

Asymmetrische nabijheidsgegevens

8.1 UITBREIDINGEN VAN CMDS

In Hoofdstuk 6 is de oervorm van het MDS-probleem behandeld, dat wil zeggen, de niet-metrische analyse van één vierkante, symmetrische, tweeweg/éénmodale matrix met ordinale afstandsgegevens. Dit noemden we het klassieke MDS-probleem (CMDS). Van dit probleem zijn er verschillende varianten mogelijk:

- de matrix met proximiteiten is vierkant, maar niet symmetrisch (*asymmetrisch*), dat wil zeggen, de observaties boven de diagonaal van de matrix verschillen van de getallen die onder de diagonaal staan;
- er zijn *twee of meer* vierkante symmetrische matrices geobserveerd; er is dus sprake van een drieweg/tweemodale datamatrix of, zo men wil, een tweeweg/éénmodale matrix met replicaties;
- er zijn twee of meer vierkante, asymmetrische matrices;
- de matrix is *rechthoekig*, dat wil zeggen, niet vierkant en niet symmetrisch: we hebben dus te maken met een tweeweg/tweemodale nabijheidsmatrix;
- er zijn twee of meer rechthoekige matrices geobserveerd; het gaat dus om een drieweg/driemodaal probleem of een tweeweg/tweemodaal probleem met replicaties.

Elk van deze varianten geeft aanleiding tot een ander type MDS-analyse. De verschillende mogelijkheden voor de analyse van *één asymmetrische vierkante matrix* zullen in dit hoofdstuk behandeld worden.

8.2 ASYMMETRISCHE GELIJKENISGEGEVENS

Bij asymmetrische nabijheidsmatrices beschikken we voor elk paar objecten i en j over twee nabijheidsmaten, o_{ij} en o_{ji} , die niet aan elkaar gelijk (hoeven te) zijn. Dit is problematisch omdat beide nabijheidsmaten een schatting zijn van één en dezelfde afstand d_{ij} , die per definitie gelijk is aan d_{ji} . De vraag is dus wat we met deze tegenstrijdigheid moeten doen.

Nu zijn er in principe drie mogelijke oorzaken van asymmetrische nabijheden te onderscheiden: de asymmetrie is te wijten aan (a) toevalsfluctuaties, (b) de methode van dataverzameling, en (c) 'echte' asymmetrische relaties tussen de objecten. Van elk van deze situaties volgt hieronder een voorbeeld.

Toeval. Stel dat we een verzameling van m stimuli (bijvoorbeeld kleuren, muzieknoten, of begrippen) door een of meer proefpersonen op hun onderlinge gelijkheid laten beoordelen. Daartoe leggen we alle $m(m-1)$ geordende paren die uit de stimuli gevormd kunnen worden aan een aantal proefpersonen voor, met de vraag om aan te geven in welke mate de stimuli van elk paar op elkaar lijken. Wij vragen de proefpersonen om daarbij getallen te gebruiken die, bijvoorbeeld, van 1 (totaal verschillend) tot en met 9 (grootst mogelijke gelijkheid) lopen. Omdat we zowel paar (i,j) als paar (j,i) aanbieden, is het niet alleen mogelijk, maar zelfs zeer waarschijnlijk dat niet alle o_{ij} precies gelijk zijn aan alle o_{ji} (hoewel dit ideaal wel het geval zou moeten zijn). In dit voorbeeld zijn de metingen dus niet perfect betrouwbaar; de observaties bevatten toevallige meetfouten.

Dataverzameling. Een voorbeeld waarin de asymmetrie noodzakelijk uit de dataverzameling volgt, is de methode van *multiple rangordeningen*. Van m stimuli bieden we beurtelings één stimulus als standaard aan en vragen we de proefpersoon de overige $m-1$ stimuli te rangordenen naar hun mate van gelijkheid met de standaardstimulus. Per standaard observeren we dus steeds dezelfde rangnummers, die van 1 (meeste gelijkheid met de standaard) tot $m-1$ (minste gelijkheid met de standaard) lopen. Maken we nu een matrix O waarin o_{ij} het rangnummer is van de mate waarin stimulus j op standaard i lijkt, dan is deze matrix altijd asymmetrisch. Ook als we geen rangnummers maar andere numerieke beoordelingen verzamelen, kunnen asymmetrische datamatrices ontstaan. Bijvoorbeeld: we bieden stimulus i aan en vragen de proefpersoon 'Hoeveel van deze stimulus is aanwezig in stimulus j ' (zie Ekman, 1963). Of we verzamelen *conditionele waarschijnlijkheden* door te vragen 'Een object heeft eigenschap i , hoe groot is de kans dat dit object ook eigenschap j heeft?' (zie Hays, 1958; Van der Kloot, 1975). In zulke gevallen is het vanzelfsprekend dat er asymmetrische data geobserveerd worden, terwijl we er in principe van uit blijven gaan dat de onderliggende afstanden wel symmetrisch zijn.

Echte asymmetrie. Naast data waarin de asymmetrie aan toevalsfouten of aan de methode van dataverzameling te wijten is, komen er ook nabijheidsrelaties voor die *van nature asymmetrisch* zijn. Een voorbeeld van zo'n relatie is het aantal wetenschappelijke artikelen in tijdschrift X dat door artikelen in tijdschrift Y wordt geciteerd: artikelen in het *Journal of Experimental Psychology* worden vaker aangehaald in het *Psychological Bulletin* dan omgekeerd (zie Coombs,

1964; Coombs, Dawes & Tversky, 1970; Jaspars & Ackermans, 1965; Weeks & Bentler, 1982). Een ander voorbeeld: in een kleine groep richt een willekeurig groepslid vaker het woord tot iemand die hoger in de statushiërarchie van de groep staat dan dat deze ‘hogere’ het woord richt tot het desbetreffende groepslid. Ten derde: het aantal telefoongesprekken dat vanuit New York City met Detroit en Miami gevoerd wordt, is groter dan het aantal telefoongesprekken vanuit Detroit en Miami naar New York (zie Harshman & Lundy, 1990). Ten slotte: als we registreren hoeveel tijd het kost om per fiets heen en weer te reizen tussen een aantal steden van Nederland, dan zal men vinden dat het meer tijd kost om van Groningen naar Den Haag te rijden dan omgekeerd. Hetzelfde zal gelden voor Utrecht - Den Haag, Arnhem - Utrecht, kortom voor alle fietstochten die van (noord)oost naar (zuid)west gaan, omdat er in Nederland doorgaans een (zuid)westenwind waait. In deze voorbeelden heeft het gebrek aan symmetrie te maken met verschillen tussen de objecten in grootte, belangrijkheid, aantrekkingskracht of moeite die het kost om een bepaalde afstand te overbruggen. Er is hier dus steeds sprake van een inhoudelijke verklaring voor de asymmetrie in de data.

MDS van asymmetrische nabijheidsdata

In overeenstemming met onze assumpties over het ontstaan van asymmetrie in de data zijn er twee soorten MDS-oplossingen te onderscheiden:

- 1 standaardoplossingen, waarin wordt aangenomen dat de ‘echte’ relaties tussen de objecten wél symmetrisch zijn, zodat we een ‘gewone’ MDS-afbeelding zoeken met $d_{ij} = d_{ji}$;
- 2 oplossingen waarbij we uitgaan van ‘echte’ asymmetrie tussen de objecten, zodat we een MDS-oplossing zoeken waarin d_{ij} niet noodzakelijk gelijk d_{ji} is. We doen dat door speciale *asymmetrieparameters* in het afstandsmodel op te nemen.

