä. Hyvällä tuurilla, voimme saada PSD-asteen ratkaistua käyttämällä näitä ylä- ja alarajoja.

Tässä tutkielmassa tulen määrittelemään PSD-asteen ja PSD-hajotelman, sekä tulen esittelemään, mihin ne ovat hyödyllisiä. Tulen myös esittelemään eri ylä- ja alarajoja PSD-asteelle, sekä sen, miten ne ovat toteutettu ja miten tehokkaita ne ovat PSD-asteen ratkaisemiseen. Tulen myös esittelemään algoritmin, joka arvioi PSD-hajotelmaa, sekä arvioin sen tehokkuutta.

2 PSD-aste

Matriisihajotelmat (engl. *matrix factorization*) ovat tärkeä sovelletun matematiikan osa-alue (Fawzi ym. 2015). Fawzi ym. (2015) esittävät julkaisussaan esimerkin yksinkertaisesta matriisihajotelmasta: olkoon M matriisi kooltaan $p \times q$ ja A $p \times k$, sekä B $k \times q$ matriiseja, joiden välinen matriisitulo muodostavat matriisin M: $M = AB$. Pienin mahdollinen k, jolla voidaan muodostaa matriisit A ja B on matriisin M aste (engl. *rank*) (Fawzi ym. 2015). Matriisin astetta merkitään $\texttt{rank}(M) = k$. Matriisihajotelmia käytetään eri informaatioteknologian osa-alueilla, kuten koneoppimisessa ja kvantti-informaation parissa (Fawzi ym. 2015). Hajotelmille voidaan asettaa myös erilaisia ehtoja (Fawzi ym. 2015). Esimerkiksi epänegatiivinen hajotelma (engl. *nonnegative factorization*) muodostuu, kun matriisit M, A ja B sisältävät ainoastaan epänegatiivisia alkioita ja toteuttavat ehdon $M = AB$ (Gillis ja Glineur 2012). Epänegatiivinen aste (engl. *nonnegative rank*) on pienin kokonaisluku k , jolla voidaan muodostaa epänegatiivinen hajotelma (Gillis ja Glineur 2012).

2.1 PSD matriisihajotelma ja -aste

Positiivisemidefiniitti (PSD) matriisihajotelma (engl. *PSD factorization*), on hajotelma, jossa epänegatiivinen matriisi M dimensioiltaan $p \times q$ hajotetaan $k \times k$ PSD matriiseihin $\{A_1, ..., A_p\}$ ja $\{B_1, ..., B_q\}$ siten, että $M_{ij} = \mathrm{Tr}(A_i B_i)$, missä k on pienin mahdollinen kokonaisluku, joka muodostaa kyseisen matriisihajotelman ja $\mathrm{Tr}(A_i B_j)$ on matriisien A_i ja B_j välisen tulon muodostaman matriisin jälki (engl. *trace*) (Lee, Wei ja Wolf 2017). Pienin kokonaisluku k, jolla voidaan muodostaa PSD matriisihajotelma, kutsutaan matriisin M PSD-asteeksi (engl. *PSD-rank*), jota merkitään seuraavasti: $\texttt{rank}_{\text{psd}}(M) = k$ (Lee, Wei ja Wolf 2017).

2.1.1 PSD-asteen ja matriisin asteen välinen yhteys

Fawzi ym. (2015) julkaisussa kerrotaan PSD-asteen ja matriisin asteen välisestä yhteydestä: jos $\texttt{rank}(M) \leq 2$, niin $\texttt{rank}_{\text{psd}}(M) = \texttt{rank}(M)$, toisaalta, jos $\texttt{rank}(M) > 2$, niin $\texttt{rank}_{\text{psd}}(M) \geq 2$. Erityisesti, jos matriisi M sisältää ainoastaan alkioita arvoiltaan 0 tai 1, pätee $\texttt{rank}_{\text{psd}}(M) \leq \texttt{rank}(M)$ (Fawzi ym. 2015). Matriisin asteen avulla voidaan myös laskea PSD-asteelle eräs alaraja (Lee, Wei ja Wolf 2017), mutta palaamme siihen myöhemmin tämän tutkielman aikana. Epänegatiiviselle asteelle pätee aina $\texttt{rank}_{\text{psd}}(M) \leq \texttt{rank}_{+}(M)$, missä $\texttt{rank}_{+}(M)$ on epänegatiivinen aste (Lee, Wei ja Wolf 2017).

2.1.2 Esimerkki PSD-hajotelmasta

Fawzi ym. (2015) julkaisussa annetaan esimerkkejä PSD-hajotelmista, joista yksi on seuraava: Olkoon M matriisi muodoltaan 3×3 seuraavanlainen:

$$M = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix} \tag{2.1}$$

Matriisi M täyttää ehdot $\texttt{rank}(M) = \texttt{rank}_{+}(M) = 3$, joten PSD-aste on korkeintaan 3. Matriisista voidaan luoda PSD-hajotelma seuraavia matriiseja käyttäen:

$$A_1 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} A_2 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} A_3 = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

$$B_1 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} B_2 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} B_3 = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

Matriisit A_i ja B_j ovat kaikki positiivi semidefiniittejä, sekä mudoltaan 2×2 ja täyttävät ehdon $M_{ij} = \mathrm{Tr}(A_i B_j)$ kaikilla $i = 1,..,3$ ja $j = 1,..,3$. Täten matriisin M PSD-aste on korkeintaan 2, ja kuten myöhemmin näemme, matriisin PSD-aste on 2.

2.2 PSD-asteen laskeminen

PSD-hajotelman muodostaminen on NP-täydellinen ongelma (Vandaele, Glineur ja Gillis 2018). PSD-asteen laskeminen on puolestaan $\exists\mathbb{R}$ -täydellinen onglema (Shitov 2017). PSD-asteelle on kuitenkin olemassa eräänlaisia ylä- ja alarajoja joiden avulla voidaan asettaa rajat PSD-asteelle ja jos hyvä tuuri käy, saadaan jopa ratkaistua PSD-aste. PSD-hajotelman arvioimiseen on myös olemassa menetelmiä. Tulemme käsittelemään molempia myöhemmin tutkielmassa.

