JYVÄSKYLÄN YLIOPISTO
MATEMATIIKAN JA TILASTO-
TIETEEN LAITOS

Pro gradu-tutkielma

Klassisten ja mallipohjaisten ordinaatiomenetelmien vertailu mikrobiaineistojen analysoinnissa

Aapo Anttila

16. elokuuta 2024

Tekijä
Aapo Anttila

Otsikko
Klassisten ja mallipohjaisten ordinaatiomenetelmien vertailu mikrobiaineistojen analysoinnissa

Tutkinto-ohjelma
Tilastotieteen ja datatieteen maisteriohjelma

Päivämäärä	**Sivumäärä**
16. elokuuta 2024	27 + 5 (liitteet)

Tiivistelmä

Mikrobiaineistot ovat ekologisia runsausaineistoja, jotka sisältävät tietoa biologisten näytteiden mikrobiomeista. Mikrobiaineistojen analysointi on usein haastavaa, sillä aineistot ovat lähes aina suuriulotteisia. Muita mikrobiaineistoille tyypillisiä piirteitä ovat harvuus ja ylihajonta. Suuriulotteisuudesta johtuen mikrobiaineistojen tutkimiseen on usein käytetty ulottuvuuksia vähentäviä ordinaatiomenetelmiä. Ordinaatiomenetelmien avulla moniulotteisen aineiston sisältämä tieto voidaan esittää kahdessa ulottuvuudessa.

Tässä tutkielmassa on tavoitteena vertailla kahta mallipohjaista menetelmää — kopula-ordinaatiomenetelmää ja yleistettyä lineaarista latenttimuuttujamenetelmää — klassisiin ordinaatiomenetelmiin kuten pääkomponenttianalyysiin ja ei-metriseen moniulotteiseen skaalaukseen. Menetelmien vertailu tehtiin simulointikokeiden avulla, joissa aineistoja generoitiin eri menetelmillä. Simuloitujen aineistojen mallina käytettiin oikeaa, Ukrainan Tšernobylin alueelta kerättyä aineistoa. Alueelta pyydystettiin metsämyyriä, joiden suolten mikrobiomeja mitattiin sekvensointimenetelmällä. Aineiston myyrät oli pyydystetty neljästä eri kohteesta, joista kaksi oli säteilyn saastuttamia ja kaksi saastumattomia.

Simulointikokeiden tulokset eivät antaneet yksiselitteistä vastausta siihen, mitkä ordinaatiomenetelmät toimivat parhaiten mikrobiaineistojen analysoimiseen. Eri menetelmät tulkitsevat aineiston sisältämän informaation eri tavoin, joten aineiston generointitapa vaikuttaa siihen, miten hyvin kukin menetelmä toimii. Tulokset osoittivat, että jakaumaperheen tai muunnoksen valinnalla — riippuen menetelmästä — on merkitystä menetelmän toimivuuden kannalta.

Avainsanat: ordinaatio, latenttimuuttujamalli, kopula, simulointi

Sisällys

1 Johdanto

Mikrobiyhteisöjen tutkiminen on tärkeää nykybiologian näkökulmasta. Mikrobeja, eli mikro-organismeja, tutkimalla voidaan tarkastella niiden evoluutiota, keskinäisiä vuorovaikutussuhteita ja niiden vaikutusta esimerkiksi ihmisen terveyteen tai hyvinvointiin. Tieteessä on jo pitkään oltu kiinnostuneita mikrobien potentiaalisista vaikutuksista esimerkiksi ihmisen mielenterveyteen tai lihavuuteen (Alonso & Guarner, 2013). Teknologian ja menetelmien kehitys on kuitenkin vasta hiljattain mahdollistanut suurien mikrobiaineistojen nopeamman ja syvällisemmän analyysin.

Mikrobiaineistot ovat ekologisia runsausaineistoja, jotka koostuvat biologisista näytteistä mitatuista mikrobilajien lukumääriä kuvaavista muuttujista. Johtuen vaikeuttavista ominaispiirteistä, kuten esimerkiksi suuriulotteisuudesta (*high dimensionality*) ja harvuudesta (*sparsity*), mikrobiaineistojen analysointi on haastavaa. Suuriulotteisuus tarkoittaa, että aineistossa on paljon suurempi määrä sarakkeita p kuin rivejä n. Suuri nollien osuus on myös hyvin tyypillistä mikrobiaineistoille (Hawinkel, 2020). Lisäksi aineistot ovat lähes aina kompositionaalisia (ks. luku 2.1). Riippuen tutkittavasta kohteesta, aineiston keruutavasta ja taksonomisesta resoluutiosta, mikrobiaineistoissa voi olla tuhansia tai kymmeniä tuhansia lajien lukumääriä kuvaavia muuttujia. Suuriulotteisuudesta johtuen analyyseissä turvaudutaan usein ulottuvuuksia vähentäviin (*dimension reduction*) menetelmiin. Tässä tutkielmassa käytettyjä (ulottuvuuksia vähentäviä) menetelmiä kutsutaan ordinaatiomenetelmiksi.

Ordinaatiomenetelmät pyrkivät kiteyttämään aineiston sisältämän informaation mahdollisimman hyvin. Ne mahdollistavat aineiston esittämisen kaksiulotteisessa koordinaatistossa. Ordinaatiokuvassa (ks. kuvio 9 luvussa 5) toisiaan lähellä olevat pisteet ovat toistensa kanssa samankaltaisempia kuin toisistaan kaukana olevat. Ordinaatiomenetelmiä käytetään pääasiassa kuvailevaan data-analyysiin, eikä niinkään hypoteesien testaamiseen. Ordinaatiomenetelmiä voidaan käyttää ekologisiin aineistoihin, kun halutaan visuaalisesti tutkia eri tutkimusalueiden tai näytteiden lajistojen samankaltaisuutta.

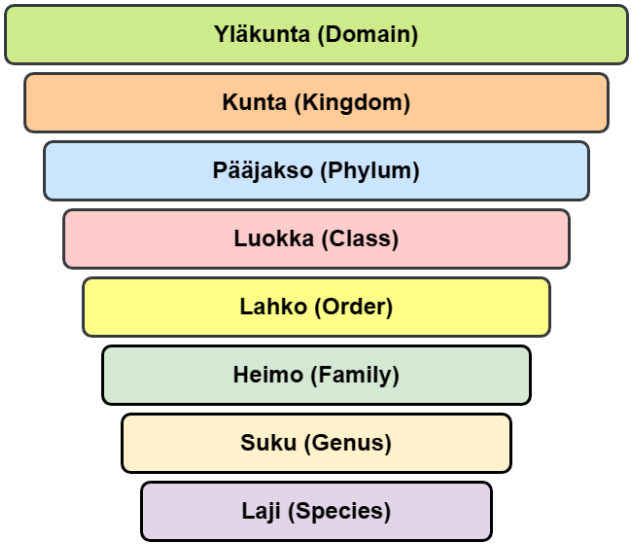
Tämän työn tavoitteena on vertailla pääkomponenttianalyysiä ja ei-metristä moniulotteista skaalausta mallipohjaiseen kopula-ordinaatioon ja yleistettyyn latenttimuuttujamalliin perustuvaan ordinaatiomenetelmään mikrobiaineistojen analysoinnissa. Kiinnostuksen kohteena on myös se, miten menetelmät vertautuvat mikrobiaineistojen haasteellisten ominaisuuksien suhteen. Menetelmiä vertaillaan simulointikokeiden ja esimerkkiaineiston avulla.

Luvussa 2 käsitellään mikrobiaineistoja ja niiden ominaisuuksia, luvussa 3 esitellään työssä käytettävät menetelmät ja luvussa 4 selitetään miten simu-

lointikokeet tehdään ja käydään läpi niiden tuloksia. Luvussa 5 on esimerkki ordinaatiosta sovellettuna mikrobiaineistoon ja luvussa 6 on pohdinta.

2 Mikrobiaineistot

Mikrobiaineistot kuvaavat mikrobien runsautta tai osuutta tutkittavassa alueessa. Mikrobiaineistot ovat dataa biologisten näytteiden mikrobiomeista, eli niiden sisältämistä mikro-organismeista. Mikrobiaineistot saadaan eristämällä näytteistä geneettinen materiaali, josta sekvensointimenetelmiä (Conesa ym., 2016) käyttäen selvitetään mikrobilajit ja niiden suhteelliset osuudet. Mikrobeita voidaan luokitella eri taksonomisilla tasoilla, kuten laji, suku tai heimo. Taksonomiset tasot ovat nimetty kuviossa 1.



Kuvio 1. Taksonomiset tasot ja niiden välinen hierarkia (mitä alempi taso, sitä tarkempi luokittelu).

Aineistojen ulottuvuudet riippuvat siitä kuinka suurta taksonomista resoluutiota käytetään, eli millä tasolla mikrobit luokitellaan. Tutkittavan alueen ominaisuudet voivat myös vaikuttaa aineistojen ulottuvuuksiin, sillä lajirikkaus voi vaihdella paljon erilaisissa ekosysteemeissä.

Kompositionaalisuudesta (ks. luku 2.1) johtuen on huomioitava, että lajien suhteelliset osuudet kompositioissa mahdollisesti yli- tai aliarvioivat lajien todellista osuutta tutkittavassa näytteessä. Tämä johtuu siitä, että sekvensointi löytää eri lajeja eri herkkyydellä. Tätä kutsutaan taksonomiseksi

harhaksi (McLaren ym., 2022). Harhan vaikutus on kuitenkin samankaltainen eri näytteissä, joten kompositioiden vertailu on järkevää. Kompositionaalisuus vaikeuttaa myös lajien välisten korrelaatioiden selvittämistä. Mikrobilajeilla voi olla monimutkaisia korrelaatiorakenteita. Lajien lukumäärät voivat olla riippuvaisia toisistaan: lajit voivat esimerkiksi kilpailla tai tehdä yhteistyötä keskenään.

Mikrobiston monimuotoisuutta näytteessä voidaan kuvata niin sanotulla α-diversiteetillä. Eri näytteiden α-diversiteettien vertailua kuitenkin haittaa se, että havaintomäärät eivät välttämättä ole kaikissa näytteissä samat. Rarefaktio-menetelmän avulla voidaan verrata biologisten näytteiden α-diversiteettiä, jos näytteissä on eri havaintomäärät (Willis, 2019). Jos havaintoja on enemmän näytteestä A kuin näytteestä B, voi näytteen A lajirikkaus vaikuttaa aineiston perusteella suuremmalta kuin näytteen B, vaikkei näytteiden välillä olisikaan eroa. Menetelmällä voidaan muuttaa näytteiden havaintomäärät samankokoisiksi, jolloin ne voivat olla vertailukelpoisempia.

2.1 Kompositionaalisuus

Lajien lukumääriä kuvaavissa aineistoissa arvot ovat aina ei-negatiivisia. Aineiston keruutavasta (sekvensoinnista) johtuen mikrobiaineistot ovat lähes aina kompositionaalisia (Gloor ym., 2017). Kompositionaalisissa aineistoissa (tai rakenneosa-aineistoissa) kompositioita rajoittaa jokin vakio, johon komponentit summautuvat (esimerkiksi vuorokauden tunnit summautuvat 24 tuntiin). Kompositioissa absoluuttisilla arvoilla ei ole suurta merkitystä, vain osuuksilla ja sitä kautta komponenttien välisillä suhteilla on merkitystä (Aitchison, 1982). Näiden rajoitteiden takia standardit tilastolliset menetelmät voivat tuottaa harhaisia tuloksia.