Standaard MDS-oplossing. Voor de MDS van asymmetrische nabijheidsdata bestaan er verschillende manieren van aanpak. Daarin willen we de asymmetrische data zodanig analyseren dat we een standaard MDS-oplossing krijgen, dat wil zeggen, een configuratie van punten die per definitie symmetrische afstanden tot elkaar hebben. Aan de oplossing is dan niet meer te zien dat de data oorspronkelijk asymmetrisch waren; de oplossing doet dan dus niet volledig recht aan de data, omdat de asymmetrie als het ware genegeerd wordt. Deze aanpak ligt voor de hand als de asymmetrie van de data veroorzaakt is door toevalsfouten of door de methode van dataverzameling. MDS van dit soort data is nu vooral een technisch probleem, met betrekking tot de vraag: hoe moeten we de nabijheidsdata corrigeren of hoe moeten we ze in het MDS-programma invoeren om een standaard MDS-oplossing te krijgen? Hiervoor bestaat een aantal mogelijkheden:

- men kan de nabijheidsmatrix symmetrisch maken;
- men kan de complete asymmetrische matrix analyseren en een oplossing zoeken die zo goed mogelijk overeenstemt met de observaties uit *beide* helften van de matrix;

- men kan de asymmetrische matrix splitsen, door van de onderdriehoek en de bovendriehoek twee afzonderlijke matrices te maken en deze simultaan te analyseren;
- men kan de asymmetrische matrix rij-conditioneel analyseren.

Deze mogelijkheden worden in de volgende paragrafen van dit hoofdstuk nader toegelicht.

MDS met asymmetrieparameters. In het geval de objecten ‘van nature’ asymmetrische relaties hebben, is het maar de vraag of bovengenoemde aanpak de meest gewenste is. Zouden we niet liever een MDS-oplossing zien, waarin de asymmetrie van de data is terug te vinden, dat wil zeggen, een oplossing waaruit men kan afleiden dat o_{ij} niet per se gelijk aan o_{ji} is. Ook hiervoor bestaan verschillende mogelijkheden. In de eerste plaats kan men een *afstandsmodel* kiezen waarin d_{ij} niet automatisch gelijk aan d_{ji} is. In de tweede plaats kan men onderscheid maken tussen de rol die een stimulus als rij-object speelt en de rol van die stimulus als kolomobject (zie het sociometrievoorbeeld: groepsleden kunnen gezien worden als gevers en als ontvangers van interactie). In dat geval kunnen de rij- en de kolomobjecten het best als aparte punten worden weergegeven. Er is dan sprake van een *tweemodale* datamatrix. Ook deze mogelijkheden worden in de volgende paragrafen nader uitgewerkt.

Symmetrisch maken van een asymmetrische datamatrix

Een asymmetrische nabijheidsmatrix kan symmetrisch gemaakt worden door een of andere vorm van *aggregatie*, bijvoorbeeld door de observaties te middelen die boven en onder de diagonaal van de datamatrix staan. Meestal gebruikt men daarvoor het rekenkundig gemiddelde, zodat $\bar{o}_{ij} = (o_{ij} + o_{ji})/2$. In sommige gevallen, bijvoorbeeld wanneer de observaties uit frequenties of proporties bestaan, ligt het meer voor de hand om het meetkundig gemiddelde te gebruiken: $\bar{o}_{ij} = \sqrt{o_{ij} \times o_{ji}}$. De symmetrisch gemaakte matrix analyseert men vervolgens als een CMDS-probleem. Men krijgt dan een oplossing waarvan de stress aangeeft hoe goed de afstanden-in-de-configuratie overeenkomen met de geaggregeerde observaties. De stress geeft echter geen inzicht in de *fit* van de oplossing met de oorspronkelijke observaties. Deze aanpak ligt dus voor de hand, als men de asymmetrie van de data opvat als oninteressante, min of meer toevallige fluctuaties. Opgemerkt moet worden dat middelen, of het nu rekenkundig of meetkundig is, alleen zin heeft als de observaties op interval- of rationiveau gemeten zijn. Als de observaties uit rangordedata bestaan, kan middeling onzinnige resultaten opleveren, zeker als de data rijconditioneel zijn. In Hoofdstuk 9 wordt een alternatief voor het zonder meer middelen van observaties besproken.

Analyse van de complete matrix

In plaats van het analyseren van een symmetrisch gemaakte datamatrix, kan men ook proberen een (symmetrische) oplossing te krijgen die optimaal overeenstemt met de observaties uit *beide* helften van de nabijheidsmatrix. We

zoeken dus coördinaten voor object i en object j , zodanig dat de stress, waarin zowel de gekwadeerde afwijkingen $\{d_{ij} - f(o_{ij})\}^2$ als $\{d_{ji} - f(o_{ji})\}^2$ meetellen, door de bank genomen (dat wil zeggen over alle stimulusparen heen) zo klein mogelijk is. De stress van zo'n oplossing laat dus wél zien hoe goed de configuratie bij *alle* observaties past. In ALSCAL voert men deze analyse uit door de specificatie /SHAPE=ASYMMETRIC /MODEL=EUCLID te geven.

Analyse van een gesplitste matrix

Men kan de asymmetrische matrix splitsen door van de bovendriehoek en de onderdriehoek twee afzonderlijke matrices te maken. Analyseert men deze matrices in ALSCAL met /SHAPE=SYMMETRIC /CONDITION=UNCONDITIONAL /MODEL=EUCLID, dan krijgt men hetzelfde resultaat als bij de analyse die in de vorige paragraaf beschreven is. Er wordt dus één oplossing gezocht die zo goed mogelijk de observaties in beide nabijheidsmatrices beschrijft en waarbij dezelfde transformatiefunctie wordt gebruikt. Een variant hierop is analyse met /SHAPE=SYMMETRIC /CONDITION=MATRIX waardoor de data uit de bovendriehoek via een andere transformatiefunctie dan de data uit de onderdriehoek optimaal geschaald worden. Zo'n aanpak heeft alleen zin als alle data uit de onderdriehoek iets met elkaar gemeen hebben waarin ze verschillen van alle data uit de bovendriehoek (bijvoorbeeld omdat ze onder verschillende condities geobserveerd zijn). In dat geval is er dus eigenlijk sprake van drieweg/tweemodale data en niet zozeer van asymmetrische tweeweg/éénmodale observatie.

Rijconditionele analyse

Wanneer de data uit een asymmetrische nabijheidsmatrix opgevat kunnen worden als 'acties' van de rijen op de kolommen of als 'reacties' van de kolommen op de rijen, dan ligt het voor de hand om de asymmetrische nabijheidsmatrix rij-conditioneel te analyseren. Voor (rangorde)data die met de methode van multiële rangordening verzameld zijn, is dit zelfs de enig zinvolle mogelijkheid. Om een optimale overeenstemming met de afstanden-in-de-configuratie te krijgen worden binnen iedere rij de observaties dan op een andere manier getransformeerd. Wat geminimaliseerd wordt, zijn de gekwadeerde afwijkingen $\{d_{ij} - f_i(o_{ij})\}^2$ en $\{d_{ji} - f_j(o_{ji})\}^2$. In ALSCAL stuurt men deze analyse aan met de opdrachten /SHAPE=ASYMMETRIC /CONDITION=ROW /MODEL=EUCLID.

8.3

MODELLEN MET PARAMETERS VOOR ASYMMETRIE

Wanneer een inhoudelijke interpretatie van de asymmetrie gewenst is, ligt het voor de hand een MDS-model te gebruiken met speciale parameters die de asymmetrie 'verklaren' of 'modelleren'. Een van de mogelijkheden is dat we een *afstandsmode*l kiezen waarin d_{ij} niet vanzelfsprekend gelijk is aan d_{ji} . Zo'n model is het zogenaamde *gewogen-afstandsmode*l dat in ALSCAL het ASCAL-model genoemd wordt. Dit model zullen we hieronder bespreken.