2.3 Reaalinen PSD-aste

PSD-hajotelman muodostavat matriisit voivat olla reaalisia ja symmetrisiä, tai kompleksisia ja hermiittisiä (Lee, Wei ja Wolf 2017). Jos asetamme rajoitteen, että mariisit saavat olla vain reaalisia ja symmetrisiä, kutsumme sitä reaaliseksi PSD-asteeksi ja merkitsemme sitä seuraavasti $\mathtt{rank}^{\mathbb{R}}_{\mathrm{psd}}(M)$ (Lee, Wei ja Wolf 2017). Reaaliselle PSD-asteelle pätee seuraava tulos:

$$\mathtt{rank}^{\mathbb{R}}_{\mathrm{psd}}(M) \leq \mathtt{rank}_{\mathrm{psd}}(M)$$

(Lee, Wei ja Wolf 2017)

Tutkielmassa tullaan olettamaan, että PSD-asteelle ei aseteta reaalisen PSD-asteen rajoitteita, ellei toisin mainita.

3 PSD-asteen käytännönkohteita

PSD-asteelle on olemassa käytännönkohteita eri aloilla, kuten kombinatorisessa optimoinnissa, kvantti-informaatiossa, sekä probabilistisessa mallintamisessa (Lahat ym. 2021). Seuraavaksi tulen käsittelemään PSD-astetta kvantti-informaatiossa.

3.1 Kvantti-informaation peruskäsitteitä

PSD-astetta käytetään työkaluna kvantti-informaation eri osa-alueilla. Jotta voimme kästitellä aihetta, on hyvä esitellä hieman peruskäsitteitä. Klassisessa informaatiossa *bitti* voi olla kahdessa eri tilassa, 0 tai 1 (Nielsen ja Chuang 2010). Kvantti-informaatiossa *qubitti* sen sijaan voi olla tilassa $|0\rangle$ tai $|1\rangle$ tai niiden superpositiossa. Superpositio on tila, joka on tilojen $|0\rangle$ ja $|1\rangle$ lineaarikombinaatio. *Kvanttitila*, on tila, jossa qubitti on. *Kvanttisysteemin* tila voidaan ilmaista *tiheysmatriisin* avulla. Tiheysmatriisi ρ määritellään seuraavasti:

$$\rho \equiv \sum_i p_i |\psi_i\rangle\langle\psi_i|, \tag{3.1}$$

missä $|\psi_i\rangle$ on eräs kvanttitila ja p_i kyseisen tilan todennäköisyys. Tiheysmatriiseilla on seuraava ominaisuus:

$$Tr(\rho) = 1 \tag{3.2}$$

Kvanttikommunikaatio on kvanttitilojen siirtämistä paikasta toiseen. Kvanttikommunikaatiossa dataa ei siirretä bittien avulla, vaan qubittien avulla. Olkoon $\{E_m\}$ joukko matriiseja, joille pätee $\sum_{i=1}^{m} E_i = I$ ja $|\psi\rangle$ jokin kvanttitila. Tällöin todennäköisyys, että kvanttitilaa mitatessa, tulos on m on $\langle\psi| E_m |\psi\rangle$. Operaattoreita E_m kutsutaan *POVM* (engl. *Positive Operator-Valued Measure*) elementeiksi ja joukko $\{E_m\}$ on itse POVM. Joukon matriisit ovat kaikki positiivisemidefiniittejä (Nielsen ja Chuang 2010).

3.2 PSD-aste kvantti-informaatiossa

Olkoon M $m \times n$ epänegatiivinen matriisi, jossa jokainen sarake on todennäköisyysjakauma. Oletetaan, että $\texttt{rank}_{\text{psd}}(M) = r$. Silloin on olemassa $r \times r$ positiivisemidefiniitti matriisit $\{E_m\}$ ja $\{F_n\}$ siten, että $M_{ij} = Tr(E_i F_j)$ (Lee, Wei ja Wolf 2017). Julkaisussa (Fiorini ym. 2012) kerrotaan, kuinka koon r matriisn M PSD-hajotelmasta voidaan konstruoida kvanttikommunikaatio protokolla, joka lähettää $(r+1)$-dimensioisia viestejä. Myöhemmin protokollaa on kehitetty lähettämään r-dimensoisia viestejä, jolloin matriisit $\{E_m\}$ ja $\{F_n\}$ täyttävät seuraavat ehdot:

$$\sum_{i=1}^{m} E_i = I \tag{3.3}$$

$$Tr(F_j) = 1 \tag{3.4}$$

(Lee, Wei ja Wolf 2017)

Toisin sanoen, matriisit $\{E_m\}$ muodostavat $POVM$:n ja matriiseja $\{F_n\}$ voidaan ajatella kvanttitiloina (Lee, Wei ja Wolf 2017).

Lisäksi julkaisussa Jain ym. (2013) esitetään kvantti-informaatioon liittyvä tulkinta PSD-asteelle, jossa $\log \texttt{rank}_{\text{psd}}(M)$ antaa mittauksen kahden satunnaismuuttujan (X,Y) väliselle korrelaatiolle.

4 Eri ylä- ja alarajoja PSD-asteelle

Kuten osiossa 2.2 todettiin, PSD-asteelle on olemassa erilaisia ylä- ja alarajoja. Koska PSD-asteen ratkaiseminen on $\exists\mathbb{R}$-täydellinen ongelma (Polyak 2007), voi olla mielekkäämpää laskea ylä- ja alarajoja PSD-asteelle (Lee, Wei ja Wolf 2017). Seuraavissa esimerkeissä tullaan puhumaan pystyrivistokastisista matriisesita. Matriisi on vaakarivistokastinen, jos matriisin vaakarivien alkiot ovat epänegatiivisia ja niiden summa on 1. Matriisi on pystyrivistokastinen, jos matriisin pystyrivien alkiot ovat epänegatiivisia ja niiden summa on 1. Vaikka tutkielmassa puhutaan pystyrivistokastisista matriiseista, toimivat esiteltävät menetelmät myös vaakarivistokastisilla matriiseilla.