Kompositionaalinen aineisto noudattaa niin sanottua Aitchisonin geometriaa simpleksillä (Filzmoser ym., 2018). Määritellään p-ulotteinen simpleksi, jossa $\boldsymbol{x}$ on jokin kompositio, seuraavasti

$$S^p = \{\boldsymbol{x} = (x_1, \ldots, x_p)^\top \in \mathbb{R}^p | x_i \geq 0, \sum_{i=1}^{p} x_i = \kappa\},$$

jossa κ on jokin ei-negatiivinen vakio.

Aitchisonin työhön perustuen on luotu erilaisia muunnoksia, joilla arvot saadaan kuvattua euklidiseen avaruuteen. Aitchisonin geometria mahdollistaa kompositioiden muuttamisen reaalikoordinaateiksi, jolloin standardeja tilastollisia menetelmiä voi käyttää (Filzmoser ym., 2018). Koska muutos komponentin arvossa vaikuttaa välttämättä jonkin toisen komponentin arvoon, on luonnollista käyttää komponenttien suhteita. Komposition summarajoitteen takia kaikki komposition arvot eivät ole riippumattomia toisistaan.

Tämä johtaa siihen, että p-ulotteinen kompositio voidaan esittää $(p-1)$-ulottuvuudessa menettämättä informaatiota.

Yksi yleisesti käytetty muunnos on clr-muunnos (*centered log ratio*, Filzmoser ym., 2018). Siinä jokainen komponentti jaetaan komposition geometrisella keskiarvolla, jonka jälkeen tehdään logaritmimuunnos

$$\boldsymbol{y} = \mathrm{clr}(\boldsymbol{x}) = (y_1, \ldots, y_p)^\top = \left(\log \frac{x_1}{\sqrt[p]{\prod_{j=1}^p x_j}}, \ldots, \log \frac{x_p}{\sqrt[p]{\prod_{j=1}^p x_j}} \right).$$

Ongelmia clr-muunnoksen yhteydessä aiheuttaa se, että kompositioissa voi olla nolla-arvoja. Koska nollan logaritmia ei ole määritelty, pitää nolla-arvot korvata jollain arvolla. Yksi yleinen tapa on lisätä jokaiseen komponentin arvoon jokin pieni luku.

Toinen mahdollinen muunnos on α-muunnos (Tsagris ym., 2016). Määritellään α-muunnos kompositionaaliselle vektorille $\boldsymbol{x} \in S^p$ seuraavasti

$$z_\alpha(\boldsymbol{x}) = \boldsymbol{H} \cdot \left(\frac{p\boldsymbol{u}_\alpha(\boldsymbol{x}) - \mathbf{1}_p}{\alpha} \right),$$

missä $\alpha > 0$ ja

$$\boldsymbol{u}_\alpha = \left(\frac{x_1^\alpha}{\sum_{k=1}^p x_k^\alpha}, \ldots, \frac{x_p^\alpha}{\sum_{k=1}^p x_k^\alpha} \right)^\top$$

on kompositionaalinen potenssimuunnos (Aitchison, 1982). Matriisin $\boldsymbol{H}$ tarkoitus on poistaa komposition summarajoitteesta johtuva tarpeeton ulottuvuus. Tsagris ym. (2016) ehdottivat matriisiksi $\boldsymbol{H}$ Helmert-matriisia (Lancaster, 1965), josta on poistettu ensimmäinen rivi. Tässä tutkielmassa ei käytetä α-muunnoksen yhteydessä $\boldsymbol{H}$-matriisia (ts. $\boldsymbol{H}$-matriisiksi asetetaan yksikkömatriisi).

Muunnoksessa α voi saada arvoja väliltä 0 ja 1. Muunnoksen raja-arvo nollassa vastaa clr-muunnosta. Jos $\alpha = 1$, niin muunnos on lineaarinen, eli kompositionaalisuuden aiheuttamat rajoitteet jäävät huomiotta. Logaritmimuunnoksiin verrattuna α-muunnoksen etu on se, että nollia ei tarvitse korvata aineistosta.

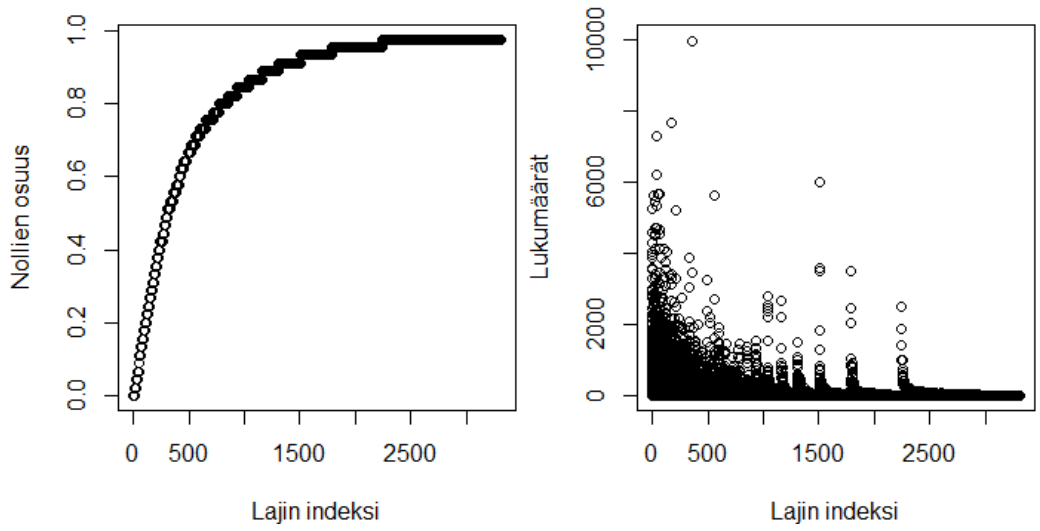
2.2 Myyräaineisto

Myöhemmin esitettävät simulointikokeet perustuvat oikeaan mikrobiaineistoon. Jernfors ym. (2024) tutkivat radioaktiivisen säteilyn vaikutusta suolen mikrobistoon Ukrainan Tšernobylin alueelta pyydystetyissä metsämyyrissä.

Ukrainan Tšernobylin alueelta pyydystettiin neljästä eri sijainnista yhteensä 45 metsämyyrää. Alueista kaksi oli säteilyn saastuttamia (23 myyrää) ja kaksi oli saastumattomia (22 myyrää). Metsämyyristä mitattiin laboratoriossa niiden fyysisiä ominaisuuksia, esimerkiksi pään leveys ja ruumiinpaino. Myyristä arvioitiin myös niiden sisäiset säteilyannokset. Myyristä otettiin ulostenäytteet, joista tutkittiin RNA-sekvensointia käyttäen mikrobilajien lukumäärät. Kaikille näytteille saatiin rarefaktio-menetelmän avulla samat lukumäärien summat.

Saadussa aineistossa on yhteensä 3310 mikrobilajia, eli datamatriisin koko on 45×3310. Lajien lukumäärien summa on kaikissa näytteissä sama, eli 74172. Koko aineistossa nollien määrän osuus on 83.7 %.

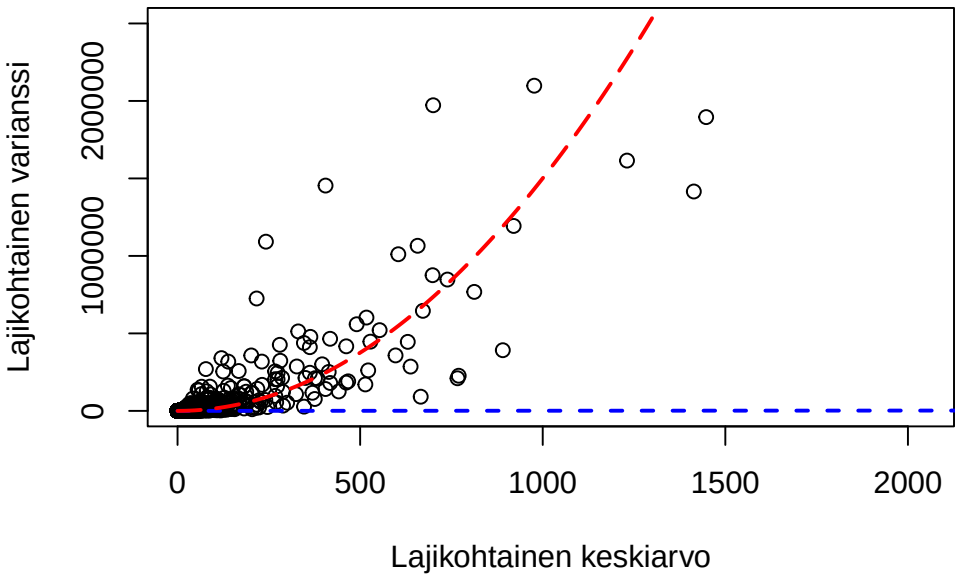
Aineiston harvuutta on havainnollistettu kuviossa 2 (vasen kuvio). Kuvioon on piirretty nollien määrät sarakkeissa, eli kuinka suuressa osassa näytteistä lajin lukumäärä on nolla. Lajien kaikkien havaintojen lukumäärät on myös piirretty (oikea kuvio). Kuviosta käy ilmi, että pieni osa mikrobilajeista vastaa suurimmasta osasta näytteistä mitatuista mikrobeista.



Kuvio 2. Vasemmassa kuvassa esitetetään nollien osuudet sarakkeissa, kun aineisto on järjestetty nollien lukumäärien mukaan. Oikeassa kuviossa esitetään lajien kaikkien havaintojen lukumäärät.

Lajien lukumäärien keskiarvojen ja varianssien suhteita on havainnollistettu kuviossa 3. Lukumäärävasteiden tapauksessa lukumäärien oletetaan usein noudattavan Poisson-jakaumaa, jonka odotusarvo μ ja varianssi σ^2 ovat yhtäsuuret. Kuviosta 3 kuitenkin käy ilmi, että varianssi on keskiarvoa paljon suurempi, joten mallinnettaessa Poisson-jakauman sijasta voisi

olla perusteltua käyttää negatiivista binomijakaumaa (NB-jakauma), joka sallii ylihajonnan. Odotusarvon ja varianssin suhde negatiivisella binomijakaumalla voidaan esittää muodossa $\sigma^2 = \mu + \mu^2\theta$, jossa θ on dispersioparametri. Nollien suuresta määrästä johtuen voisi myös olla perusteltua käyttää nollapaisutettua negatiivista binomijakaumaa (ZINB-jakauma, Greene, 1994).



Kuvio 3. Myyräaineiston lajien lukumäärien varianssit lukumäärien keskiarvoja vasten. Sininen suora kuvaa Poisson-jakauman odotusarvon ja varianssin välistä yhteyttä. Punainen käyrä kuvaa negatiivisen binomijakauman odotusarvon ja varianssin välistä yhteyttä, kun $\theta = 1.5$.

3 Menetelmät

Tässä osiossa esitellään tutkielmassa käytetyt ordinaatiomenetelmät sekä menetelmien vertailuissa käytetty Prokrustes-analyysi. Merkitään tästedes aineistoa matriisina $\boldsymbol{Y} = (\boldsymbol{y}_1 \dots \boldsymbol{y}_n)^\top$, missä $\boldsymbol{y}_i = (y_{i1}, \dots, y_{ip})^\top$, ja y_{ij} on mikrobilajin j lukumäärä näytteessä i.