Het gewogen-afstandsmodel

Een afstandsmodel waarmee men asymmetrische nabijheden kan verklaren, is het gewogen-afstandsmodel waarin

$$d_{ij} = \sqrt{\sum_s [v_{is} (x_{is} - x_{js})^2]} \quad [8.1]$$

en analoog

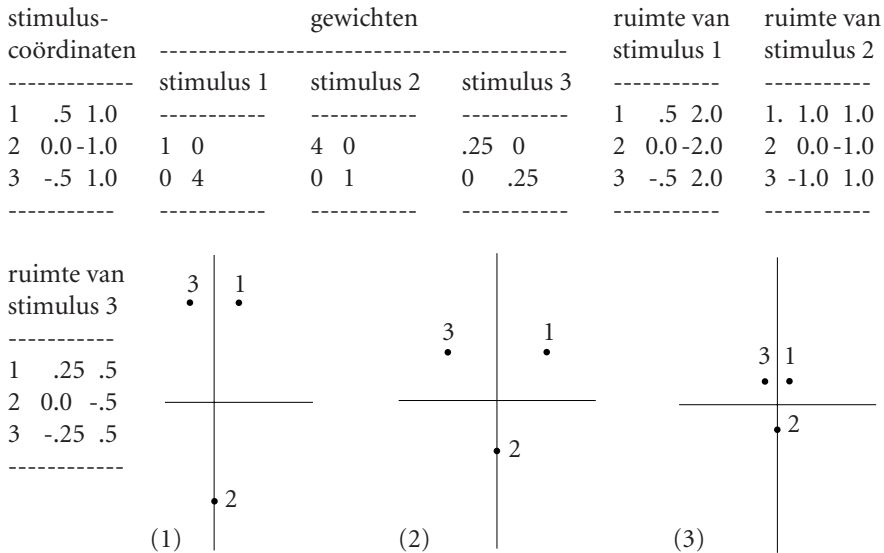
$$d_{ji} = \sqrt{\sum_s [v_{js} (x_{is} - x_{js})^2]}. \quad [8.1a]$$

Omdat v_{is} in dit model niet gelijk hoeft te zijn aan v_{js} hoeft d_{ij} niet gelijk te zijn aan d_{ji} . Bovenstaande formules kunnen in matrixnotatie weergegeven worden (zie Formule 4.4), door

$$d_{ij}^2 = (\mathbf{x}'_i - \mathbf{x}'_j) \mathbf{V}_i (\mathbf{x}'_i - \mathbf{x}'_j)' \quad [8.2]$$

\mathbf{V}_i is een diagonale matrix met evenveel rijen en kolommen als er dimensies zijn. Op de diagonaal van deze matrix staan getallen v_{is} die aangeven in welke mate het verschil tussen object i en object j op dimensie s bijdraagt aan de afstand tussen object i en object j ¹. Als $\mathbf{V}_i \neq \mathbf{V}_j$ is het alsof de punten van de configuratie op een andere plaats liggen als ze met i vergeleken worden dan als ze met j vergeleken worden. We zouden kunnen zeggen: als we in het standpunt van i gaan staan, dan ziet de configuratie van alle punten er anders uit dan wanneer we in het standpunt van j staan. Vanuit standpunt i hebben de punten in de configuratie de coördinaten $\mathbf{X}_{(i)} = \mathbf{X} \mathbf{V}_i^{1/2}$ waarbij $\mathbf{V}_i^{1/2}$ een diagonale matrix is met elementen $\sqrt{v_{is}}$. We kunnen dit dus opvatten als een (niet-afstandsbehoudende) transformatie met $\mathbf{V}_i^{1/2}$ van een gemeenschappelijke configuratie \mathbf{X} . De afstand d_{ij} wordt dus als het ware in een andere configuratie berekend dan de afstand d_{ji} . Een voorbeeld van dit model wordt weergegeven in Figuur 8.1.

1 Het symbool v_{is} is de aanduiding van het element in rij s en kolom s van matrix \mathbf{V}_i ; v_{is} is dus een verkorte schrijfwijze van $v_{ss}^{(i)}$. Omdat \mathbf{V}_i diagonaal is, kan er echter geen verwarring ontstaan.



Figuur 8.1 Voorbeeld van een afstandsmodel met asymmetriegewichten

Het berekenen van de asymmetriegewichten

Het MDS-probleem bestaat nu uit het vinden van twee soorten parameters: de coördinaten van de punten en de gewichten die bij de afzonderlijke punten horen. Om de stimulusgewichten te berekenen is in *ALSCAL*² een extra stap in het iteratieve algoritme opgenomen (zie Figuur 8.2). Deze stap komt ná het transformeren van de observaties in de optimal scaling-stap en vóór de verbetering van de configuratie in de *model estimation*-stap. Om de stimulusgewichten te bepalen maakt het algoritme gebruik van multiële-regressieanalyse. Dat gaat als volgt. Volgens Formule [8.1] geldt voor elke stimulus i dat

$$d_{ij} = \sqrt{\sum_s v_{is}(x_{is} - x_{js})^2} \text{ zodat } d_{ij}^2 = \sum_s v_{is}(x_{is} - x_{js})^2.$$

Omdat idealiter de afstanden d_{ij} gelijk zijn aan de getransformeerde observaties $f(o_{ij})$ vullen we deze in op de plaats van de d_{ij} . In het ideale geval geldt dus – bij r dimensies – dat

$$\begin{aligned} f(o_{ij})^2 &= \sum_s^r v_{is}(x_{is} - x_{js})^2 \\ &= v_{i1}(x_{i1} - x_{j1})^2 + v_{i2}(x_{i2} - x_{j2})^2 + \dots + v_{ir}(x_{ir} - x_{jr})^2. \end{aligned} \quad [8.3]$$

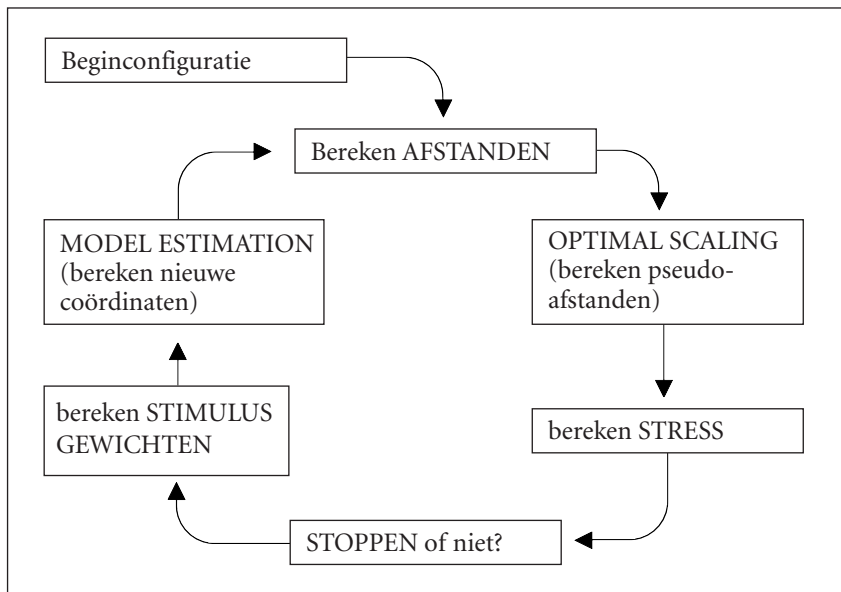
Voor ieder punt i zijn er $m - 1$ van bovenstaande vergelijkingen, zodat we voor elk afzonderlijk punt te maken hebben met een multiële-regressieprobleem waarin we de uit iteratie t bekende, gekwadrateerde pseudo-afstanden

2 *ALSCAL* is het enige computerprogramma dat de mogelijkheid biedt asymmetrische data volgens het gewogen-afstandsmodel te analyseren.

$(f(o_{ij}))^{(t)2}$ willen ‘voorspellen’ uit de eveneens bekende coördinaten $x_{is}^{(t)}$. De bijbehorende regressiecoëfficiënten zijn dan de optimale kleinste-kwadraten-schattingen van de gewichten $v_{is}^{(t)}$. Om deze gewichten te vinden moeten we voor ieder afzonderlijk punt (dus in totaal m keer) een multiële-regressieprobleem oplossen. Zijn de gewichten eenmaal gevonden, dan kunnen via Formule [8.1] weer nieuwe afstanden berekend worden. Daarmee begint het computerprogramma aan een nieuwe iteratie. Gebruikmakend van de zojuist berekende gewichten en afstanden berekent de *model estimation*-stap nieuwe coördinaten voor de stimuli. Vervolgens worden in de *optimal scaling*-stap weer nieuwe transformaties bepaald die – na berekening van de S-stress en na de beslissing om door te gaan – leiden tot verbetering van de gewichten. Daarna worden weer nieuwe coördinaten berekend, enzovoort, net zolang totdat er een optimale oplossing is gevonden. Het iteratieve algoritme dat in Figuur 8.2 is afgebeeld, bevat dus drie stappen waarin altemerend drie kleinste-kwadratenproblemen worden opgelost:

- Stap 1: ga uit van de laatste coördinaten en stimulusgewichten en bepaal de optimale transformaties van de data: de pseudo-afstanden of disparities.
- Stap 2: ga uit van de laatste coördinaten en pseudo-afstanden en bepaal de optimale stimulusgewichten.
- Stap 3: ga uit van de laatste coördinaten, pseudo-afstanden en stimulusgewichten en bepaal nieuwe, optimale coördinaten.