4.1 Alarajoja

4.1.1 $B_1(M)$ ja $B_1^{'}(M)$

Olkoon M Epänegatiivinen matriisi. Tällöin pätee:

$$B_1(M) = \sqrt{\texttt{rank}(M)} \leq \texttt{rank}_{\text{psd}}(M) \tag{4.1}$$

$$B_1^{'}(M) = \frac{1}{2}(\sqrt{1+8\texttt{rank}(M)}-1) \leq \texttt{rank}_{\text{psd}}^{\mathbb{R}}(M) \tag{4.2}$$

(Lee, Wei ja Wolf 2017)

Esimerkissä 2.1.2 todettiin, että matriisin 2.1 PSD-aste on korkeintaan 2. Nyt osoitamme, että matriisin PSD-aste on 2. Tiedetään, että matriisin 2.1 aste on 3. Sijoittamalla kaavaan 4.1 matriisin asteen, saamme tulokseksi:

$$\sqrt{3} \leq \texttt{rank}_{\text{psd}}(M)$$

Koska PSD-asteen tulee olla kokonaisluku, pyöristämme ylöspäin: $\lceil\sqrt{3}\rceil = 2$ Nyt huomataan, että:

$$2 \leq \texttt{rank}_{\text{psd}}(M) \leq 2$$

Eli PSD-aste on 2.

4.1.2 $B_2(M)$

Olkoon M kaksiulotteinen todennäköisyysjakauma kahden pelaajan Alice (A) ja Bob (B) välillä, jolloin:

$$B_2(M) = 2^{H(A:B)} \leq \texttt{rank}_{\text{psd}}(M), \tag{4.3}$$

missä $H(A:B)$ on pelaajien välinen yhteinen informaatio (Lee, Wei ja Wolf 2017)

4.1.3 $B_3(M)$ ja $B_3^{'}(M)$

Olkoon M pystyrivistokastinen matriisi, tällöin:

$$B_3(M) = \max_q \frac{1}{\sum_{i,j} q_i q_j F(M_i, M_j)^2} \leq \texttt{rank}_{\text{psd}}(M), \tag{4.4}$$

missä M_i on i:s pystyrivi ja kaavaa maksimoidaan todennäköisyysjakauman $q = \{q_j\}$ yli (Lee, Wei ja Wolf 2017). Kaavassa esiintyvä $F(M_i, M_j)^2$ on matriisin pystyrivien i ja j välinen fideliteetti (engl. $fidelity$) (Lee, Wei ja Wolf 2017), joka lasketaan seuraavasti:

$$\sum_k \sqrt{M_{ki} M_{kj}} \tag{4.5}$$

Voimme muokata $B_3(M)$ siten, että kaava pätee epänegatiivisille matriiseille, jotka eivät ole stokastisia (Lee, Wei ja Wolf 2017):

$$B_3^{'}(M) = \max_{q,D} \frac{1}{\sum_{i,j} q_i q_j F((DM)_i, (DM)_j)^2} \leq \texttt{rank}_{\text{psd}}(M), \tag{4.6}$$

missä $q = \{q_j\}$ on todennäköisyysjakauma, D on epänegatiivinen diagonaalimatriisi ja $(DM)_i$ on matriisin DM i:s pystyrivi normalisoituna (Lee, Wei ja Wolf 2017).

4.1.4 $B_4(M)$ ja $B_4^{'}(M)$

Pystyrivistokastiselle matriisille M pätee:

$$B_4(M) = \sum_i \max_j M_{ij} \leq \texttt{rank}_{\text{psd}}(M) \tag{4.7}$$

(Lee, Wei ja Wolf 2017)

Myös tässä tapauksessa voimme skaalata matriisia M matriisilla D, jolloin kaava pätee myös epänegatiivisille, ei-stokastiselle (Lee, Wei ja Wolf 2017):

$$B_4^{'}(M) = \max_D \sum_i \max_j ((DM)_j)_i \leq \texttt{rank}_{\text{psd}}(M) \tag{4.8}$$

missä D on epänegatiivinen diagonaalimatriisi ja $(DM)_j$ on todennköisyysjakauma, joka saadaan, kun normalisoidaan matriisin j:s pystyrivi, ja $((DM)_j)_i$ on jakauman $(DM)_j$ i:s alkio (Lee, Wei ja Wolf 2017).

4.1.5 $B_5(M)$ ja $B_5^{'}(M)$

Pystyrivistokastiselle matriisille M pätee:

$$B_5(M) = \sum_i \max_{q^{(i)}} \frac{\sum_k q_k^{(i)} M_{ik})}{\sqrt{\sum_{s,t} q_s^{(i)} q_t^{(i)} F(M_s, M_t)^2}} \leq \texttt{rank}_{\text{psd}}(M), \tag{4.9}$$

missä M_s on matriisin M pystyrivi ja jokaista i:tä kohden $q^{(i)} = \{q_k^{(i)}\}$ on todennäköisyysjakauma (Lee, Wei ja Wolf 2017). Kuten aiemminkin, on mahdollista skaalata matriisilla D, jolloin kaava pätee epänegatiivisille, ei-stokastisille matriiseille (Lee, Wei ja Wolf 2017):

$$B_5'(M) = \max_D \sum_i \max_{q^{(i)}} \frac{\sum_k q_k^{(i)}((DM)_k)_i}{\sqrt{\sum_{s,t} q_s^{(i)} q_t^{(i)} F((DM)_s, (DM)_t)^2}} \leq \mathtt{rank}_{\mathrm{psd}}(M), \tag{4.10}$$

4.2 Ylärajoja

4.2.1 $\min\{p,q\}$

Olkoon epänegatiivinen matriisi M dimensioiltaan $p \times q$, tällöin pätee $\min\{p,q\} \leq \mathtt{rank}_{\mathrm{psd}}(M)$ (Fawzi ym. 2015).

4.2.2 $\mathtt{rank}_+(M)$

Kuten osiossa 2.1.1 todettiin, epänegatiivinen aste on eräs yläraja PSD-asteelle (Lee, Wei ja Wolf 2017).