3.1 Ordinaatiomenetelmät

Ordinaation tarkoituksena on tehdä moniulotteisen aineiston ja sen rakenteiden tutkimisesta helpompaa vähentämällä ulottuvuuksia samalla kun pyritään säilyttämään mahdollisimman paljon aineiston sisältämää informaatiota. Yleensä ordinaatiomenetelmillä ulottuvuudet vähennetään kahteen tai kolmeen ulottuvuuteen, sillä tätä suurempia määriä ulottuvuuksia on vaikea visualisoida. Ordinaatiokuvista voidaan visuaalisesti tutkia havaintojen välisiä samankaltaisuuksia ja erilaisuuksia, sekä esimerkiksi tutkia muodostavatko havainnot rypäitä.

Ekologisten aineistojen analysointiin käytetyistä ordinaatiomenetelmistä eniten käytettyjä ovat pääkomponenttianalyysi (*principal component analysis*, PCA, Hotelling, 1933), moniulotteinen skaalaus (*multidimensional scaling*, MDS, Kruskal, 1964a) ja korrespondenssianalyysi (*correspondence analysis*, CA, Greenacre, 2017).

Tässä tutkielmassa vertaillaan klassisia ordinaatiomenetelmiä, pääkomponenttianalyysiä ja ei-metristä moniulotteista skaalausta (*non metric multidimensional scaling*, NMDS, Kruskal, 1964b), mallipohjaisiin ordinaatiomenetelmiin, eli yleistettyyn lineaariseen latenttimuuttujamalliin (*generalized linear latent variable model*, GLLVM, Niku ym., 2017) ja kopula-ordinaatioon (*copula ordination*, CORD, Popovic ym., 2022). Seuraavissa luvuissa esitellään vertailtavat menetelmät.

3.2 Pääkomponenttianalyysi

Yksinkertaisin ordinaatiomenetelmä on pääkomponenttianalyysi. Pääkomponenttianalyysi on yksi käytetyimmistä monimuuttujamenetelmistä ja moniulotteisen aineiston analyysi alkaa usein siitä.

Pääkomponenttianalyysi perustuu aineiston kovarianssimatriisin ominaisarvoihin ja ominaisvektoreihin. Pääkomponenttianalyysissä muodostetaan uusia muuttujia, pääkomponentteja, jotka ovat lineaarisia kombinaatioita aineiston alkuperäisistä muuttujista. Nämä uudet muuttujat ovat toistensa

kanssa korreloimattomia ja ne on järjestetty sen mukaan, kuinka suuri niiden varianssi on. Ordinaatiossa pääkomponentit muodostavat ordinaatioakselit.

Mikrobiaineistoissa muuttujia (ts. lajeja) on usein paljon enemmän kuin näytteitä, minkä johdosta käytetään singulaariarvohajotelmaa (*singular value decomposition*, SVD, Wall ym., 2003) pääkomponenttianalyysin toteuttamiseen. Merkitään $n \times p$ -matriisia $\boldsymbol{Y}$, jolloin sen singulaariarvohajotelma on muotoa $\boldsymbol{Y} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^\top$, jossa $\boldsymbol{U}$ on $n \times n$ -matriisi, $\boldsymbol{V}^\top$ on $p \times p$ -matriisi ja $\boldsymbol{S}$ on $n \times p$ -matriisi. Matriisit $\boldsymbol{U}$ ja $\boldsymbol{V}$ ovat ortogonaalisia, ja $\boldsymbol{S}$ on singulaariarvoista koostuva diagonaalimatriisi. Hajotelmasta pääkomponentit $\boldsymbol{P}$ saadaan kaavalla $\boldsymbol{P} = \boldsymbol{U}\boldsymbol{S}$.

Kompositionaalisuudesta johtuen on suositeltavaa tehdä aineistolle jokin muunnos, joka poistaa siltä komposition rajoitteet (Filzmoser ym., 2018). Yleisesti PCA-menetelmän yhteydessä on käytetty clr-muunnosta (ks. luku 2.1). Toinen mahdollinen muunnos on α-muunnos (ks. luku 2.1). Myöhemmin tehdyissä simulointivertailuissa käytetään molempia muunnoksia.

Pääkomponenttianalyysi toteutetaan R:n (R Core Team, 2024) `stats`-paketilla käyttäen `prcomp()`-funktiota.

3.3 Ei-metrinen moniulotteinen skaalaus

Yksi suosituimmista mikrobiaineistoihin sovelletuista ordinaatiomenetelmistä on moniulotteinen skaalaus. Tässä tutkielmassa käytetään parametritonta ei-metristä moniulotteista skaalausta, joka eroaa moniulotteisesta skaalauksesta siten, että etäisyyksien sijaan käytetään niiden järjestyslukuja.

Moniulotteisessa skaalauksessa valitaan jokin etäisyysmitta. Jokaiselle aineiston havaintoparille lasketaan niiden etäisyys. Etäisyysmitan valintaan vaikuttaa pääasiassa aineiston ominaisuudet. Bray-Curtis-etäisyys (Bray & Curtis, 1957) on suosittu ekologisten aineistojen analysoinnissa. Se ottaa huomioon sekä runsauden että näytteen mukana- ja poissaolon. Se soveltuu hyvin mikrobiaineistoille, joissa nollien määrä on usein hyvin suuri. Määritellään Bray-Curtis-etäisyys kahden näytteen h ja i välillä

$$d_{hi} = \frac{\sum_{j=1}^{p} |y_{hj} - y_{ij}|}{\sum_{j=1}^{p} (y_{hj} + y_{ij})},$$

jossa $j = 1, \ldots, p$ on lajin indeksi.

Merkitään näytettä h vastaavia koordinaatteja $\boldsymbol{x}_h = (x_{h1}, \ldots, x_{hq})^\top$, jossa q on valittu ulottuvuuksien määrä. Asetetaan alkuarvot koordinaateille, jonka jälkeen lasketaan koordinaattien väliset euklidiset etäisyydet $\hat{d}_{hi} = \|\boldsymbol{x}_h - \boldsymbol{x}_i\|$. Koordinaatteja siirrellään siten, että etäisyyksien yhteensopivuus-

mitta (*stress*), joka on muotoa

$$S = \sqrt{\frac{\sum(d_{hi} - \hat{d}_{hi})^2}{\sum d_{hi}^2}},$$

minimoituu. Jos yhteensopivuusmitan arvo on nolla, niin yhteensopivuus on täydellinen (Kruskal, 1964a). NMDS-menetelmässä yhteensopivuusmitta perustuu etäisyyksien järjestyslukuihin. Lopulliset koordinaatit muodostavat ordinaatiopisteet ordinaatiokuvassa.

NMDS tehdään aineistolle käyttämällä R-pakettia `vegan` (Oksanen ym., 2022).

3.4 Yleistetty lineaarinen latenttimuuttujamalli

Latenttimuuttujamallien juuret ovat Spearmanin faktorianalyysissä (Spearman, 1904). Latenttimuuttujamallien motivaationa on, että usein tärkeitä muuttujia ei ole havaittu tai muuten osattu ottaa huomioon. Niiden avulla voidaan myös huomioida tai tutkia havaintojen välisiä korrelaatiosuhteita pienemmässä ulottuvuudessa (Bartholomew ym., 2011).

Yleistetty lineaarinen latenttimuuttujamalli laajentaa yleistetyn lineaarisen mallin moniulotteiselle aineistolle käyttämällä faktorianalyysi-tyyppistä lähestymistapaa (Niku ym., 2017). Jokaiselle tutkittavalle alueelle määrätään pieni määrä (yleensä kaksi) latenttimuuttujaa, joiden avulla huomioidaan lajien välinen korrelaatio. Olkoon nyt $\mu_{ij} = E(y_{ij})$ vasteen odotusarvo, latenttimuuttujamalli olettaa, että

$$g(\mu_{ij}) = \eta_{ij} = \alpha_i + \beta_{0j} + \boldsymbol{u}_i^\top \boldsymbol{\gamma}_j,$$

jossa $g(\cdot)$ on linkkifunktio ja β_{0j} lajikohtainen vakio. Kerroin α_i on kiinteä riviparametri, jonka avulla pyritään mallintamaan suhteellista runsautta (Warton ym., 2015). Vektori $\boldsymbol{u}_i = (u_{i1}, \ldots, u_{id})^\top$ koostuu d standardinormaalijakautuneesta latenttimuuttujasta. Vektori $\boldsymbol{\gamma}_j = (\gamma_{j1}, \ldots, \gamma_{jd})^\top$ sisältää lajikohtaiset lataukset. Latentit muuttujat oletetaan olevan riippumattomia ja niiden oletetaan noudattavan standardinormaalijakaumaa. Kun menetelmää käytetään ordinaatioon, latenttien muuttujien ennusteet ovat ordinaatiopisteitä.

Lajien välisiä jäännöskorrelaatioita kuvataan $p \times p$ -matriisilla $\boldsymbol{\Sigma} = \boldsymbol{\Gamma}\boldsymbol{\Gamma}^\top$, jossa $\boldsymbol{\Gamma} = [\boldsymbol{\gamma}_1, \ldots, \boldsymbol{\gamma}_p]^\top$. Jotta malli olisi identifioituva, latentit muuttujat $\boldsymbol{u}_i$ oletetaan standardinormaalijakautuneiksi. Lisäksi matriisin $\boldsymbol{\Gamma}$ yläkolmio tulee asettaa nollaksi ja diagonaalit positiivisiksi. Normaalisuusoletus kiinnittää lokaation ja skaalan, ja yläkolmiorajoite kiinnittää rotaation.

3.4.1 Estimointi

Malli sovitetaan käyttämällä uskottavuuspäättelyä. Havainnot y_{ij} ovat riippumattomia, kun ne ehdollistetaan u_{ij}:lla. Uskottavuusfunktio GLLVM-menetelmälle voidaan kirjoittaa muodossa

$$L(\boldsymbol{\Psi}; \boldsymbol{u}) = \prod_{i=1}^{n} \left(\prod_{j=1}^{p} (f(y_{ij}|\boldsymbol{u_i}, \boldsymbol{\Psi})) \right) f(\boldsymbol{u}_i),$$

jossa $\boldsymbol{\Psi}$ sisältää kaikki mallin parametrit (Niku, 2020). Latentit muuttujat eivät ole havaittuja, joten niiden mahdollisten arvojen yli tulee integroida. Tavoitteena on maksimoida marginaali log-uskottavuusfunktiota (Niku, 2020), joka on muotoa

$$l(\boldsymbol{\Psi}) = \sum_{i=1}^{n} \log(f(\boldsymbol{y}_i, \boldsymbol{\Psi})) = \sum_{i=1}^{n} \log \left(\int_{R^d} \prod_{j=1}^{p} f(y_{ij}|\boldsymbol{u}_i; \boldsymbol{\Psi}) f(\boldsymbol{u}_i) d\boldsymbol{u}_i \right).$$

Log-uskottavuusfunktion integraalille ei ole useimmissa tapauksissa suljetun muodon ratkaisua, joten log-uskottavuuden sijasta maksimoidaan sen approksimaatiota. Approksimointiin voidaan käyttää esimerkiksi variaatioapproksimaatiota (VA, Hui ym., 2017), Laplacen approksimaatiota (LA, Huber ym., 2004) tai näiden yhdistelmää, laajennettua variaatioapproksimaatiota (EVA, Korhonen ym., 2023), riippuen käytettävästä jakaumaperheestä (Niku ym., 2019). LA on helpoin ja toimii aina, mutta on epätarkempi kuin VA. VA ei toimi jokaisella jakaumaperheellä, sillä suljetun muodon approksimaatio on saatu vain tietyille vastejakauma ja linkkifunktio pareille (Korhonen ym., 2023). EVA toimii myös aina.