Een voorbeeld van een ALSCAL-analyse met dit gewogen-afstandsmodel wordt in Blok 8.1 besproken.



Figuur 8.2 Het iteratieve MDS-algoritme, met een stap voor de berekening van stimulusgewichten als asymmetrieparameters

Andere asymmetriemodellen

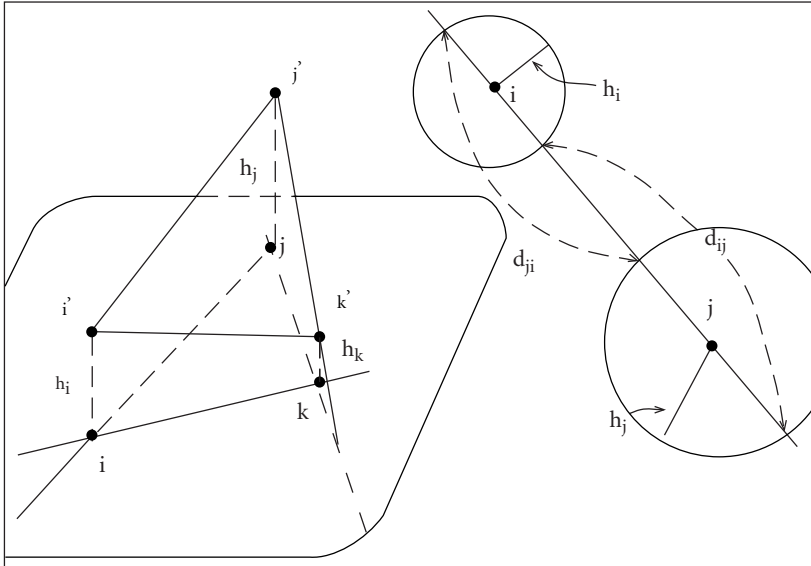
Naast het gewogen-afstandsmodel dat hierboven besproken is, bestaan er nog enkele andere modellen waarmee men speciale parameters schat die de asymmetrie in de nabijheidsdata modelleren. In al die modellen hoort bij de asymmetrische nabijheidsmatrix \mathbf{O} een eveneens asymmetrische afstandsmatrix \mathbf{D} die wordt opgesplitst in een symmetrische matrix $\mathbf{S} = (\mathbf{D} + \mathbf{D}')/2$ en een *antisymmetrische*³ matrix $\mathbf{A} = \mathbf{D} - (\mathbf{D} + \mathbf{D}')/2 = (\mathbf{D} - \mathbf{D}')/2$ (merk op: $\mathbf{D} = \mathbf{S} + \mathbf{A}$). De afstanden $\{s_{ij}\}$ in de symmetrische matrix \mathbf{S} zijn normale Euclidische afstanden en bestaan uit de gemiddelden van o_{ij} en o_{ji} . Als verklaring voor het antisymmetrische deel \mathbf{A} zijn verschillende modellen voorgesteld. Gower (1977) beschreef er drie die we respectievelijk het *bergländmodel*, het *tegenwindmodel* en het *cycloonmodel* zullen noemen.

Het bergländmodel. Dit model is een speciaal geval van een familie van modellen die door Gower (1977) zijn voorgesteld om een asymmetrische matrix $\mathbf{D} = \mathbf{S} + \mathbf{A}$ te analyseren. Dit speciale geval is nagenoeg identiek aan de asymmetriemodellen die door Weeks en Bentler (1982) en Okada (1988) beschreven zijn. Om de antisymmetrische matrix \mathbf{A} te verklaren wordt een lineair, subtractief model voorgesteld, waarin $a_{ij} = h_j - h_i$. In dit model komt het erop neer dat bij iedere symmetrische afstand s_{ij} een stukje wordt opgeteld dat gelijk is aan $(h_j - h_i)$.⁴ Het model is dus $d_{ij} = s_{ij} + h_j - h_i$. Is $h_j > h_i$ dan is dus $d_{ij} > d_{ji}$. MDS met behulp van dit model levert (a) een verzameling coördinaten $\{x_{is}\}$ op met afstanden die het symmetrische deel \mathbf{S} zo goed mogelijk verklaren en (b) een verzameling h -waarden die het antisymmetrische deel \mathbf{A} zo goed mogelijk benaderen. Daarbij kunnen we de h 's zo kiezen dat $h \geq 0$. Deze resultaten zijn samen in één grafiek af te beelden. In het tweedimensionale geval gaat dat als volgt. Met behulp van de coördinaten $\{x_{is}\}$ maakt men op de gebruikelijke wijze een 'landkaart' van de objecten, terwijl men de h -waarden als coördinaten op een derde dimensie gebruikt. Punten met een grote h -waarde liggen dus hoog boven het door \mathbf{X} opgespannen grondvlak, punten met lage h -waarden liggen laag boven dit vlak of zelfs erin (als $h = 0$). Ieder object kunnen we dus opvatten als een 'berg' of 'heuvel' in een driedimensionale ruimte. Gegeven de zwaartekracht kost het minder tijd en moeite om van boven naar beneden te gaan dan omgekeerd. Wanneer men van boven naar beneden gaat lijken de afstanden tussen twee punten dus kleiner dan wanneer men van beneden naar boven gaat: vandaar de in dit boek voorgestelde naam *bergländmodel* (zelf heeft Gower dit model geen naam gegeven). Okada noemt dit model het *radius-model* omdat we de h -waarden ook kunnen afbeelden door middel van cirkels rond de punten; punt i wordt dan omgeven door een cirkel met straal h_i , punt j heeft een cirkel met straal h_j . Trekken we nu een rechte door de punten i

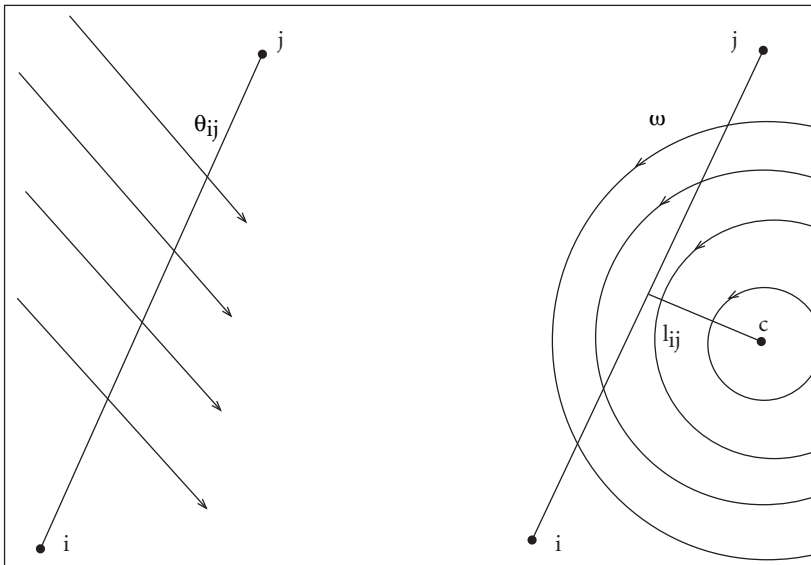
3 Een antisymmetrische matrix \mathbf{A} is een matrix waarvoor geldt dat $a_{ij} = -a_{ji}$; de corresponderende onder- en bovendagonale elementen van \mathbf{A} tellen dus op tot nul.

4 Dit is het model zoals het door Okada (1988) geformuleerd is. Bij Gower (1977) en Weeks en Bentler (1977) is het model: $a_{ij} = p_i - p_j$ zodat $p_i = -h_i$ en $p_j = -h_j$.

en j dan snijdt deze lijn beide cirkels op twee plaatsen. De nabijheid tussen punt i en j definieert Okada nu als de afstand tussen het binnenste snijpunt op de cirkel rond i en het buitenste snijpunt op de cirkel rond j . Voor de nabijheid van j tot i geldt het omgekeerde (zie Figuur 8.3).