4.2.3 $\mathtt{rank}_{\sqrt{}}(M)$

Hadamardin neliöjuuri on operaatio, jossa jokainen matriisin alkio vaihdetaan sen neliöjuurella joko miinus tai plus merkkisenä (Fawzi ym. 2015). Epänegatiivisen matriisin M neliöjuuri-aste, merkitään $\mathtt{rank}_{\sqrt{}}(M)$ (engl. *square root rank*), on matriisin M pienin aste, kun siitä otetaan Hadamardin neliöjuuri $\sqrt{M}$ (Fawzi ym. 2015). Esimerkiksi, kun matriisiin, jonka aste on 3

$$M = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 4 \\ 1 & 1 & 1 \end{bmatrix}$$

operoidaan Hadamardin neliöjuurella, saadaan matriisi

$$\sqrt{M} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & -2 \\ 1 & 1 & -1 \end{bmatrix},$$

jonka aste on 2 (Fawzi ym. 2015). Täten $\mathtt{rank}_{\mathrm{psd}}(M) \leq \mathtt{rank}_{\sqrt{}}(M) = 2$ (Fawzi ym. 2015).

5 Toteutuksia PSD-rajoille

Osa aiemmin esitellyistä ylä- ja alarajoista on toteutettavissa numeerisella ohjelmoinnilla. Olen toteuttanut osan niistä Python ohjelmointikielellä ja tulen vertailemaan niitä tässä osiossa.

5.1 Toteutetut alarajat

5.1.1 $B_1^{'}$ ja B_4

Kaavojen $B_1^{'}$ ja B_4 toteutukset ovat molemmat hyvin yksinkertaisia. $B_1^{'}$ toteutuksessa lasketaan kysessä olevan matriisin aste ja sijoitetaan se kaavaan, jonka jälkeen palautetaan kaavan antama alaraja reaaliselle PSD-asteelle. Samoin B_4 toteutuksessa otetaan operoitavan matriisin jokaisen rivin suurin alkio ja palautetaan niiden summa.

5.1.2 B_3

Sen sijaan, että koko B_3 maksimoitaisiin, on yksinkertaisempaa minimoida vain kaavan nimittäjä, sillä derivaatta on helpompi laskea. Pelkän nimittäjän minimointi on ekvivalenttia koko B_3 maksimointiin:

$$\max_q \frac{1}{\sum_{i,j} q_i q_j F(M_i, M_j)^2} = \frac{1}{\min_q \sum_{i,j} q_i q_j F(M_i, M_j)^2} \tag{5.1}$$

Minimointi q:n suhteen on toteutettu kahdella eri tavalla: laskevan gradientin (engl. *gradient descent*) menetelmällä ja Newtonin menetelmällä (engl. $Newton's\ method$).

5.1.3 B_3 laskeva gradientti

Laskevan gradientin menetelmä on yksi käytetyimmistä optimoinnin menetelmistä (Vandaele, Glineur ja Gillis 2018). Aloitus pisteestä q_0 joukko pisteitä $\{q_t\}$ muodostetaan ottamalla askel gradientin vastakkaiseen suuntaan kerrottuna jollakin vakiolla $-\eta \nabla B_3(q_{t-1})$ jokaisella iteraatiolla $t = 1, 2, ...$ (Vandaele, Glineur ja Gillis 2018). Pisteen q_{t-1} jälkeinen piste lasketaan seuraavasti: $q_t = q_{t-1} - \eta \nabla B_3(q_{t-1})$, missä oppimisaste η (engl. *learning rate*) määrää otetun askeleen pituuden kohti jyrkintä laskua, eli vastakkainen suunta gradientin suhteen (Vandaele, Glineur ja Gillis 2018). Haluamme minimoida parametrin q suhteen, joten gradienttivektori muodostetaan sen derivaatoista, jotka ovat muotoa:

$$\frac{\partial}{\partial q_i} = 2q_i F(M_i, M_i)^2 + 2\sum_{i \neq j} q_j F(M_i, M_j)^2 \tag{5.2}$$

Näistä derivaatioista muodostetaan gradienttivektori:

$$\nabla B_3(q) = \begin{bmatrix} \frac{\partial}{\partial q_1} & \frac{\partial}{\partial q_2} & \frac{\partial}{\partial q_3} & \dots & \frac{\partial}{\partial q_k} \end{bmatrix} \tag{5.3}$$

Jonka jälkeen vektoria q päivitetään gradienttivekotrin avulla:

$$q = q - \eta \nabla B_3(q), \tag{5.4}$$

Tämän jälkeen lasketaan gradientti uudestaan ja päivitetään jälleen q vektoria. Tätä toistetaan niin kauan, kunnes vektorin q muutos on jonkin iteraation jälkeen niin pientä, että ei ole järkeä iteroida pidemmälle, tai kunnes iterointia on toteutettu jonkin ennalta määrätyn määrän verran. Tämän jälkeen ollaan todennäköisesti lähellä minimiä. Seuraavaksi on enää sijoitettava q kaavaan 4.4, jolloin saamme jonkin alarajan.

5.1.4 B_3 Newtonin menetelmällä

Newtonin menetelmä on hyvin samantapainen verrattuna laskevan gradientin menetelmään. Menetelmässä muodostetaan gradienttivekotri, sekä Hessen matriisi (Polyak 2007). Hessen matriisi muodostuu vekotrin q toisen kertaluvun derivaatioista:

$$\nabla^2 B_3(q) = \begin{bmatrix} \frac{\partial}{\partial q_1 \partial q_1} & \cdots & \frac{\partial}{\partial q_1 \partial q_k} \\ \frac{\partial}{\partial q_2 \partial q_1} & \cdots & \frac{\partial}{\partial q_2 \partial q_k} \\ \vdots & \ddots & \vdots \\ \frac{\partial}{\partial q_k \partial q_1} & \cdots & \frac{\partial}{\partial q_k \partial q_k} \end{bmatrix}, \tag{5.5}$$

jonka jälkeen päivitetään vekotria q (Polyak 2007):

$$q = q - \eta[\nabla^2 B_3(q)]^{-1} \nabla B_3(q) \tag{5.6}$$

Hessen matriisin ja gradienttivektorin muodostaminen, ja vektorin q päivittäminen toteutetaan siihen saakka, kunnes samat ehdot kuin laskevan gradientin menetelmässä toteutuvat, jonka jälkeen lasketaan alaraja käyttäen kaavaa 4.4. Newtonin menetelmä toteutuu jos ja vain jos Hessen matriisi on kääntyvä (engl. *invertible*) (Polyak 2007). Molemmissa B_3:n toteutuksessa todennäköisyysjakauma q arvotaan 100 kertaa, ja jokaiselle arvotulle vektorille q toteutetaan minimointiprosessi. Jokaisen vektorin q minimointiprosessin jälkeen lasketaan alaraja käyttäen kaavaa 4.4, joista paras, eli suurin, palautetaan.