Mallin sovitukseen käytetään R-paketin `gllvm` (Niku ym., 2019) funktiota `gllvm()`.

3.5 Kopula

Kopuloilla mallintaminen perustuu Sklarin teoreemaan (Sklar, 1959). Teoreeman mukaan p-muuttujaisen satunnaismuuttujan yhteiskertymäfunktio $H(x_1, x_2, \ldots, x_p)$ voidaan esittää muodossa

$$H(x_1, x_2, \ldots, x_p) = C(F_1(x_1), F_2(x_2), \ldots, F_p(x_p)),$$

jossa F_j on kertymäfunktio lajille j, ja $C : [0, 1]^p \to [0, 1]$ on kopula.

Kopuloita käytetään satunnaismuuttujien välisten riippuvuuksien kuvailemiseen. Kopulamallit saavat nimityksensä (lat. *copula*, 'yhdysside') sen

vuoksi, että ne yhdistävät moniulotteisen jakauman (esimerkiksi moniulotteisen normaalijakauman) minkä tahansa marginaalijakaumajoukon kanssa (Popovic ym., 2019). Kopulamallissa lajien runsauksia kuvataan normaalijakautuneilla kopula-arvoilla siten, että seuraava ehto täyttyy

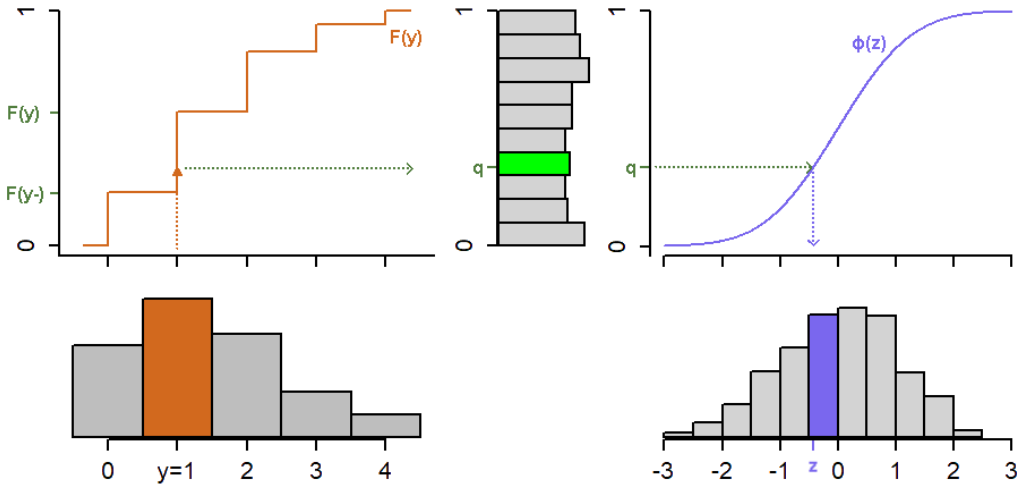
$$F_j(y_{ij}^-) \leq \Phi(z_{ij}) < F_j(y_{ij}),$$

jossa Φ on standardinormaalijakauman kertymäfunktio ja $F_j(y_{ij}^-)$ tarkoittaa kertymäfunktion F_j raja-arvoa y_{ij}:ssä. Diskreeteille lukumäärävasteille pätee $F_j(y_{ij}^-) = F_j(y_{ij} - 1)$.

Kopulan arvojen $\boldsymbol{z}_j$ oletetaan noudattavan moniulotteista normaalijakaumaa ja täyttävän ehdon

$$z_{ij} = \boldsymbol{u}_i^\top \boldsymbol{\gamma}_j + \epsilon_{ij}, \quad \epsilon_{ij} \sim N(0, \sigma_j^2),$$

jossa $\boldsymbol{u}_i$, eli latentit arvot, ja $\boldsymbol{\gamma}_j$, eli lataukset, ovat kuten edellisessä luvussa. Parametri σ_j^2 on lajikohtainen varianssi. Sekä latentit muuttujat että virheet ϵ_{ij} ovat riippumattomia ja normaalijakautuneita. Ordinaatiokuvassa latenttien muuttujien arvot muodostavat ordinaatiopisteet. Kopula-arvon muodostumista on havainnollistettu kuviossa 4.



Kuvio 4. Kuviossa on havainnollistettu kopula-arvon muodostamista. Marginaalimallin mukainen (empiirinen) kertymäfunktio $F(y)$ kuvataan standardinormaalijakauman kertymäfunktioon. Koska $F(y)$ on porrasfunktio, käytetään kopulan arvon saamiseksi apuna marginaalimallista laskettuja Dunn-Smyth-jäännöksiä (ks. luku 3.5.1).

Kopula-ordinaatio tehdään R:ssä `ecoCopula`-paketin (Popovic ym., 2019) `cord()`-funktiolla. Funktio ottaa parametrinä aineistoon sovitetun saman paketin `stackedsdm`-objektin tai `mvabund`-paketin (Wang ym., 2022) `manyglm`-objektin. Funktiolla `manyglm()` voidaan sovittaa jokaiselle lajille oma yleistetty lineaarinen malli. Myös `stackedsdm()` sovittaa jokaiselle lajille oman regressiomallin, sillä erolla, että se mahdollistaa eri jakaumaperheiden käytön eri lajeille. Tässä tutkielmassa käytetään marginaalimallien sovittamiseen `gllvm()`-funktiota ilman latentteja muuttujia, sillä se mahdollistaa sekä nollapaisutetun negatiivisen binomijakauman käytön että riviparametrin estimoinnin. Toteutuksen R-koodi löytyy liitteistä.

3.5.1 Kopulan estimointi

Esitetään kopulan estimointi mukaillen artikkelia Popovic ym. 2022.

Kopulan estimointi alkaa marginaalijakaumien $F_j(\cdot)$ estimoinnista käyttäen yleistettyjä lineaarisia malleja. Seuraavaksi estimoidaan kovarianssiparametrit $\boldsymbol{\theta} = (\boldsymbol{\gamma}_1^\top, \ldots, \boldsymbol{\gamma}_p^\top, \sigma_1, \ldots, \sigma_p)^\top$ käyttäen Monte Carlo Expectation Maximisation (MCEM) -algoritmia (Levine & Casella, 2001). MCEM-algoritmi on muunnelma EM-algoritmista (Dempster ym., 1977), jossa E-askeleessa odotusarvo lasketaan numeerisesti käyttäen Monte Carlo -simulointeja.

MCEM-algoritmin Q-funktio m:nnessä iteraatiossa saa muodon

$$Q(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(m)}) = \sum_{i=1}^{n} \int f(\boldsymbol{z}_i|\boldsymbol{y}_j; \boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}^{(m)}}) \log f(\boldsymbol{z}_i; \boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}^{(m)}}) d\boldsymbol{z}_i,$$

jossa $f(\boldsymbol{z}_i; \boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}^{(m)}}) d\boldsymbol{z}_i = N_p(\boldsymbol{z}_i; \boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}^{(m)}})$.

E-askeleessa otetaan otoksia ehdollisesta jakaumasta $f(\boldsymbol{z}_i|\boldsymbol{y}_j; \boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}^{(m)}})$. Otosten saamiseen käytetään Dunn-Smyth-jäännöksiin (Dunn & Smyth, 1996) pohjautuvaa tärkeysotantaa. Hui ym. (2015) määrittelivät Dunn-Smyth-jäännökset seuraavasti

$$r_{ij} = \Phi^{-1}(t_{ij} F_{ij}(y_{ij}) + (1 - t_{ij}) F_{ij}^{-}(y_{ij})),$$

jossa $\Phi(\cdot)$ on normaalijakauman ja $F_{ij}(\cdot)$ on vastemuuttujan y_{ij} kertymäfunktio. F_{ij}^{-} on F_{ij}:n raja-arvo, kun sitä lähestytään negatiiviselta puolelta. Arvot t_{ij} generoidaan tasajakaumasta $U(0, 1)$, jolloin todennäköisyysmassa saadaan levitetyksi tasaisesti diskreettien arvojen välille (Hui ym., 2015). Dunn-Smyth-jäännökset ovat ei-normaalisti jakautuneille muuttujille suunnattuja, kvantiileihin perustuvia satunnaistettuja jäännöksiä, joiden tulisi olla normaalisti jakautuneita mallin ollessa yhteensopiva aineiston kanssa.

Merkitään nyt Dunn-Smyth-jäännöksiä $\boldsymbol{r}_i = (r_{i1}, \ldots, r_{ip})^\top$. Painot tärkeysotantaan saadaan kaavalla

$$w_{ik}(\boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}^{(m)}}) = \frac{N_p(\boldsymbol{r}_i^{(k)}; \boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}^{(m)}})}{N_p(\boldsymbol{r}^{(k)}; I)},$$

jossa $\boldsymbol{r}^{(k)}$ on otos $k = 1, \ldots, K$ Dunn-Smyth-jäännöksistä. Q-funktiota approksimoidaan seuraavasti

$$Q(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}^{(m)}) \approx \sum_{i=1}^{n} \sum_{k=1}^{K} w_{ik}(\boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}^{(m)}}) \log f(\boldsymbol{r}_i^{(k)}; \boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}^{(m)}}).$$

Approksimaatiota maksimoimalla saadaan päivitettyä estimaatit $\hat{\boldsymbol{\theta}}^{(m+1)}$. Maksimointi voidaan tehdä käyttämällä faktorianalyysin algoritmeja, jossa aineiston kovarianssimatriisi korvataan painotetulla Dunn-Smyth-jäännösten kovarianssimatriisilla. E- ja M-askelia toistetaan, kunnes algoritmi konvergoi. Latenttien muuttujien arvoille saadaan estimaatit käyttämällä faktorianalyysiä.

3.6 Prokrustes-analyysi

Menetelmien vertailuun käytetään Prokrustes-analyysiä (Peres-Neto & Jackson, 2001). Prokrustes-analyysi on menetelmä, jolla kahden (tai useamman) pistekonfiguraation erojen neliösummat minimoidaan. Konfiguraatiot yritetään yhteensovittaa siirtämällä, skaalaamalla, kääntämällä ja mahdollisesti peilaamalla.

Simulointikokeissa simuloituihin aineistoihin sovelletaan eri ordinaatiomenetelmiä, ja niiden tuottamia ordinaatiokuvia verrataan aineiston generointiin käytettäviin latenttien muuttujien arvoihin (ks. luku 4). Prokrustes-analyysillä saadaan tuloksena käännetty matriisi, jota verrataan todellisten arvojen konfiguraatioon laskemalla pisteiden etäisyyksien (ts. virheiden) summat. Paremmin toimivat menetelmät tuottavat pienemmät virheet.