Figuur 8.3 Het berglandmodel volgens Gower, Weeks en Bentler en Okada



Figuur 8.4 Het tegenwindmodel en het cycloonmodel

Het tegenwindmodel. Dit model, dat door Gower het *jet-stream-model* genoemd is, berust op het feit dat de tijd die het kost om met constante snelheid v van punt i naar punt j te reizen, gelijk is aan $t_{ij} = d_{ij}/v$. Waait er nu een tegenwind met windsnelheid w uit een richting die een hoek θ_{ij} met de richting van i naar j maakt (zie Figuur 8.4), dan is de reistijd van i naar j gelijk aan $t_{ij} = d_{ij}/(v - w \cos \theta_{ij})$. De reistijd van j naar i is dan $t_{ji} = d_{ij}/(v + w \cos \theta_{ij})$ zodat in het algemeen $t_{ij} \neq t_{ji}$. MDS volgens dit model komt dus neer op het schatten van (a) de coördinaten van de punten, (b) de verhouding w/v , en (c) de hoek waaruit de wind waait.

Het cycloonmodel. Ook in dit model is er sprake van een wind die sommige reistijden langer en sommige reistijden korter maakt. In dit geval gaat het om een cycloon, een wervelwind die met constante hoeksnelheid ω rond een punt C op de landkaart draait (zie Figuur 8.4). Met tegenwind uit de richting van j naar i is de reistijd van i naar j nu $t_{ij} = d_{ij}/(v - \omega l_{ij})$ en is $t_{ji} = d_{ij}/(v + \omega l_{ij})$; l_{ij} is de lengte van de loodlijn vanuit C op de lijn van i naar j . In MDS volgens dit model moeten schattingen gevonden worden voor de coördinaten van de punten, de coördinaten van C en de verhouding van v en ω .

De reden om bovengenoemde modellen hier te bespreken is dat zij drie verschillende manieren aangeven om over (het ontstaan van) asymmetrische nabijheidsdata na te denken. Het is echter niet zo dat er op dit moment algemeen bruikbare computerprogramma's beschikbaar zijn waarmee men de bijbehorende analyses zonder meer kan uitvoeren. Mogelijk zal dit in de toekomst wel het geval zijn.

Rij- en kolomobjecten als afzonderlijke punten

Zelfs al hebben de rijen en de kolommen betrekking op dezelfde objecten, toch kan men in sommige gevallen bijna niet anders dan de rij- en kolomobjecten als twee verschillende entiteiten beschouwen, dus als elementen van twee verschillende verzamelingen, twee modi. Een voorbeeld van zo'n situatie is een zogenaamde *sociomatrix*.

Een sociomatrix krijgt men door de leden van een groep (bijvoorbeeld de kinderen uit een bepaalde klas, de spelers van een hockeyteam, enzovoort) te observeren en te registreren hoe vaak groepslid i met groepslid j een interactie aangaat (wie spreekt tegen wie? Wie speelt wie de bal toe?). Ook zou men de groepsleden kunnen vragen hoe aardig zij de andere leden van de groep vinden (of: hoe graag zij met elk van de anderen naar de bioscoop zouden gaan; hoe graag zij zouden samenwerken, enzovoort). Deze gegevens kunnen we onderbrengen in een vierkante matrix, waarvan zowel de rijen als de kolommen met de leden van de groep corresponderen. In de cellen van de matrix staan dan getallen die aangeven hoe vaak de rij-persoon een interactie begint met de kolom-persoon of wat een bepaalde rij-persoon van een bepaalde kolom-persoon vindt. Deze matrix is hoogstwaarschijnlijk niet symmetrisch, omdat niet alle relaties binnen een groep wederzijds zijn (een bekend gegeven uit de groepsdynamica is dat groepsleden verschillen met betrekking tot hun status in de groep en dat sociometrische keuzen meestal gericht zijn op

personen van hogere of gelijke status, naar niet op personen van lagere status; zie Forsyth, 1990).

Omdat er inhoudelijke, zinvolle verklaringen voor de asymmetrie van de observaties zijn, zullen we hiermee in onze analyses rekening moeten houden. Een van de mogelijkheden is dat we doen alsof de rij-persoon en de kolom-persoon tot twee verschillende verzamelingen behoren: een persoon als kiezer is iemand anders, heeft een andere rol, dan een persoon als gekozen. Het kan voorkomen dat persoon i in zijn gedrag als rij-persoon veel lijkt op rij-persoon k (en dus dicht bij k afgebeeld moet worden) maar dat persoon i en persoon k als kolompersonen (dus bijvoorbeeld als ontvangers van interactie) helemaal niet op elkaar lijken. In zulke gevallen heeft het dus weinig zin om rij- en kolompersonen door middel van één verzameling punten af te beelden, ook al gaat het in wezen om dezelfde individuen. We kunnen dan beter een MDS-oplossing zoeken waarin zowel de rijen als de kolommen van de nabijheidsmatrix door afzonderlijke punten worden weergegeven. De matrix wordt dan niet langer opgevat als vierkant, maar als *rechthoekig* (in `ALSCAL: /SHAPE = RECTANGULAR`). Als we verder geen speciale eisen stellen aan de oplossing, dan is er sprake van een *klassiek ontvouwingsprobleem* (`CMDU`); op dit type analyse zullen we later, in Hoofdstuk 11, terugkomen.

Stellen we wél extra eisen aan de oplossing van bovengenoemd type matrix, dan noemen we dit *ontvouwing met restricties*. Een voorbeeld van deze aanpak is het *slide vector model* van Zielman en Heiser (1993) waarin geëist wordt dat de configuratie van kolomobjecten een zuivere *translatie* is van de configuratie van rij-objecten. Deze translatie kan beschreven worden door middel van een verschuivingsvector; conceptueel is hier dus ook sprake van een soort tegenwindmodel.

Modelkeuze

In de praktijk is het niet altijd gemakkelijk uit te maken of we een asymmetrische nabijheidsmatrix moeten opvatten als vierkant of als rechthoekig. De volgende overwegingen kunnen daarbij als leidraad dienen.

Als we een (asymmetrische) matrix als vierkante matrix analyseren, dan gaan we er impliciet van uit dat de nabijheid van object i met zichzelf groter is dan de nabijheid van object i tot elk van de overige objecten. In ons voorbeeld uit de sociometrie gaan we er dus van uit dat persoon i zichzelf minstens zo aardig vindt als alle anderen. In dat geval kunnen we i als rijpersoon best samen laten vallen met i als kolompersoon. De vraag is nu of we dat altijd zonder meer kunnen aannemen (hebben alle personen wel positieve zelfbeelden?). In dit voorbeeld hebben we daar geen empirische gegevens over. Maar als het in een bepaald onderzoek voorstelbaar is dat de nabijheid tussen i en zichzelf kleiner is dan tussen i en iemand anders, dan ligt het voor de hand rij- en kolompersonen als twee verschillende modi te beschouwen en als afzonderlijke punten af te beelden.

Er bestaan asymmetrische nabijheidsmatrices waarin gegevens over de relatie van i met zichzelf wél aanwezig zijn. Bijvoorbeeld: in een zogenaamde verwar-

ringsmatrix⁵ weten we hoe vaak een object met zichzelf verward wordt en hoe vaak met de andere objecten (hoe frequenter de verwarring, hoe kleiner de afstand). Als object i nu vaker verward wordt met een ander object dan met zichzelf, ligt het niet zonder meer voor de hand om object i als één punt (dus met $d_{ii} = 0$) af te beelden. In dat geval zouden de rijstimuli niet samen hoeven te vallen met de kolomstimuli.

In Blok 8.1 behandelen we de aansturing en resultaten van ALSCAL-analyses volgens enkele van de hierboven besproken modellen. Daarvoor nemen we weer het voorbeeld van de acht Nederlandse steden, die we in een denkbeeldige fietstocht zullen aandoen.