5.1.5 $B_4^{'}$

$B_4^{'}$ toteutuksessa käytetään myös gradienttimenetelmää, mutta tällä kertaa maksimoimiseen minimoinnin sijasta. Funktio ei ole derivoituva, joten derivaattaa on arvioitu käyttämällä erotuosamäärää. Ensin lasketaan $P_1 = DM$, missä M on matriisi jolle halutaan laskea PSD-asteen alaraja ja D on epänegatiivinen diagonaalimatriisi, jonka alkiot ovat arvottu. Tämän jälkeen matriisin D diagonaalin arvoja päivitetään $D_{ii}^{'} = D_{ii} + \varepsilon$ ja lasketaan uudestaan $P_2 = D^{'}M$. Nyt matriiseille P_1 ja P_2 lasketaan B_4 mukainen arvo, ja näillä estimoidaan derivaatta erotusosamäärän avulla. Tämä toistetaan jokaiselle matriisin D diagonaali alkiolle:

$$\frac{\partial}{\partial D_{ii}} B_4^{'}(M) \approx \frac{B_4(P_2) - B_4(P_1)}{\varepsilon} \tag{5.7}$$

ja sijoitetaan se gradienttivekotriin:

$$\nabla B_4'(D) = \begin{bmatrix} \frac{\partial}{\partial D_{11}} & \frac{\partial}{\partial D_{22}} & \frac{\partial}{\partial D_{33}} & \dots & \frac{\partial}{\partial D_{kk}} \end{bmatrix} \tag{5.8}$$

Seuraavaksi päivitetään D:n alkioita gradienttivektorin avulla:

$$D_{ii} = D_{ii} + \eta \nabla B_4'(D) \tag{5.9}$$

Lopetus ehdot ovat samat, kuten B_3 ja B_3' toteutuksissa. Matriisi D arvotaan 10 kertaa ja jokaisella matriisilla suoritetaan maksimointi ja tallennetaan laskettu alaraja. Paras alaraja palautetaan.

5.2 Analyyttinen tapa laskea B3

Kyseistä tapaa ei ole toteutettu, mutta on mahdollista laskea B3 analyyttisesti käyttäen Lagrangen menetelmää. Lagrangen menetelmä on optimointimenetelmä, jossa f on minimointi- tai maksimointitehtävän kohdefunktio ja g rajoite-ehtofunktio (Rockafellar 1970). Lagrangen menetelmää siis käytetään optimointitehtäviin, joissa on jokin rajoite. Tässä tapauksessa vekotrin q alkioiden summa tulee olla 1, eli $g(q) = 1 - (q_1 + q_2 + q_3 + ... + q_n) = 0$. Muodostamme Lagrangen funktion (Rockafellar 1970):

$$L(q,\lambda) = B_3(q) - \lambda \cdot g(q) = \frac{1}{\min_q \sum_{i,j} q_i q_j F(M_i, M_j)^2} - \lambda(1 - \sum_{i=1}^{n} q_i), \tag{5.10}$$

missä λ on Lagrangen kerroin (Rockafellar 1970). Seuraavaksi lasketaan gradientti muuttujien q ja λ suhteen:

$$\nabla L(q,\lambda) = \begin{bmatrix} \frac{\partial L}{\partial q_1} & \frac{\partial L}{\partial q_2} & \dots & \frac{\partial L}{\partial q_n} & \frac{\partial L}{\partial \lambda}, \end{bmatrix} \tag{5.11}$$

missä derivaatat ovat muotoa:

$$\frac{\partial L}{\partial q_i} = 2q_i F(M_i, M_i)^2 + 2(\sum_{i \neq j} q_j F(M_i, M_j)^2) + \lambda \tag{5.12}$$

$$\frac{\partial L}{\partial \lambda} = -1 + q_1 + q_2 + \ldots + q_n \tag{5.13}$$

Lagrangen funktion minimi löytyy, kun kaikki osittaisderivaatat ovat nollia (Rockafellar 1970). Lasketaan osion 2.1.2 matriisille alaraja käyttäen menetelmää. Laskemalla fideliteetit saamme muodostettua seuraavan yhtälöryhmän:

$$\begin{cases} 8q_1 + 2q_2 + 2q_3 + \lambda = 0 \\ 8q_2 + 2q_1 + 2q_3 + \lambda = 0 \\ 8q_3 + 2q_1 + 2q_2 + \lambda = 0 \\ -1 + q_1 + q_2 + q_3 = 0 \end{cases} \iff \begin{cases} q_1 = \frac{1}{3} \\ q_2 = \frac{1}{3} \\ q_3 = \frac{1}{3} \\ \lambda = -4 \end{cases} \tag{5.14}$$

Sijoittamalla vektorin $q = \begin{bmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{bmatrix}$ ja osion 2.1.2 matriisin yhtälöön $B3$ saadaan alarajaksi 2, joka on matriisin PSD-aste.

5.3 Toteutetut ylärajat

Ensimmäinen toteutettu yläraja perustuu operoitavan matriisin dimensioihin. Olkoon matriisi M dimensioiltaan $p \times q$, jolloin ylärajana toimii näistä pienin ja se palautetaan. Toinen yläraja, eli neliöjuuriaste, on toteutettu kahdella eri tavalla.

5.3.1 $\texttt{rank}_{\sqrt{}}$

Neliöjuuriasteen ensimmäinen toteutus tapa on seuraavanlainen: muodostetaan jokainen yhdistelmä matriisin alkioiden neliöjuurista, joilla on etumerkkinä joko miinus tai plus merkki.