Prokrustes-virheiden summa lasketaan seuraavasti

$$\sum_{i=1}^{n} \sum_{r=1}^{d} (x_{ir,rot} - x_{ir})^2,$$

jossa $x_{ir,rot}$ on jollain ordinaatiomenetelmällä (k.s luku 3) saatu koordinaatti latentille muuttujalle r näytteessä i, joka on käännetty käyttäen Prokrustes-analyysiä. Koordinaatti x_{ir} on latentin muuttujan r todellinen arvo näytteessä i.

Prokrustes-analyysi tehdään `vegan`-paketin (Oksanen ym., 2022) `procrustes()`-funktiolla.

4 Simulointikokeet

Verrataan seuraavaksi simulointikokeiden avulla mallipohjaisia menetelmiä klassisiin ordinaatiomenetelmiin. Tarkennettuna, kokeissa verrataan kopula-ordinaatiota ja GLLVM-menetelmää PCA- ja NMDS-menetelmiin. PCA-menetelmän yhteydessä käytetään sekä clr- että α-muunnosta. Koska clr-muunnos ei toimi, kun aineistossa on nollia, lisätään jokaiseen aineiston arvoon luku yksi ennen muunnoksen tekemistä. Kokeillaan myös α-muunnosta käyttäen arvoja $\alpha = 0.2, 0.5$ ja 0.8.

Eri menetelmien vertailemiseksi simuloidaan aineistoja. Aineistoja simuloitaessa voidaan itse määrätä todelliset näytteiden välisiä suhteita kuvaavien latenttien muuttujien arvot. Tällöin voidaan suoraan verrata eri menetelmien tuottamia ordinaatiokuvia todellisten latenttien muuttujien arvojen muodostamaan kuvaan. Koska työn tarkoituksena on vertailla ordinaatiomenetelmiä, käytetään aineistojen simuloinnissa kahta latenttia muuttujaa.

Simulointikokeiden tarkoituksena on selvittää miten simulointiasetelma vaikuttaa tuloksiin (ks. luku 4.1). Niiden tarkoituksena on myös selvittää miten aineistojen koko vaikuttaa tuloksiin.

4.1 Simulointiasetelma

Vertaillaan luvussa 3 esitettyjä menetelmiä neljällä eri simulointiasetelmalla. Simuloinnit perustuvat myyräaineistoon (ks. luku 2.2). Käytetään vertailuissa erikokoisia osa-aineistoja, ja tutkitaan, miten lajien määrä, ja siten aineiston harvuus, vaikuttaa tuloksiin.

Simulointikokeen vaiheet, kun aineisto generoidaan GLLVM-menetelmällä, ovat seuraavat:

1: Järjestetään aineiston sarakkeet nollien määrän mukaan laskevaan järjestykseen ja redusoidaan aineisto m sarakkeeseen pudottamalla pois sarakkeen m jälkeiset sarakkeet.

2: Sovitetaan NB tai ZINB-latenttimuuttujamalli redusoituun aineistoon `gllvm()`-funktiolla.

3: Generoidaan lukumäärävasteet NB tai ZINB-jakaumasta käyttämällä sovitetun mallin parametrien estimaatteja ja ennustettuja latentteja muuttujia.

4: Estimoidaan simuloidusta aineistosta ordinaatiopisteet luvussa 3 esitetyillä menetelmillä.

5: Verrataan kohdassa 4 saatuja pistekonfiguraatioita kohdassa 2 saatuun konfiguraatioon laskemalla Prokrustes-virheet, kuten luvussa 3.6.

6: Toistetaan kohdat 2-5 K kertaa.

Kopula-menetelmällä generoitaessa vaiheet ovat samat lukuun ottamatta vaiheita 2 ja 3. Vaiheessa 2 sovitetaan moniulotteinen yleistetty lineaarinen malli `gllvm()`-funktiolla ja tehdään kopula-ordinaatio käyttäen sovitettuja malleja. Vaiheessa 3 saatuihin kopula-arvoihin lisätään moniulotteisesta normaalijakaumasta simuloidut arvot ja kuvataan uudet kopula-arvot takaisin lukumääriksi.

4.2 Simulointien tulokset

Merkitään nyt `cnb` tarkoittamaan kopula-menetelmää NB-jakaumalla, ja `czi` ZINB-jakaumalla. GLLVM-menetelmälle vastaavat ovat `gnb` ja `gzi`. PCA clr-muunnoksella on `clr` ja `a0.2`, `a0.5` ja `a0.8` ovat PCA α-muunnoksella, jossa numero ilmaisee α-arvon. Ei-metristä moniulotteista skaalausta vastaa merkintä `nmds`.

Kuviossa 5 on esitetty Prokrustes-virheet, kun aineistot generoitiin käyttämällä kopula-menetelmää ja ZINB-jakaumaa. Kuvio 7 esittää vastaavat tulokset, kun käytettiin kopula-menetelmää ja NB-jakaumaa. Tulokset, kun aineistot generoitiin GLLVM-menetelmällä ovat kuvioissa 6 (ZINB-jakauma) ja 8 (NB-jakauma). Lajien määrät olivat 50, 100, 200 ja 400, ja jokaisella lajien määrällä generoitiin $K = 1000$ aineistoa.

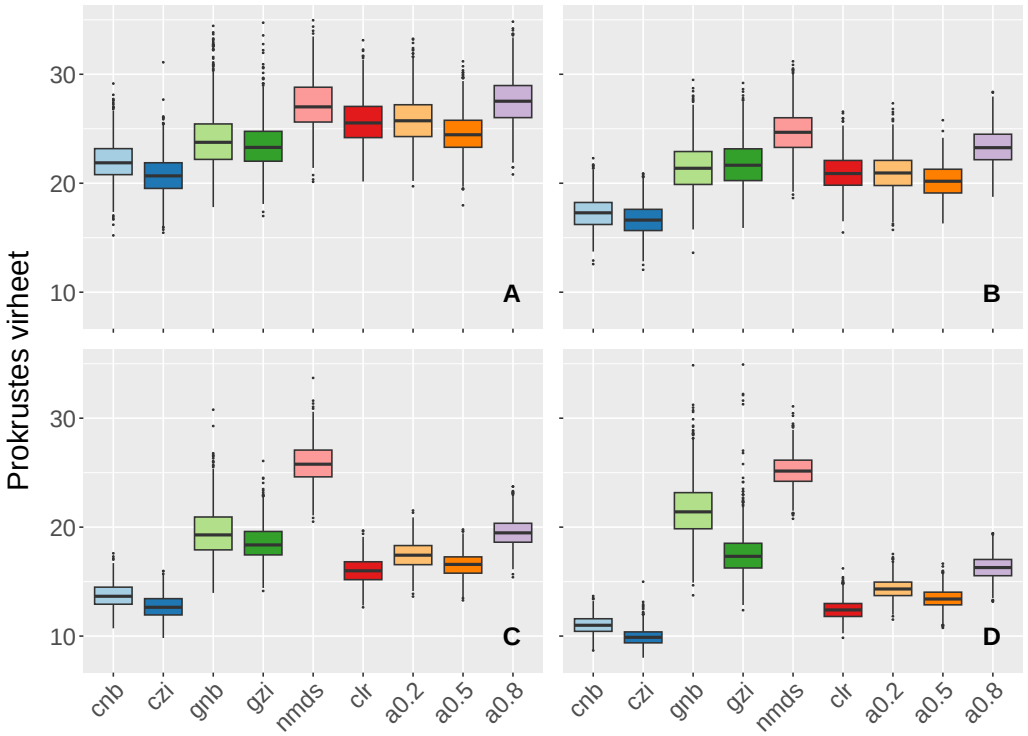
Kuviosta 5 nähdään, että kopula-menetelmällä (ZINB-jakauma) generoitujen aineistojen tapauksessa kopula-menetelmä toimi parhaiten. Kun lajien määrä oli 50, GLLVM toimi paremmin kuin klassiset menetelmät, mutta sitä isommilla määrillä PCA oli parempi. NMDS tuotti suurimmat Prokrustes-virheet aineiston koosta riippumatta. Simulointien tuloksista on vaikea sanoa kumpi muunnos — clr vai α — toimii PCA-menetelmän yhteydessä paremmin. Kopula-menetelmän tapauksessa ZINB-jakauma antoi paremmat tulokset NB-jakaumaan verrattuna. Pienemmillä lajien määrillä ero oli vähäinen. GLLVM-menetelmän tapauksessa NB- ja ZINB-jakaumien käytöllä ero oli huomattava vain, kun lajien määrä oli 400. Lukuun ottamatta NMDS- ja GLLVM-menetelmää NB-jakaumalla, tulokset paranivat aineiston koon kasvaessa.

Kun aineistot generoitiin ZINB-jakaumasta käyttämällä GLLVM-menetelmää (kuvio 6), antoi GLLVM parhaimmat tulokset. ZINB-jakauma toimi paremmin kuin NB-jakauma. ZINB-jakaumaa käytettäessä kopula toimi paremmin kuin klassiset menetelmät, mutta ero PCA-menetelmään, kun muunnoksena käytettiin α-muunnosta α-arvoilla 0.5 ja 0.8, pieneni aineiston koon kasvaessa. Sekä NMDS että PCA clr-muunnoksella toimivat huonosti. GLLVM-menetelmän antamat tulokset paranivat aineiston koon kasvaessa. Myös kopula-menetelmän antamat tulokset ZINB-jakaumaa käytettäessä paranivat, mutta ei NB-jakaumaa käytettäessä.

Taulukoissa 1 ja 2 on esitetty suoritusajat kaikille menetelmille. Käytetyillä aineiston ko'oilla mallipohjaiset menetelmät olivat hitaampia kuin klassiset menetelmät. Kopula-menetelmän suoritusajasta suurin osa kului marginaalimallien sovitukseen. Jos mallit olisi sovitettu käyttämällä esimerkiksi `manyglm()`-funktiota, olisi kopula nopeampi kuin GLLVM.

Taulukko 1: Menetelmien suoritusaikojen keskiarvot sekunneissa (Intel Xeon CPU E7-8890 v4 (2.20GHz)), kun aineistot generoitiin kopula-menetelmällä (ZINB-jakauma). Kopula-menetelmän tapauksessa suluissa oleva aika on marginaalimallien sovitukseen kulunut aika.