BLOK 8.1 NEDERLAND OP DE FIETS: WIND MEE EN WIND TEGEN

Om uit de symmetrische afstandsmatrix van onze acht Nederlandse steden een asymmetrische nabijheidsmatrix te verkrijgen hebben we schattingen gemaakt van de tijd die het zou kosten om per fiets van de ene stad naar de andere stad te reizen. Omdat de wind in Nederland doorgaans uit het (zuid)westen waait, is aangenomen dat de fietstijd van Rotterdam naar Groningen aanmerkelijk korter is dan de tijd die men nodig heeft om deze afstand in omgekeerde richting af te leggen. Deze redenering is toegepast op alle reistijden: fietsen van (zuid)west naar (noord)oost gaat sneller dan fietsen van (noord)oost naar (zuid)west. De resulterende nabijheidsmatrix staat in Tabel 8.1; deze bevat het (geschatte) aantal uren dat men doet over een fietstocht van een rij-stad naar een kolom-stad. Deze matrix is dus asymmetrisch gemaakt volgens het tegenwindmodel.

Tabel 8.1 De reistijd per fiets in uren tussen acht Nederlandse steden

	A d m	R d m	D n h	U t r	E n d	A r n	Z w o	G r o
Amsterdam	0.0	6.6	5.2	3.1	10.0	7.8	8.6	13.5
Rotterdam	5.6	0.0	1.9	3.8	9.3	9.0	10.0	16.1
Den Haag	4.4	1.8	0.0	4.7	11.0	8.9	11.1	16.9
Utrecht	3.4	5.7	5.5	0.0	7.3	4.8	6.3	13.0
Eindhoven	10.9	10.1	12.0	7.9	0.0	6.2	11.5	19.3
Arnhem	8.9	11.0	10.5	5.6	7.4	0.0	5.2	12.9
Zwolle	10.2	15.1	15.0	9.4	13.6	6.1	0.0	7.8
Groningen	20.1	24.2	25.3	19.5	22.8	15.3	9.3	0.0

5 Een recent artikel over de vraag hoe men asymmetrische verwarringsdata kan verklaren met behulp van symmetrische afstanden, is dat van Nosofsky (1991). Deze auteur maakt daarbij gebruik van de begrippen *stimulus similarity*, *stimulus bias*, *response similarity* en *response bias*.

Bovenstaande matrix is met behulp van het ALSCAL-programma op vier verschillende manieren geanalyseerd: (a) als asymmetrisch MDS-probleem waarin voor de afstanden-in-de-configuratie een optimale *fit* met de observaties in de complete matrix gezocht wordt, (b) als asymmetrisch, rijconditioneel probleem, (c) als asymmetrisch probleem met het gewogen-afstandsmodel, en (d) als rechthoekig probleem waarin elke stad door middel van twee punten wordt afgebeeld. De resultaten van deze analyses zullen we hieronder nader bekijken.

Analyse van de complete matrix

De eerste manier waarop de data uit Tabel 8.1 zijn geanalyseerd, was met behulp van de volgende ALSCAL-opdrachten.

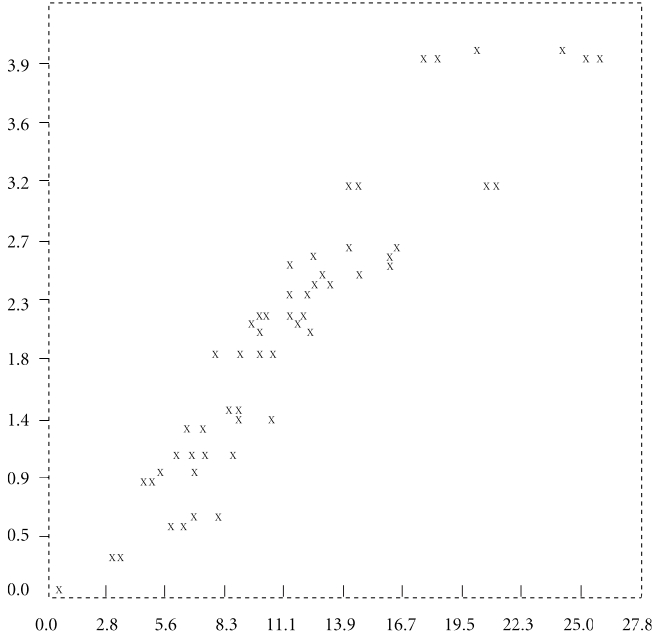
```
TITLE `REISTIJDEN TUSSEN NEDERLANDSE STEDEN PER FIETS'.
SUBTITLE `SIMULTANE ANALYSE VAN DE COMPLETE ASYMMETRISCHE MATRIX'.
ALSCAL VARIABLES=ADM RDM DNH UTR END ARN ZWO GRO
/SHAPE=ASYMMETRIC
/LEVEL=ORDINAL
/CONDITION=MATRIX
/MODEL=EUCLID
/CRITERIA DIMENSION(2) ITER(100)
CONVERGENCE(.00001) STRESSMIN(.005).
```

Deze analyse convergeerde in vijftien iteraties naar een oplossing met $S\text{-stress} = .15533$, $RSQ = .959$ en Kruskals $Stress_1 = .110$. De goodness-of-fit van deze oplossing is op het oog dus aanmerkelijk lager dan die van de symmetrische CMDS-analyse uit Blok 6.2 waarin $S\text{-stress} = .011125$, $RSQ = .999$ en $Stress_1 = .014$. Dit verschil is het gevolg van het feit dat de symmetrische afstanden d_{ij} elk met twee verschillende waarden, $f(o_{ij})$ en $f(o_{ji})$, overeen moeten stemmen. De oplossing is dus meestal een compromis, waarin d_{ij} zowel ten opzichte van $f(o_{ij})$ als ten opzichte van $f(o_{ji})$ afwijkingen heeft. Immers, ook de pseudo-afstanden, de optimaal getransformeerde observaties, zijn meestal niet symmetrisch. De afwijkingen ten opzichte van d_{ij} zijn af te lezen in het bijbehorende Shepard-diagram. Pseudo-afstanden en Shepard-diagram zijn hieronder weergegeven. De uiteindelijke configuratie lijkt echter als twee druppels water op de configuratie uit de analyse van de symmetrische afstandsmatrix.

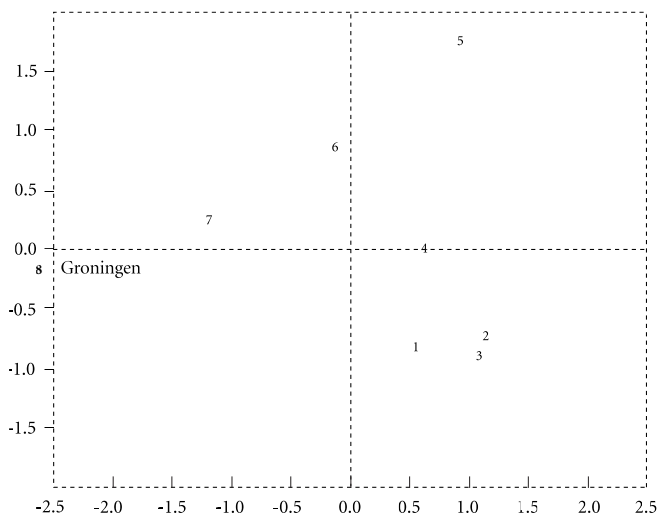
OPTIMALLY SCALED DATA (DISPARITIES) FOR SUBJECT 1

1	2	3	4	5	6	7	8
1	0.000	0.969	0.780	0.780	2.181	1.884	2.181
2	1.295	0.000	0.218	0.969	2.181	2.181	2.709
3	0.969	0.218	0.000	0.969	2.411	2.181	2.709
4	0.780	0.780	0.969	0.000	1.507	0.969	1.884
5	2.181	1.884	2.181	1.295	0.000	1.295	2.709
6	1.507	1.884	1.884	0.969	1.201	0.000	1.201
7	1.884	2.181	2.411	1.295	2.411	0.969	0.000
8	2.709	3.426	3.426	2.709	3.426	2.603	1.507

PLOT OF NONLINEAR FIT:
DISTANCES (VERTICAL) VS OBSERVATIONS (HORIZONTAL)



DERIVED STIMULUS CONFIGURATION:
DIMENSION 1 (HORIZONTAL) VS DIMENSION 2 (VERTICAL)



ADM	1	0.4655	-0.8121
RDM	2	1.0255	-0.7280
DNH	3	0.9691	-0.9382
UTR	4	0.5376	0.0146
END	5	0.8723	1.3696
ARN	6	-0.0904	0.8678
ZWO	7	-1.2502	0.2688
GRO	8	-2.5295	-0.0426

Rijconditionele analyse

Met onderstaande opdrachten werd een rijconditionele analyse van Tabel 8.1 verkregen.

```

TITLE 'REISTIJDEN TUSSEN NEDERLANDSE STEDEN PER FIETS'.
SUBTITLE 'RIJCONDITIONELE ANALYSE VAN DE ASYMMETRISCHE MATRIX'.
ALSCAL VARIABLES=ADM RDM DNH UTR END ARN ZWO GRO
/SHAPE=ASYMMETRIC
/LEVEL=ORDINAL
/CONDITION=ROW
/MODEL=EUCLID
/CRITERIA DIMENSION(2) ITER(100)
CONVERGENCE(.00001) STRESSMIN(.005).