Jokaisesta tällaisesta yhdistelmästä muodostuu oma matriisinsa. Lasketaan kaikkien näiden matriisien aste ja palautetaan niistä pienin. Muodostettujen matriisien määrä on $2^{p\times q}$. Jos matriisi on esimerkiksi dimensioiltaan 3×3, muodostetaan matriiseja $2^{3\times 3} = 512$ kpl, joka ei ole tietokoneelle raskasta laskea. Jos kyseessä on kuitenkin 5×5 matriisi, on matriisien määrä $2^{5\times 5} = 33554432$ kpl, joka on jo hyvinkin raskasta laskea. Kuten huomamme, neliöjuuriasteen kompleksisuus kasvaa hyvin nopeasti ja se on jopa NP-täydellinen ongelma ratkaista (Fawzi ym. 2015). Toisessa toteutus tavassa ei lasketa kaikkia mahdollisia matriiseja, vaan arvotaan ne. Operoitavan matriisin jokaisen alkion neliöjuuri lasketaan ja arvotaan tuleeko yksittäisen alkion eteen miinus vai plus merkki, jonka jälkeen kyseisen matriisin aste lasketaan. Tämä arvonta toistetaan 100000 kertaa ja paras yläraja, eli pienin, palautetaan. Toista tapaa käytetään vain, jos matriisi on liian iso ensimmäiseen tapaan.

5.4 Matriisit ja niiden PSD-asteet

Julkaisussaan Heinosaari, Kerppo ja Leppäjärvi (2020) esittävät joukon matriiseja ja niiden asteen, epänegatiivisen asteen ja PSD-asteen. Kun kyseisiin matriiseihin operoidaan aiemmin esitetyille ylä- ja alarajoilla, ja verrataan tuloksia julkaisun tuloksiin, huomataan, että matriiseille saadaan ratkaistua PSD-aste käyttäen pelkästään ylä- ja alarajoja.

5.4.1 Matriisit

Matriisit, jotka löytyvät julkaisusta (Heinosaari, Kerppo ja Leppäjärvi 2020):

$$K_+ = \frac{1}{2}\begin{bmatrix}1&1&0&0\\1&0&1&0\\1&0&0&1\\0&1&0&1\\0&0&1&1\end{bmatrix}, \quad K = \frac{1}{2}\begin{bmatrix}1&1&0&0\\1&0&1&0\\0&1&0&1\\0&0&1&1\end{bmatrix}, \quad K_- = \frac{1}{2}\begin{bmatrix}1&1&0&0\\1&0&1&0\\0&1&0&1\end{bmatrix},$$

$$D_{3,1/3} = \begin{bmatrix}\frac{2}{3}&\frac{1}{6}&\frac{1}{6}\\\frac{1}{6}&\frac{2}{3}&\frac{1}{6}\\\frac{1}{6}&\frac{1}{6}&\frac{2}{3}\end{bmatrix}, \quad A = \begin{bmatrix}1&0&0\\\frac{1}{2}&\frac{1}{2}&0\\\frac{1}{2}&0&\frac{1}{2}\end{bmatrix}, \quad B = \begin{bmatrix}\frac{2}{3}&\frac{1}{3}&0\\0&\frac{2}{3}&\frac{1}{3}\\\frac{1}{3}&0&\frac{2}{3}\end{bmatrix},$$

$$C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & 0 & \frac{1}{2} \end{bmatrix}, \quad D = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 1 \end{bmatrix}.$$

5.4.2 Matriisien PSD-asteet

Seuraavaksi verrataan julkaisussa Heinosaari, Kerppo ja Leppäjärvi 2020 esiintyvien matriisien todellisia PSD-asteita eri menetelmillä löytyviin ylä- ja alarajoihin. Jos jokin ylä- tai alaraja ennustaa PSD-asteen oikein, on se maalattu punaisella taulukoissa 1 ja 2.

Taulukko 1. Matriisien PSD-asteiden alarajat verrattuna todellisiin PSD-asteisiin

	K_+	K	K_-	$D_{3,1/3}$	A	B	C	D
$\mathtt{rank}_{\mathrm{psd}}$	3	3	3	2	3	3	3	3
B_1'	3	2	2	2	2	2	2	2
B_3 gradientti menetelmä	3	3	3	2	2	3	3	3
B_3 Newtonin menetelmä	3	3	2	2	2	2	2	3
B_4	3	2	2	2	2	2	2	3
B_4'	3	3	3	2	3	2	3	3

Taulukko 2. Matriisien PSD-asteiden ylärajat verrattuna todellisiin PSD-asteisiin

	K_+	K	K_-	$D_{3,1/3}$	A	B	C	D
$\mathtt{rank}_{\mathrm{psd}}$	3	3	3	2	3	3	3	3
$\min\{p,q\}$	4	4	3	3	3	3	3	3
$\mathtt{rank}_{\sqrt{}}$	3	3	3	2	3	3	3	3

Kuten taulukoita 1 ja 2 tulkitsemalla huomataan, jokaiselle julkaisussa Heinosaari, Kerppo ja Leppäjärvi (2020) esiintyvälle matriisille saadaan ratkaistua PSD-aste käyttämällä eri ylä- ja alarajoja. Esimerkiksi matriisille K_+ saadaan ratkaistua PSD-aste käyttämällä B_3 gradientti menetelmää alarajana ja neliöjuuriastetta ($\mathtt{rank}_{\sqrt{}}$) ylärajana, sillä molemmat antavat rajaksi arvon 3. Taulukoista huomataan, että mikään alaraja ei ole toista parempi. Jotkin ala-rajat antavat joillakin matriiseilla paremman tuloksen, ja joillakin huonomman. Yläarajoista neliöjuuriaste laskee kaikilla matriiseilla oikean ylärajan, kun taas matriisin dimensioihin perustuva yläraja ei.

Kaikille matriiseille ei kuitenkaan löydy PSD-astetta pelkästään käyttämällä ylä- ja alarajoja. Esimerkiksi julkaisussa Fawzi ym. (2015) esiintyvälle matriisille

$$M = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}$$

paras alaraja käyttäen eri metodeja on 2, ja paras yläraja on 4. Täten voimme todeta, että $2 \leq \mathtt{rank}_{\mathrm{psd}}(M) \leq 4$, mutta emme tiedä mikä matriisin M todellinen PSD-aste on. Toisaalta rajoittamalla PSD-asteen reaaliseksi, saamme matriisille alarajan 3, eli $3 \leq \mathtt{rank}^{\mathbb{R}}_{\mathrm{psd}}(M) \leq 4$.