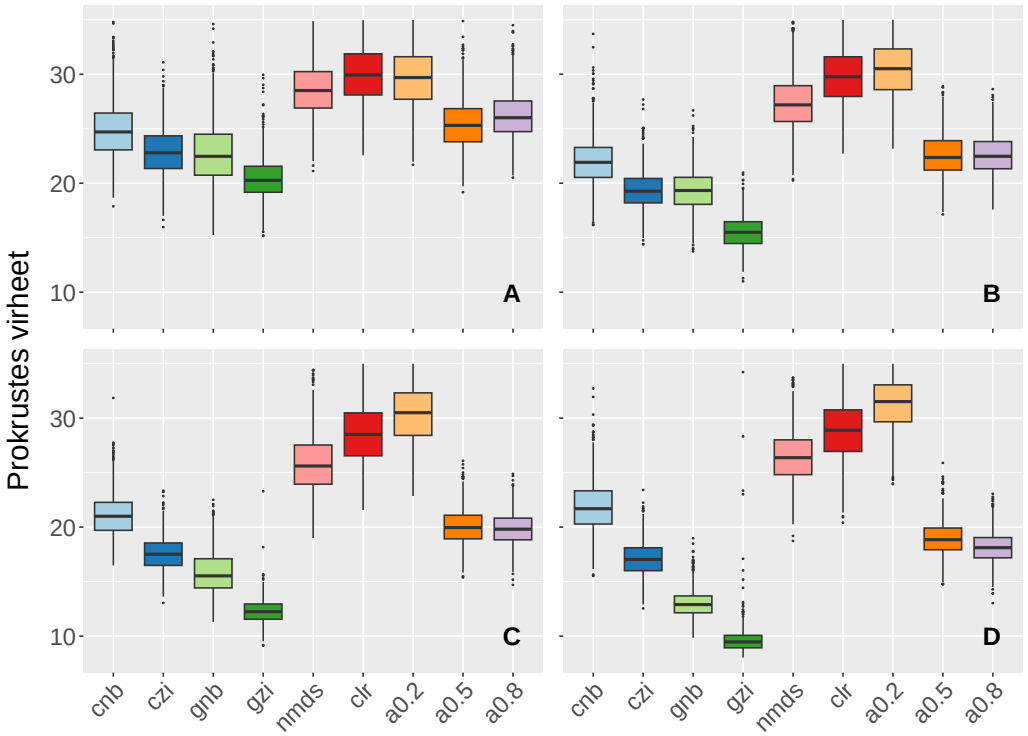
m	cnb	czi	gnb	gzi	nmds	pca
50	0.9 (3.7)	1.8 (4.1)	2.7	3.7	0.1	<0.1
100	1.3 (10.7)	4.1 (11.8)	5.3	8.6	0.1	<0.1
200	2.4 (41.2)	8.9 (43.2)	17.6	29.0	0.1	<0.1
400	5.0 (178.8)	24.0 (185.5)	70.5	115.8	0.1	<0.1



Kuvio 5. Prokrustes-virheiden viiksikuviot, kun aineistot generoitiin kopula-menetelmällä ja kun jakaumana käytettiin ZINB-jakaumaa. Lajien lukumäärät ovat 50 (**A**), 100 (**B**), 200 (**C**) ja 400 (**D**).

Taulukko 2: Menetelmien suoritusaikojen keskiarvot sekunneissa (Intel Xeon CPU E7-8890 v4 (2.20GHz)), kun aineistot generoitiin GLLVM-menetelmällä (ZINB-jakauma). Kopula-menetelmän tapauksessa suluissa oleva aika on marginaalimallien sovitukseen kulunut aika.

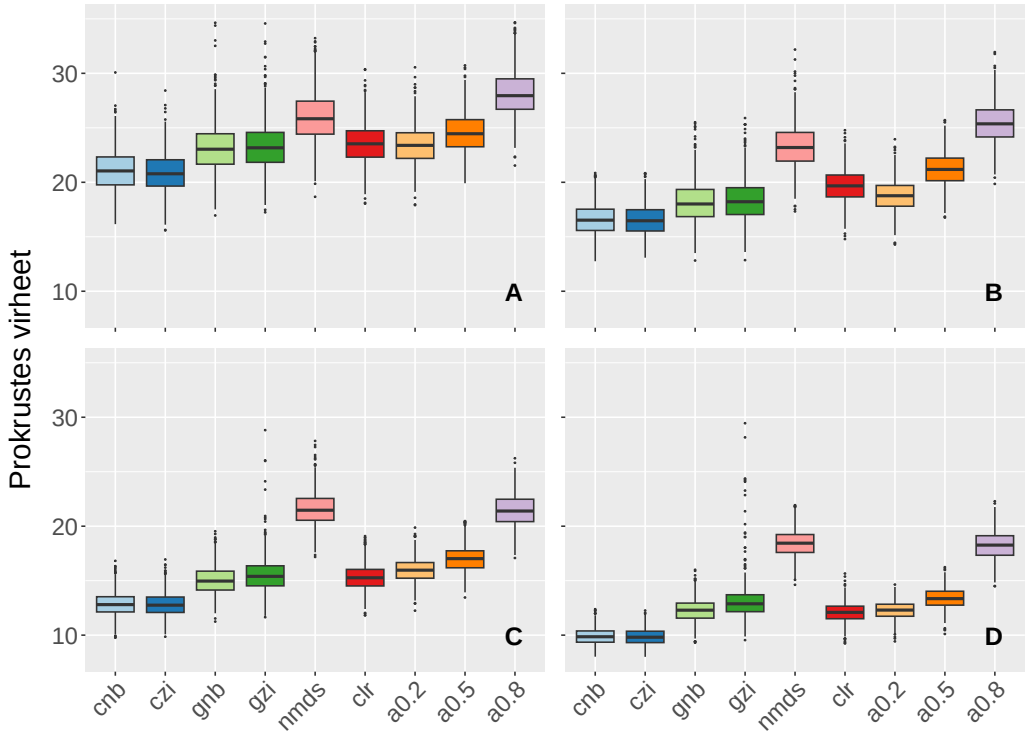
m	cnb	czi	gnb	gzi	nmds	pca
50	0.9 (3.6)	1.8 (4.0)	2.8	3.9	0.1	<0.1
100	1.2 (10.5)	4.1 (11.6)	5.3	9.1	0.1	<0.1
200	2.5 (42.2)	8.8 (48.9)	16.5	31.3	0.2	<0.1
400	4.6 (159.7)	19.3 (192.5)	53.0	99.5	0.2	<0.1



Kuvio 6. Prokrustes-virheiden viiksikuviot, kun aineistot generoitiin GLLVM-menetelmällä ja kun jakaumana käytettiin ZINB-jakaumaa. Lajien lukumäärät ovat 50 (**A**), 100 (**B**), 200 (**C**) ja 400 (**D**).

Kuviossa 7 on esitetty tulokset, kun aineistot generoitiin NB-jakaumasta kopula-menetelmällä. Menetelmistä kopula toimi parhaiten. PCA- ja GLLVM-menetelmien antamat tulokset eivät eronneet merkittävästi, kun PCA-menetelmän yhteydessä käytettiin clr-muunnosta tai α-muunnosta pienillä α-arvoilla. PCA toimi huonosti, kun $\alpha = 0.8$. Käytetyllä jakaumalla ei ollut mallipohjaisten menetelmien yhteydessä merkittävää vaikutusta tuloksiin. Menetelmistä NMDS toimi huonoiten. Kaikkien menetelmien antamat tulokset paranivat aineiston koon kasvaessa.

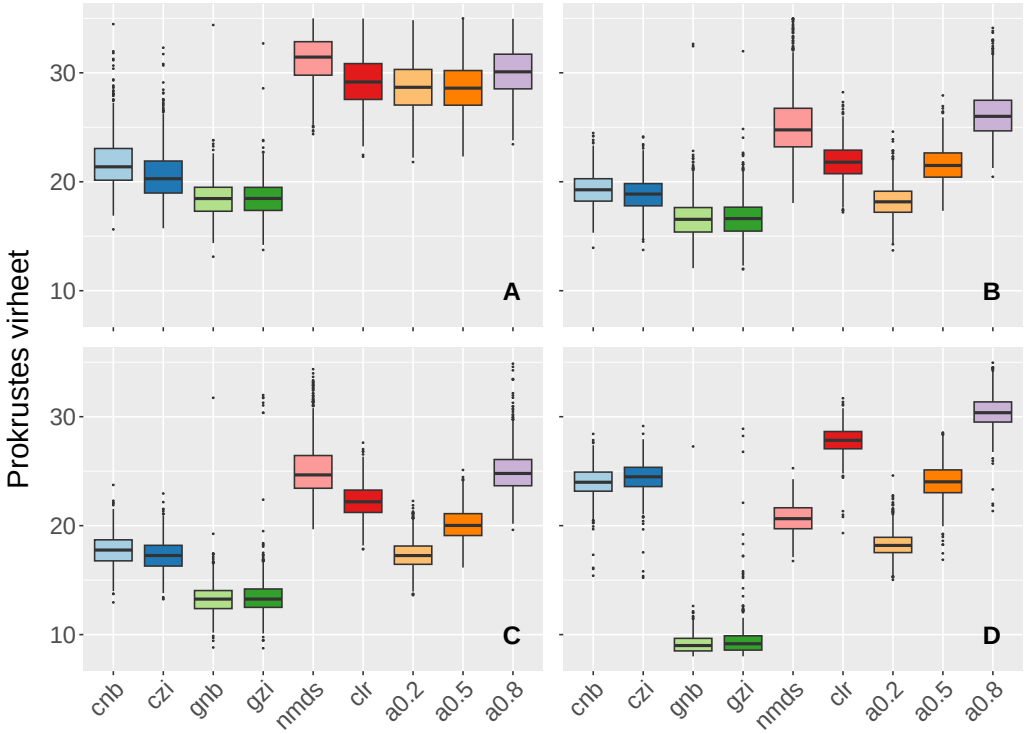
Jos verrataan kuvioissa 7 ja 5 esitettyjä tuloksia, huomataan, että GLLVM-menetelmä toimi paremmin, kun aineiston generointiin käytettiin NB-jakaumaa ZINB-jakauman sijasta kopula-menetelmän yhteydessä.



Kuvio 7. Prokrustes-virheiden viiksikuviot, kun aineistot generoitiin kopula-menetelmällä ja kun jakaumana käytettiin NB-jakaumaa. Lajien lukumäärät ovat 50 (**A**), 100 (**B**), 200 (**C**) ja 400 (**D**).

Kuviossa 8 on esitetty tulokset, kun aineistot generoitiin NB-jakaumasta GLLVM-menetelmällä. GLLVM-menetelmä toimi parhaiten, mutta käytetyllä jakaumalla ei ollut vaikutusta tuloksiin. Kopula-menetelmä toimi paremmin kuin PCA, kun lajien määrä oli 50. Kun lajien määrä oli 100 tai 200, PCA α-muunnoksella, kun $\alpha = 0.2$, toimi yhtä hyvin kuin kopula-menetelmä. NMDS-menetelmä toimi huonoiten, kun lajien määrät olivat 50, 100 tai 200, mutta kopula-menetelmä toimi huonoiten, kun aineiston koko oli 400.

GLLVM- ja NMDS-menetelmien antamat virheet pienenivät lajien määrän kasvaessa. PCA- ja kopula-menetelmien antamat virheet olivat pienimmillään, kun lajien määränä oli 200.