```

Deze analyse leidde in 45 iteraties tot een oplossing met een *S-stress* die kleiner dan .005 was. Omdat de data voor iedere rij afzonderlijk een eigen transformatiefunctie krijgen, is het interessant om voor iedere rij apart een stresswaarde te berekenen. ALSCAL geeft dan ook per rij de waarden van Kruskals $Stress_i$ weer. Door deze waarden te kwadrateren en te middelen en vervolgens de wortel uit dit gemiddelde te trekken verkrijgt men één waarde voor $Stress_j$ voor de totale oplossing (de zogenaamde *root mean square*). Deze grootheden zijn terug te vinden in de ALSCAL-uitvoer die hieronder is weergegeven.

ITERATION	S-STRESS	IMPROVEMENT
1	0.04206	
2	0.03203	0.01003
3	0.02605	0.00598
.
4 5	0.00507	0.00009
4 6	0.00498	0.00009

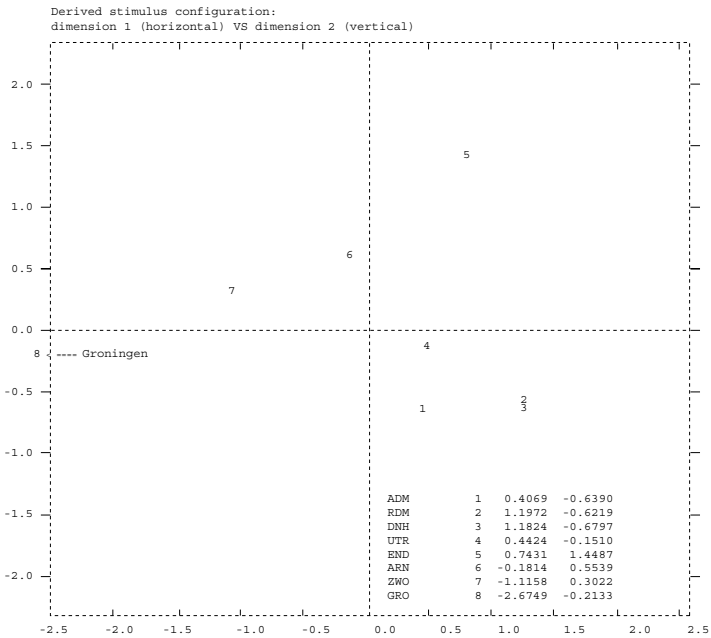
ITERATIONS STOPPED BECAUSE S-STRESS LESS THAN 0.005000
STRESS VALUES ARE KRUSKAL'S STRESS FORMULA 1.

STIMULUS	STRESS	RSQ	TIMULUS	STRESS	RSQ	STIMULUS	STRESS	RSQ
1	0.000	1.000	2	0.013	1.000	3	0.000	1.000

STIMULUS	STRESS	RSQ	STIMULUS	STRESS	RSQ	STIMULUS	STRESS	RSQ
4	0.009	1.000	5	0.001	1.000	6	0.006	1.000

STIMULUS	STRESS	RSQ	STIMULUS	STRESS	RSQ
7	0.003	1.000	8	0.001	1.000

AVERAGED (RMS) OVER STIMULI: STRESS = 0.006 RSQ = 1.000



Bovenstaande figuur is vrijwel identiek aan de configuratie van de matrix-conditionele analyse. De stress is echter veel lager, omdat de observaties van iedere rij afzonderlijk getransformeerd worden. Deze betere *fit* is te zien aan de waarden van de pseudo-afstanden. Ook in dit geval zijn die asymmetrisch, maar de verschillen tussen de corresponderende waarden onder en boven de diagonaal zijn nu kleiner dan in de eerste analyse.

OPTIMALLY SCALED DATA (DISPARITIES) FOR SUBJECT 1									
	1	2	3	4	5	6	7	8	
1	0.000	0.842	0.777	0.489	2.117	1.330	1.790	3.114	
2	0.790	0.000	0.060	0.914	2.117	1.826	2.495	3.889	
3	0.777	0.060	0.000	0.914	2.179	1.826	2.495	3.889	
4	0.489	0.842	0.909	0.000	1.628	0.955	1.632	3.114	
5	2.115	2.120	2.173	1.628	0.000	1.287	2.184	3.801	
6	1.330	1.812	1.839	0.914	1.287	0.000	0.968	2.609	
7	1.790	2.491	2.499	1.623	2.179	0.955	0.000	1.642	
8	3.111	3.894	3.885	3.118	3.801	2.609	1.632	0.000	

Analyse met het gewogen-afstandsmodel

```

TITLE `REISTIJDEN TUSSEN NEDERLANDSE STEDEN PER FIETS'.
SUBTITLE `ANALYSE MET HET GEWOGEN-AFSTANDSMODEL'.
ALSCAL VARIABLES=ADM RDM DNH UTR END ARN ZWO GRO
/SHAPE=ASYMMETRIC
/LEVEL=ORDINAL
/CONDITION=MATRIX
/MODEL=ASCAL
/CRITERIA DIMENSION(2) ITER(100)
CONVERGENCE(.00001) STRESSMIN(.005).