6 Heuristiset menetelmät PSD-hajotelman ratkaisemiseen

Kuten tutkielmassa aiemmin todettiin, PSD-hajotelmien muodostaminen on NP-täydellinen ongelma (Vandaele, Glineur ja Gillis 2018). PSD-hajotelman muodostaminen on myös epäkonveksi ongelma (Vandaele, Glineur ja Gillis 2018). Voimme kuitenkin muodostaa ongelmasta konveksin, jos hajotelman muodostavat matriisit $\{A_1,..,A_m\}$ ja $\{B_1,..,B_n\}$ ovat ennalta määrätty (Vandaele, Glineur ja Gillis 2018). Näin toteutetaan eräs algoritmi, joka esiintyy julkaisussa Vandaele, Glineur ja Gillis (2018).

6.1 Algoritmi PSD-hajotelman arvioimiseen

Olkoon matriisi M $m \times n$ epänegatiivinen matriisi ja $\{A_1,...,A_m\}$ $\{B_1,...,B_n\}$ joukot matriiseja, jotka muodostavat PSD-hajotelman. Vähintään toinen joukoista on ennalta määrätty (Vandaele, Glineur ja Gillis 2018) esimerkiksi arpomalla. Nyt optimoimme molempia joukkoja vuorottelevaa strategiaa käyttäen, julkaisussa Vandaele, Glineur ja Gillis (2018) esiintyvää algoritmia käyttäen:

Algorithm 1 Vuorotteleva strategia PSD-hajotelman muodostamiseen

Input: $M \in \mathbb{R}_+^{m\times n}$ ja alkuarvaukset matriiseille $\{A_1,...,A_m\}$ ja $\{B_1,...,B_n\}$.

Output: $\{A_1,...,A_m\}$ ja $\{B_1,...,B_n\}$.

1: **while** pysäytysehtoa ei ole saavutettu **do**
2: $\quad\{A_1,...,A_m\} \leftarrow$ `optimoi aliongelma` $(M,\{B_1,...,B_n\})$,
3: $\quad\{B_1,...,B_n\} \leftarrow$ `optimoi aliongelma` $(M^T,\{A_1,...,A_m\})$.
4: **end while**

Algoritmin osa `optimoi aliongelma` toteutetaan joukon $\{A_1,...,A_m\}$ matriiseille seuraavasti:

$$A_i \longleftarrow \min_{\mathbf{A}_i \in S_+^k} \sum_{i=1}^{m} \sum_{j=1}^{n} \left(M_{ij} - \langle \mathbf{A}_i, \mathbf{B}_j \rangle\right)^2 \tag{6.1}$$

(Vandaele, Glineur ja Gillis 2018). Matriisit A_i eivät vaikuta toisiinsa, jolloin epäkonveksista ongelmasta muodostuu konveksi (Vandaele, Glineur ja Gillis 2018). Samoin optimoidaan matriistit $\{B_1,...,B_n\}$, vaihtamalla matriisien A_i ja B_j paikkaa:

$$B_i \longleftarrow \min_{\mathbf{B}_i \in S_+^k} \sum_{i=1}^{n} \sum_{j=1}^{m} \left(M_{ij}^T - \langle \mathbf{B}_i, \mathbf{A}_j \rangle\right)^2 \tag{6.2}$$

joka on myös konveksi onglema.

6.2 Algoritmin tehokkuus

Olen toteuttanut julkaisussa Vandaele, Glineur ja Gillis (2018) esiintyvän algoritmin Python ohjelmointikielellä. Toteutuksessa lasketaan ensin PSD-aste käyttämällä aiemmin mainittuja menetelmiä, jonka jälkeen lasketaan jokaiselle mahdolliselle PSD-asteelle oma PSD-hajotelma. Osiossa 5.4.2 esiintyvälle matriisille on arvioitu PSD-hajotelma käyttäen Vandaele, Glineur ja Gillis (2018) algoritmia. Matriisin PSD-astetta ei saada ratkaistua, mutta tiedetään, että se on vähintään 2 ja korkeintaan 4. Jokaiselle mahdolliselle PSD-asteelle muodostetaan oma hajotelma, eli tässä tapauksessa PSD-asteet 2, 3 ja 4. Hajotelman tarkkuutta arvioidaan käyttämällä Frobeniuksen normia: $\|M-X\|_F = \sqrt{\sum_{i=1}^{k}\sum_{j=1}^{l}|(m-x)_{kl}|^2}$, missä M on alkuperäinen matriisi ja X on matriisien $\{A_1,...,A_m\}$ ja $\{B_1,...,B_n\}$ muodostama matriisi. Mitä lähempänä normi on nollaa, sitä tarkempi hajotelma on (Vandaele, Glineur ja Gillis 2018). Esimerkiksi osion 5.4.2 matriisille PSD-asteella 2 saadaan seuraavat PSD-hajotelman muodostavat matriisit:

$$A_1 = \begin{bmatrix} 0.55+0.0i & -0.67-0.32i \\ -0.67+0.32i & 1.0+0.0i \end{bmatrix}$$

$$\vdots$$

$$A_4 = \begin{bmatrix} 0.08+0.0i & 0.16+0.12i \\ 0.16-0.12i & 0.5+0.0i \end{bmatrix}$$

$$B_1 = \begin{bmatrix} 1.56+0.0i & 1.05+0.49i \\ 1.05-0.49i & 0.86+0.0i \end{bmatrix}$$

$$\vdots$$

$$B_4 = \begin{bmatrix} 0.78+0.0i & -0.24-0.2i \\ -0.24+0.2i & 0.12+0.0i \end{bmatrix}$$

Matriisit $\{A_1,...,A_4\}$ ja $\{B_1,...,B_4\}$ muodostavat matriisin $X_{ij} = Tr(A_iB_j)$, joka on muotoa:

$$X = \begin{bmatrix} 1\cdot 10^{-4} & 1 & 1 & 1 \\ 1 & 1\cdot 10^{-4} & 1 & 1 \\ 1 & 1 & 1\cdot 10^{-4} & 1 \\ 1 & 1 & 1 & 1\cdot 10^{-4} \end{bmatrix}$$

Seuraavaksi laskemme Frobeniuksen normin ja saamme tulokseksi $\|M-X\|_F = 1,5\cdot 10^{-4}$, joka on hyvin lähellä nollaa, joka viittaisi matriisin PSD-asteen olevan 2. Emme kuitenkaan saa muodostettua täydellistä PSD-hajotelmaa, sillä normi ei ole tasan nolla. Julkaisussa Fawzi ym. (2015) esitetään tarkka PSD-hajotelma matriisille PSD-asteella 2, eli matriisin PSD-aste on 2. Julkaisun PSD-hajotelma poikkeaa heuristisen menetelmän arviosta. Saamme kuitenkin arvioitua PSD-hajotelmaa hyvällä tarkkuudella.