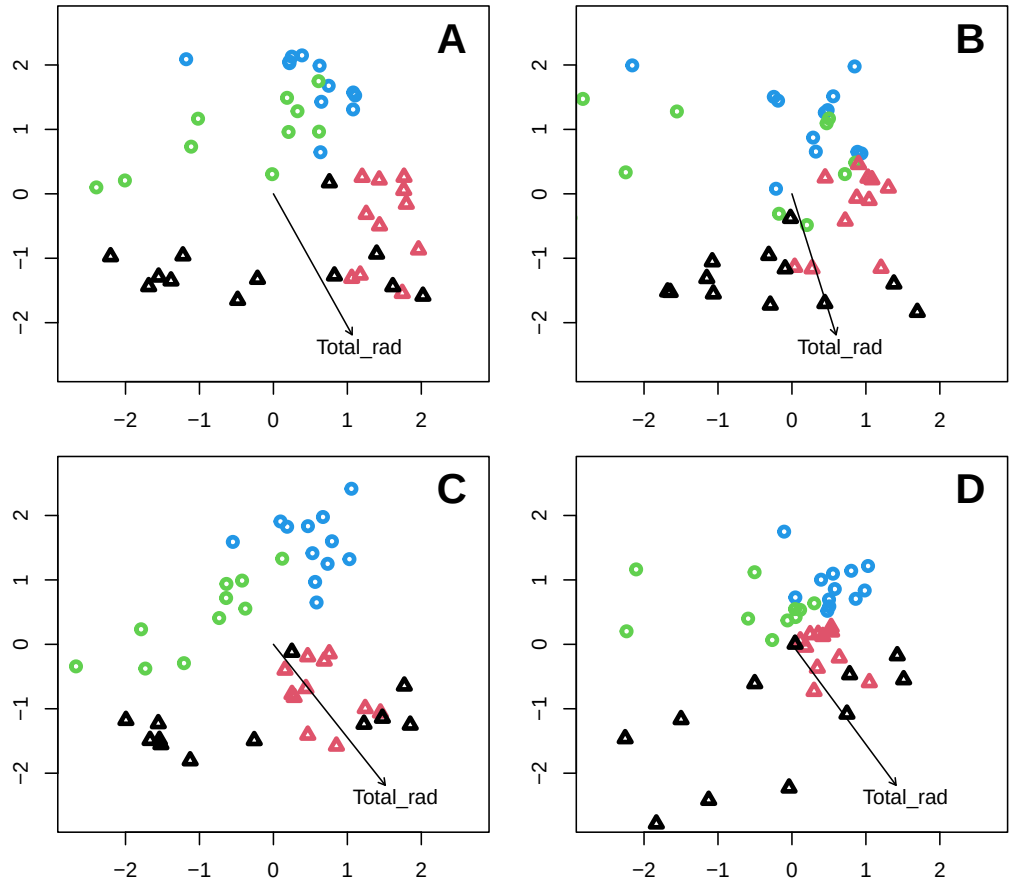


Kuvio 8. Prokrustes-virheiden viiksikuviot, kun aineistot generoitiin GLLVM-menetelmällä ja kun jakaumana käytettiin NB-jakaumaa. Lajien lukumäärät ovat 50 (**A**), 100 (**B**), 200 (**C**) ja 400 (**D**).

5 Ordinaatioesimerkki myyräaineistoon

Tässä luvussa sovelletaan klassisia ja mallipohjaisia ordinaatiomenetelmiä myyräaineistoon. Redusoidaan myyräaineisto 400 sarakkeeseen ja tehdään ordinaatio luvussa 3 esitetyillä menetelmillä.

Kuviossa 9 on eri menetelmillä saadut ordinaatiokuvat. Ordinaatiokuvien vertailun helpottamiseksi ordinaatiokuvia käännettiin Prokrustes-analyysiä käyttäen, kun kohdekonfiguraationa käytettiin kopula-ordinaatiolla saatua konfiguraatiota. Ordinaatiokuviin on myös piirretty `vegan`-paketin `envfit()`-funktiolla ordinaatioihin sovitetut kokonaissäteilyn vaikutukset.



Kuvio 9. Ordinaatioesimerkki kopula- (**A**), GLLVM- (**B**), PCA clr-muunnoksella (**C**) ja NMDS-menetelmällä (**D**). Kolmiot ovat näytteitä saastuneilta alueilta ja pallot saastumattomilta. Alueet on eroteltu väreillä. Kuviin on piirretty myös kokonaissäteilyn suhde ordinaatioakseleihin (`Total_rad`).

Kaikki menetelmät onnistuivat erottamaan saastuneet ja saastumatto-

mat alueet toisistaan, ainoastaan GLLVM-menetelmällä saadussa ordinaatiokuvassa oli hieman päällekkäisyyttä niiden välillä. Ordinaatiokuvien perusteella kopula- ja PCA-menetelmät toimivat parhaiten. Ordinaatiomenetelmät osasivat suurimmaksi osaksi erottaa näytteet myös aluetasolla. GLLVM toimi muihin menetelmiin verrattuna huonommin saastumattomien alueiden erottelemiseen, mutta saastuneissa alueissa GLLVM-menetelmällä saadussa kuvassa oli vähiten päällekkäisyyttä.

6 Pohdintaa

Tämän tutkielman tavoitteena oli vertailla klassisia ordinaatiomenetelmiä mallipohjaisiin ordinaatiomenetelmiin, tarkennettuna pääkomponenttianalyysia ja ei-metristä moniulotteista skaalausta yleistettyyn lineaariseen latenttimuuttujamalliin sekä kopula-ordinaatioon mikrobiaineistoihin sovellettuna. Menetelmiä vertailtiin simulointikokeiden avulla, joissa käytettiin pohjana oikeaa mikrobiaineistoa, jossa esiintyy mikrobiaineistoille tyypillisiä piirteitä, kuten suuriulotteisuus ja harvuus. Mielenkiinnon kohteena oli myös se, miten lajien määrä vaikuttaa menetelmien antamiin tuloksiin.

Simuloinnit tehtiin neljällä tapaa neljälle eri lajien määrälle. Aineistoja generoitiin sekä kopula- että GLLVM-menetelmällä, joissa molemmissa käytettiin erikseen NB- ja ZINB-jakaumaa. Kaikissa asetelmissa generointiin käytetty menetelmä toimi parhaiten. Pääasiassa Prokrustes-virheet pienenivät tai pysyivät samoina kaikilla menetelmillä lajien määrän kasvaessa. Ainoastaan GLLVM-menetelmällä NB-jakaumasta simuloitaessa kopula- ja PCA-menetelmien tulokset huononivat selvästi, kun lajien määrä kasvoi 200:sta 400:aan. Tulosten perusteella NMDS osoittautui huonoimmaksi menetelmäksi kaikissa asetelmissa.

Koska simulointiasetelmat suosivat generointiin käytettävää menetelmää, simulointikokeiden tulosten nojalla voi olla vaikeaa tehdä johtopäätöksiä sen suhteen, mitkä ordinaatiomenetelmät toimivat yleisesti ottaen parhaiten. Simulointien tuloksista kuitenkin ilmeni joitakin huomionarvoisia seikkoja. Esimerkiksi se, miten muunnoksen valinta vaikutti PCA-menetelmän tuloksiin. PCA-menetelmän yhteydessä mielenkiintoa herätti se, että eri menetelmillä generoitaessa eri α-arvot antoivat parempia tuloksia. Tulkinta arvolle α oli, että mitä lähempänä nollaa α-arvo on, sitä enemmän muunnos huomioi aineiston kompositionaalista luonnetta.

GLLVM-menetelmällä generoitaessa kopula-ordinaatio toimi hyvin, mutta ero PCA-menetelmään, kun käytettiin α-muunnosta, pieneni lajien lukumäärän kasvaessa. Kopula-menetelmällä generoitaessa PCA toimi paremmin

tai yhtä hyvin kuin GLLVM.

GLLVM-menetelmä on suhteellisen epävakaa aineistoilla, joissa nollien määrä on suuri. Toisin kuin kopula-menetelmä, joka sovittaa jokaiselle lajille oman regressiomallin, GLLVM estimoi kaikki parametrit samanaikaisesti.

Mikrobiaineistot voivat olla hyvinkin erilaisia. Esimerkiksi luvussa 2.2 esitetty aineisto sisälsi suuren määrän nollia. On kuitenkin aineistoja, joissa mikrobien kokonaismäärät ovat niin suuria, että nollien määrä jää pieneksi. Voisi olettaa, että mallipohjaiset GLLVM- ja kopula-menetelmät toimisivat tällaisiin aineistoihin paremmin kuin harvoihin aineistoihin.

Jatkotutkimuksissa olisi hyödyllistä tutkia tarkemmin, kuinka hyvin molemmat mallipohjaiset menetelmät pystyvät ennustamaan aineistoja, eli kuinka paljon malleilla generoidut arvot eroavat todellisista arvoista.

Simuloinneissa olisi ollut hyvä käyttää myös jotain neutraalimpaa menetelmää tai jakaumaa, joka suosisi vähemmän tiettyjä menetelmiä. Mikrobiaineistojen kompositionaalisuudesta johtuen luontainen valinta jakaumalle olisi multinomijakauma (Hawinkel, 2020). Multinomijakaumalla on kuitenkin sama ongelma kuin Poisson-jakaumalla: se ei mallinna hyvin aineistoja, joissa on ylihajontaa. Dirichlet-multinomijakauma on tässä suhteessa multinomijakaumaa parempi vaihtoehto. Dirichlet-multinomijakaumaan liittyy kuitenkin kaksi ongelmaa. Ensiksikin se ei huomioi rakenteellisia nollia, eli kaikki nollat johtuvat käytännössä vain otoskoon pienuudesta. Toiseksi, se mahdollistaa vain negatiiviset korrelaatiot lajien välillä, joka ei ole biologisille aineistoille kovin realistista. Nollapaisutettu Dirichlet-multinomijakauma voi olla parempi vaihtoehto, jos aineiston nollien määrä on todella suuri.

Viitteet

Aitchison, J. (1982). "The statistical analysis of compositional data". *Journal of the Royal Statistical Society: Series B (Methodological)* 44.2, 139–160.

Alonso, V. R. & Guarner, F. (2013). "Linking the gut microbiota to human health". *British Journal of Nutrition* 109.S2, S21–S26.

Bartholomew, D., Knott, M. & Moustaki, I. (2011). *Latent Variable Models and Factor Analysis: A Unified Approach*. Chichester: John Wiley & Sons.

Bray, J. R. & Curtis, J. T. (1957). "An ordination of the upland forest communities of southern Wisconsin". *Ecological Monographs* 27.4, 326–349.

Conesa, A., Madrigal, P., Tarazona, S., Gomez-Cabrero, D., Cervera, A., McPherson, A., Szcześniak, M. W., Gaffney, D. J., Elo, L. L., Zhang, X. ym. (2016). "A survey of best practices for RNA-seq data analysis". *Genome Biology* 17, 1–19.

Dempster, A. P., Laird, N. M. & Rubin, D. B. (1977). "Maximum likelihood from incomplete data via the EM algorithm". *Journal of the Royal Statistical Society: Series B (Methodological)* 39.1, 1–22.

Dunn, P. K. & Smyth, G. K. (1996). "Randomized quantile residuals". *Journal of Computational and Graphical Statistics* 5.3, 236–244.

Filzmoser, P., Hron, K. & Templ, M. (2018). *Applied Compositional Data Analysis*. Cham: Springer.

Gloor, G. B., Macklaim, J. M., Pawlowsky-Glahn, V. & Egozcue, J. J. (2017). "Microbiome datasets are compositional: and this is not optional". *Frontiers in Microbiology* 8, 2224.

Greenacre, M. (2017). *Correspondence Analysis in Practice*. New York: Chapman & Hall/CRC.

Greene, W. (1994). *Accounting for Excess Zeros and Sample Selection in Poisson and Negative Binomial Regression Models*. Working Paper EC-94-10. New York University.

Hawinkel, S. (2020). "Statistical Analysis of Microbiome Sequence Count Data". Tohtorinväitöskirja. Ghent University.

Hotelling, H. (1933). "Analysis of a complex of statistical variables into principal components." *Journal of Educational Psychology* 24.6, 417.

Huber, P., Ronchetti, E. & Victoria-Feser, M.-P. (2004). "Estimation of generalized linear latent variable models". *Journal of the Royal Statistical Society Series B: (Statistical Methodology)* 66.4, 893–908.

Hui, F. K., Taskinen, S., Pledger, S., Foster, S. D. & Warton, D. I. (2015). "Model-based approaches to unconstrained ordination". *Methods in Ecology and Evolution* 6.4, 399–411.

Hui, F. K., Warton, D. I., Ormerod, J. T., Haapaniemi, V. & Taskinen, S. (2017). "Variational approximations for generalized linear latent variable models". *Journal of Computational and Graphical Statistics* 26.1, 35–43.

Jernfors, T., Lavrinienko, A., Vareniuk, I., Landberg, R., Fristedt, R., Tkachenko, O., Taskinen, S., Tukalenko, E., Mappes, T. & Watts, P. C. (2024). "Association between gut health and gut microbiota in a polluted environment". *Science of the Total Environment* 914, 169804.

Korhonen, P., Hui, F. K., Niku, J. & Taskinen, S. (2023). "Fast and universal estimation of latent variable models using extended variational approximations". *Statistics and Computing* 33.1, 26.

Kruskal, J. B. (1964a). "Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis". *Psychometrika* 29.1, 1–27.

— (1964b). "Nonmetric multidimensional scaling: a numerical method". *Psychometrika* 29.2, 115–129.

Lancaster, H. (1965). "The helmert matrices". *The American Mathematical Monthly* 72.1, 4–12.

Levine, R. A. & Casella, G. (2001). "Implementations of the Monte Carlo EM algorithm". *Journal of Computational and Graphical Statistics* 10.3, 422–439.

McLaren, M. R., Nearing, J. T., Willis, A. D., Lloyd, K. G. & Callahan, B. J. (2022). "Implications of Taxonomic Bias for Microbial Differential-Abundance Analysis". *bioRxiv*, 2022–08.

Niku, J., Hui, F., Taskinen, S. & Warton, D. (2019). "gllvm: Fast analysis of multivariate abundance data with generalized linear latent variable models in R." *Methods in Ecology and Evolution* 10.12, 2173–2182.

Niku, J. (2020). "On Modeling Multivariate Abundance Data with Generalized Linear Latent Variable Models". Tohtorinväitöskirja. Jyväskylän yliopisto.

Niku, J., Warton, D. I., Hui, F. K. & Taskinen, S. (2017). "Generalized linear latent variable models for multivariate count and biomass data in ecology". *Journal of Agricultural, Biological and Environmental Statistics* 22, 498–522.

Oksanen, J., Simpson, G. L., Blanchet, F. G., Kindt, R., Legendre, P., Minchin, P. R., O'Hara, R., Solymos, P., Stevens, M. H. H., Szoecs, E., Wagner, H., Barbour, M., Bedward, M., Bolker, B., Borcard, D., Carvalho, G., Chirico, M., De Caceres, M., Durand, S., Evangelista, H. B. A., FitzJohn, R., Friendly, M., Furneaux, B., Hannigan, G., Hill, M. O., Lahti, L., McGlinn, D., Ouellette, M.-H., Ribeiro Cunha, E., Smith, T., Stier, A., Ter Braak,

C. J. & Weedon, J. (2022). *vegan: Community Ecology Package*. R package version 2.6-4. URL: https://CRAN.R-project.org/package=vegan.

Peres-Neto, P. R. & Jackson, D. A. (2001). "How well do multivariate data sets match? The advantages of a Procrustean superimposition approach over the Mantel test". *Oecologia* 129, 169–178.

Popovic, G. C., Hui, F. K. & Warton, D. I. (2022). "Fast model-based ordination with copulas". *Methods in Ecology and Evolution* 13.1, 194–202.

Popovic, G. C., Warton, D. I., Thomson, F. J., Hui, F. K. & Moles, A. T. (2019). "Untangling direct species associations from indirect mediator species effects with graphical models". *Methods in Ecology and Evolution* 10.9, 1571–1583.

R Core Team (2024). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria. URL: https://www.R-project.org/.

Sklar, A. (1959). "Fonctions de répartition à n dimensions et leurs marges". *Publications de l'Institut de Statistique de l'Université de Paris* 8, 229–231.

Spearman, C. (1904). "General intelligence, objectively determined and measured". *American Journal of Psychology* 15, 201–293.

Tsagris, M., Preston, S. & Wood, A. T. (2016). "Improved classification for compositional data using the α-transformation". *Journal of Classification* 33, 243–261.

Wall, M. E., Rechtsteiner, A. & Rocha, L. M. (2003). "Singular value decomposition and principal component analysis". Teoksessa: *A Practical Approach to Microarray Data Analysis*. Boston, MA: Springer, 91–109.

Wang, Y., Naumann, U., Eddelbuettel, D., Wilshire, J. & Warton, D. (2022). *mvabund: Statistical Methods for Analysing Multivariate Abundance Data*. R package version 4.2.1. URL: https://CRAN.R-project.org/package=mvabund.

Warton, D. I., Blanchet, F. G., O'Hara, R. B., Ovaskainen, O., Taskinen, S., Walker, S. C. & Hui, F. K. (2015). "So many variables: Joint modeling in community ecology". *Trends in Ecology & Evolution* 30.12, 766–779.

Willis, A. D. (2019). "Rarefaction, alpha diversity, and statistics". *Frontiers in Microbiology* 10, 492464.

Liitteet

Koodi kopula-ordinaatiolle `gllvm`-objektilla

```
clv <- function(data, gllvm.fam = "ZINB", reff = "fixed", lv.n = 0, dispf
   = NULL) {

 stacked_models <- gllvm(data, family = gllvm.fam, num.lv = lv.n, row.eff
     = reff, disp.formula = dispf)

 res_list <- simulate_res_S(obj = stacked_models, n.res = 500)
 res_list <- plyr::alply(res_list, 3)
 S_list <- lapply(res_list, function(x) cov(x) %>% cov2cor)

 P <- dim(S_list[[1]])[1]

 A <- factor_opt(nlv = 2,
                 S.list = S_list,
                 full = TRUE,
                 quick = FALSE,
                 nobs = nrow(data))

 Th.out <- A$theta
 Sig.out <- A$sigma
 colnames(Sig.out) <- rownames(Sig.out) <- colnames(Th.out) <- rownames(
    Th.out) <- seq(1,dim(data)[2])

 res_list.mean <- plyr::aaply(plyr::laply(res_list,function(x) x), c(2,3)
    , weighted.mean , weights = A$weights)
 Scores <- t(as.matrix(A$loadings)) %*% A$theta %*% t(res_list.mean)

 make_cordobject <- list(loadings = A$loadings,
                         scores = t(Scores),
                         sigma = Sig.out, theta = Th.out,
                         obj = stacked_models)
 class(make_cordobject) <- "coord"

 return(make_cordobject)
```

```
}



sim.co <- function (make_cordobject) {

  true.mod <- make_cordobject
  true.ords <- true.mod$scores

  if(true.mod$obj$family == "negative.binomial"){
    sig <- true.mod$sigma[[1]]
    eta <- t(replicate(nrow(as.data.frame(true.mod$obj$data)), true.mod$obj
       $params$beta0)) + true.mod$obj$params$row.params
    phi.inv <- t(replicate(nrow(as.data.frame(true.mod$obj$data)), true.mod
       $obj$params$inv.phi))
    true.load <- as.matrix(true.mod$loadings)
    Psi <- diag(diag(sig - true.load %*% t(true.load)))
    cx.z <- scale(true.ords%*%t(true.load) + rmvnorm(nrow(as.data.frame(
       true.mod$obj$data)),rep(0,ncol(as.data.frame(true.mod$obj$data))),
       Psi))
    cw.y <- qnbinom(pnorm(cx.z), mu = exp(eta), size = phi.inv)
  }

  if(true.mod$obj$family == "ZINB"){
    sig <- true.mod$sigma[[1]]
    eta <- t(replicate(nrow(as.data.frame(true.mod$obj$data)), true.mod$obj
       $params$beta0)) + true.mod$obj$params$row.params
    phi.inv <- t(replicate(nrow(as.data.frame(true.mod$obj$data)), true.mod
       $obj$params$ZINB.inv.phi))
    probs <- t(replicate(nrow(as.data.frame(true.mod$obj$data)), true.mod$
       obj$params$phi))
    true.load <- as.matrix(true.mod$loadings)
    Psi <- diag(diag(sig - true.load %*% t(true.load)))
    cx.z <- scale(true.ords%*%t(true.load) + rmvnorm(nrow(as.data.frame(
       true.mod$obj$data)),rep(0,ncol(as.data.frame(true.mod$obj$data))),
       Psi))
    cw.y <- qzinegbin(pnorm(cx.z), size = phi.inv, munb = exp(eta), pstr0 =
        probs)
  }
  return(cw.y)
}
```

```
.fix_inf <- function(mat, lim = 5) {
  mat[mat > lim] = lim
  mat[mat < (-lim)] = -lim
  mat
}


simulate_res_S <- function(obj, n.res = 500, seed = NULL) {
  single_res_fn <- function() {
    res <- residuals(obj)$residuals
    out <- .fix_inf(res)
    return(out)
  }
  out <- replicate(n.res, single_res_fn())
  set.seed(NULL)
  return(out)
}


L.icov.prop = function(S, Theta) {
  exp(-1/2 * sum(S * (Theta - diag(dim(Theta)[1]))))
}


factor_opt = function(nlv, S.list, full = FALSE, quick = FALSE, nobs) {
  P <- dim(S.list[[1]])[1]
  J <- length(S.list)
  eps <- 1e-10
  maxit <- 10
  array.S <- array(unlist(S.list), c(P, P, J))

  S.init <- cov2cor(apply(array.S, c(1, 2), mean))
  weights <- rep(1, J)/J

  A <- factanal(NA, nlv, covmat = S.init, nobs = nobs, nstart = 3)
  L <- A$loadings
  Fac <- A$factors

  Psi <- diag(diag(S.init - L %*% t(L)))
  PsiInv <- diag(1/diag(Psi))
```

```
Sest <- L %*% t(L) + Psi
Test <- solve(Sest)

Sigma.gl <- Theta.gl = list()
Sigma.gl[[1]] <- Sest
Theta.gl[[1]] <- Test
A$sigma <- Sest
A$theta <- Test

if (!(quick)) {
  count <- 1
  diff <- eps + 1
  while ((diff > eps & count < maxit) & any(!is.na(Theta.gl[[count]])))
     {

    weights <- plyr::laply(S.list, L.icov.prop, Theta = Theta.gl[[count
       ]])
    weights <- weights/sum(weights)
    count <- count + 1
    Sigma.gl[[count]] <- cov2cor(apply(array.S, c(1, 2), weighted.mean, w
        = weights))

    A <- factanal(NA, nlv, covmat = Sigma.gl[[count]], nobs = nobs,
       nstart = 3)
    L <- A$loadings
    Fac <- A$factors

    Psi <- diag(diag(S.init - L %*% t(L)))
    PsiInv <- diag(1/diag(Psi))
    Sest <- L %*% t(L) + Psi
    Test <- solve(Sest)

    Sigma.gl <- Theta.gl = list()
    Sigma.gl[[count]] <- Sest
    Theta.gl[[count]] <- Test
    A$sigma <- Sest
    A$theta <- Test
    A$weights <- weights

    if (any(!is.na(Theta.gl[[count]]))) {
      diff = sum(((Theta.gl[[count]] - Theta.gl[[count - 1]])^2)/(P^2))
```

```
    } else {
      diff = sum(((Sigma.gl[[count]] - Sigma.gl[[count - 1]])^2)/(P^2))
    }
  }

}

if (full) {
  return(A)
}
else {
  return(A$theta)
}
}
```