7 Yhteenveto

Tutkielmassa esitelty PSD-aste on hyödyllinen työkalu eri aloilla, kuten erityisesti kvantti-informaatiossa. PSD-asteen ratkaisemiseen ei löydy analyyttisiä metodeja, joten joudumme tyytymään numeerisesti ratkaistaviin ylä- ja alarajoihin. Nämä ylä- ja alarajat ovat kuitenkin joissakin tapauksissa riittäviä, sillä niiden avulla saadaan mahdollisesti ratkaistua matriisin PSD-aste. Pythonilla toteutetut ylä- ja alarajat pystyivät ennustamaan PSD-asteen usealle eri matriisille. Tuloksista kävi myös ilmi, ettei jokin tietty alaraja ollut välttämättä toista parempi toisin kuin ylärajoissa. Jotkin alarajat toimivat hyvin tapauskohtaisesti paremmin kuin muut, mutta voivat myös joissakin tapauksissa toimia huonommin. Myös Pythonilla toteutettu PSD-hajotelman arvoiminen onnistui hyvin, vaikka tulos ei ollutkaan täydellinen. Toteutetut koodit PSD-asteen rajoille ja PSD-hajotelman arvioinnille löytyvät lähteestä (Alanen 2024).

Lähteet

Alanen, Eetu. 2024. *PSD rank upper and lower bounds.* GitHub repository. https://github.com/Eetutsu/PSD-rank-upper-and-lower-bounds.

Fawzi, Hamza, João Gouveia, Pablo A. Parrilo, Richard Z. Robinson ja Rekha R. Thomas. 2015. “Positive semidefinite rank”. *Mathematical Programming* 153, numero 1 (heinäkuu): 133–177. ISSN: 1436-4646. https://doi.org/10.1007/s10107-015-0922-1. http://dx.doi.org/10.1007/s10107-015-0922-1.

Fiorini, Samuel, Serge Massar, Sebastian Pokutta, Hans Raj Tiwary ja Ronald de Wolf. 2012. “Linear vs. semidefinite extended formulations: exponential separation and strong lower bounds”. Teoksessa *Proceedings of the Forty-Fourth Annual ACM Symposium on Theory of Computing,* 95–106. STOC ’12. New York, New York, USA: Association for Computing Machinery. ISBN: 9781450312455. https://doi.org/10.1145/2213977.2213988. https://doi.org/10.1145/2213977.2213988.

Gillis, Nicolas ja François Glineur. 2012. “On the geometric interpretation of the nonnegative rank”. *Linear Algebra and its Applications* 437, numero 11 (joulukuu): 2685–2712. ISSN: 0024-3795. https://doi.org/10.1016/j.laa.2012.06.038. http://dx.doi.org/10.1016/j.laa.2012.06.038.

Heinosaari, Teiko, Oskari Kerppo ja Leevi Leppäjärvi. 2020. “Communication tasks in operational theories”. *Journal of Physics A: Mathematical and Theoretical* 53, numero 43 (lokakuu): 435302. ISSN: 1751-8121. https://doi.org/10.1088/1751-8121/abb5dc. http://dx.doi.org/10.1088/1751-8121/abb5dc.

Jain, Rahul, Yaoyun Shi, Zhaohui Wei ja Shengyu Zhang. 2013. “Efficient Protocols for Generating Bipartite Classical Distributions and Quantum States”. *IEEE Trans. Inf. Theor.* 59, numero 8 (elokuu): 5171–5178. ISSN: 0018-9448. https://doi.org/10.1109/TIT.2013.2258372. https://doi.org/10.1109/TIT.2013.2258372.

Lahat, Dana, Yanbin Lang, Vincent Y. F. Tan ja Cédric Févotte. 2021. "Positive Semidefinite Matrix Factorization: A Connection With Phase Retrieval and Affine Rank Minimization". *IEEE Transactions on Signal Processing* 69:3059–3074. https://doi.org/10.1109/TSP.2021.3071293.

Lee, Troy, Zhaohui Wei ja Ronald de Wolf. 2017. "Some upper and lower bounds on PSD-rank". *Mathematical Programming* 162 (1): 495–521. ISSN: 1436-4646. https://doi.org/10.1007/s10107-016-1052-0. https://doi.org/10.1007/s10107-016-1052-0.

Nielsen, Michael A. ja Isaac L. Chuang. 2010. *Quantum Computation and Quantum Information.* 10th anniversary edition. Cambridge: Cambridge University Press. ISBN: 978-0-521-63503-5.

Polyak, B.T. 2007. "Newton's method and its use in optimization". *European Journal of Operational Research* 181 (3): 1086–1096. ISSN: 0377-2217. https://doi.org/https://doi.org/10.1016/j.ejor.2005.06.076. https://www.sciencedirect.com/science/article/pii/S0377221706001469.

Rockafellar, R. Tyrrell. 1970. *Convex Analysis.* Princeton, NJ: Princeton University Press.

Shitov, Yaroslav. 2017. "The Complexity of Positive Semidefinite Matrix Factorization". *SIAM Journal on Optimization* 27 (3): 1898–1909. https://doi.org/10.1137/16M1080616. https://doi.org/10.1137/16M1080616.

Vandaele, Arnaud, François Glineur ja Nicolas Gillis. 2018. "Algorithms for positive semidefinite factorization". *Computational Optimization and Applications* 71, numero 1 (maaliskuu): 193–219. ISSN: 1573-2894. https://doi.org/10.1007/s10589-018-9998-x. http://dx.doi.org/10.1007/s10589-018-9998-x.