```

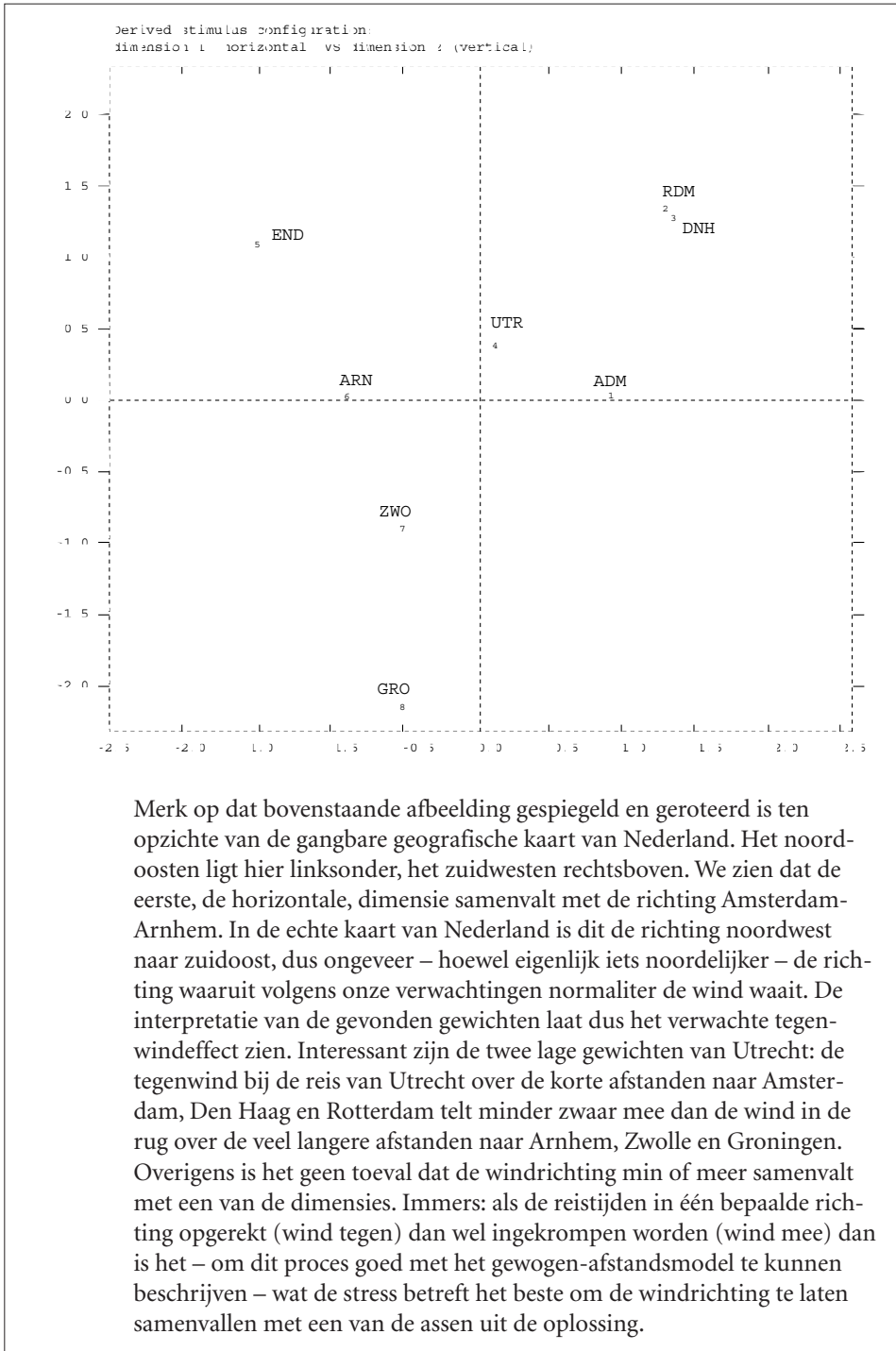
Deze analyse levert na honderd iteraties een oplossing met $S\text{-stress} = .0167$, $RSQ = .999$ en $Stress_1 = .021$, hoewel daarbij opgemerkt moet worden dat het algoritme na honderd iteraties nog niet geconvergeerd is. Dat betekent dat het verschil tussen $S\text{-stress}^{(t=99)}$ en $S\text{-stress}^{(t=100)}$ groter is dan het convergentie criterium van .00001 (het verschil bedraagt .00006). Hoewel het gevaarlijk kan zijn een oplossing te gebruiken die nog niet gestabiliseerd is, lijkt dat gevaar in dit voorbeeld niet zo groot. Immers de $S\text{-stress}$ is al vanaf de vijf-en-vijftigste iteratie kleiner dan .02, zodat we vanaf dat moment al een oplossing met een zeer goede *fit* hebben. Het is dus niet zo

heel erg interessant of het algoritme na nog veel meer iteraties een nog veel lagere *S-stress* weet te bereiken (aan de andere kant: het algoritme kan bezig zijn op een gedegenereerde oplossing af te stevenen; zie Van der Kloot, 1993).

Het bijzondere van een analyse met /MODEL=ASCAL is dat er voor iedere stad niet alleen coördinaten berekend worden, maar ook zogenaamde stimulusgewichten. Deze coördinaten en gewichten staan in onderstaande ASCAL-uitvoer.

STIMULUS COORDINATES:				STIMULUS WEIGHTS:	
		DIMENSIONS		DIMENSIONS	
STIMULUS	NAME	1	2	1	2
1	ADM	0.9186	-0.0027	0.4632	0.5081
2	RDM	1.2482	0.8196	0.3373	0.4734
3	DNH	1.3025	0.7661	0.4569	0.4617
4	UTR	0.1070	0.3654	0.4254	0.3021
5	END	-1.5723	1.0687	0.4977	0.4407
6	ARN	-0.8969	0.0403	0.7506	0.4125
7	ZWO	-0.5333	-0.9004	1.4661	0.5339
8	GRO	-0.5737	-2.1572	1.5188	0.7536

Laten we eens kijken naar de gewichten op de eerste dimensie. Wat we al direct aan die gewichten kunnen zien is dat vanuit Amsterdam de afstanden langs de eerste dimensie (veel) korter zijn dan vanuit Groningen omdat het gewicht van Amsterdam ($v_{AMS,1} = .4632$) kleiner is dan dat van Groningen ($v_{GRO,1} = 1.5188$). Kijken we naar de gewichten op de tweede dimensie, dan zien we soortgelijke verschillen die echter minder uitgesproken zijn. Om de gewichten inhoudelijk te kunnen interpreteren hebben we natuurlijk een afbeelding van de configuratie nodig: we moeten namelijk weten hoe de eerste en tweede dimensie van de oplossing er precies uitzien.



Ten slotte: de kaart van Nederland ziet er in bovenstaande figuur iets ronder uit dan in werkelijkheid. Dat komt omdat de ALSCAL-oplossing zodanig gestandaardiseerd is dat voor iedere dimensie de som van de gekwadrateerde coördinaten gelijk is aan m , het aantal objecten (hier dus acht). Dit wordt weer 'goedgemaakt' door de gewichten: de verhouding van de gewichten van de horizontale as en die van de verticale as is gemiddeld zodanig dat de kaart van Nederland weer normale proporties krijgt, als de coördinaten vermenigvuldigd worden met de wortel uit de stimulusgewichten.

Tweemodale analyse

Ten slotte volgen hieronder de resultaten van een ALSCAL-analyse waarin elk van de steden door middel van twee punten is afgebeeld. De rij-steden, dat wil zeggen, de steden van waaruit men vertrekt, krijgen een andere positie in de landkaart dan de steden waar men naartoe fietst, ook al gaat het om dezelfde stad. Dus: Amsterdam-als-bestemming heeft een andere positie dan Amsterdam-als-vertrekpunt. De aansturing van deze analyse is:

```
TITLE `REISTIJDEN TUSSEN NEDERLANDSE STEDEN PER FIETS'.
SUBTITLE `TWEEMODALE ANALYSE VAN DE ASYMMETRISCHE MATRIX'.
ALSCAL VARIABLES=ADM RDM DNH UTR END ARN ZWO GRO
      /SHAPE=RECTANGULAR /LEVEL=ORDINAL
      /CONDITION=ROW /MODEL=EUCLID
      /CRITERIA DIMENSION(2) ITER(100)
      CONVERGENCE(.00001) STRESSMIN(.005).
```

Deze analyse is een voorbeeld van een klassiek ontvouwingsprobleem (classical multidimensional unfolding; CMDU). In plaats van de optie /CONDITION=R O W, die noodzakelijk is bij de ontvouwning van (preferentie) rangordeningen (zie Hoofdstuk 11), hadden we hier ook de optie /CONDITION=MATRIX kunnen gebruiken. In het algemeen geeft een matrixconditionele analyse (als hij tenminste mogelijk is) een betrouwbaarder oplossing dan een rijconditionele. Een rij-conditionele analyse kan gemakkelijk *degenereren* (zie Hoofdstuk 11). In dit voorbeeld is dat overigens niet gebeurd. Delen van de ALSCAL-uitvoer zijn hieronder weergegeven.

HISTORY FOR THE 2 DIMENSIONAL SOLUTION (IN SQUARED DISTANCES)
YOUNGS S-STRESS FORMULA 2 IS USED.

ITERATION	S-STRESS	IMPROVEMENT
1	0.04378	
2	0.03910	0.00467
3	0.03562	0.00349
..
50	0.00946	0.00009
..
99	0.00751	0.00002
100	0.00749	0.00002

ITERATIONS STOPPED BECAUSE THIS IS ITERATION 100

STRESS AND SQUARED CORRELATION (RSQ) IN DISTANCES
STRESS VALUES ARE KRUSKAL'S STRESS FORMULA 2.

MATRIX 1 (ROW STIMULI ONLY)

STIMULUS	STRESS	RSQ	STIMULUS	STRESS	RSQ	STIMULUS	STRESS	RSQ
1	0.055	0.997	2	0.067	0.996	3	0.063	0.996

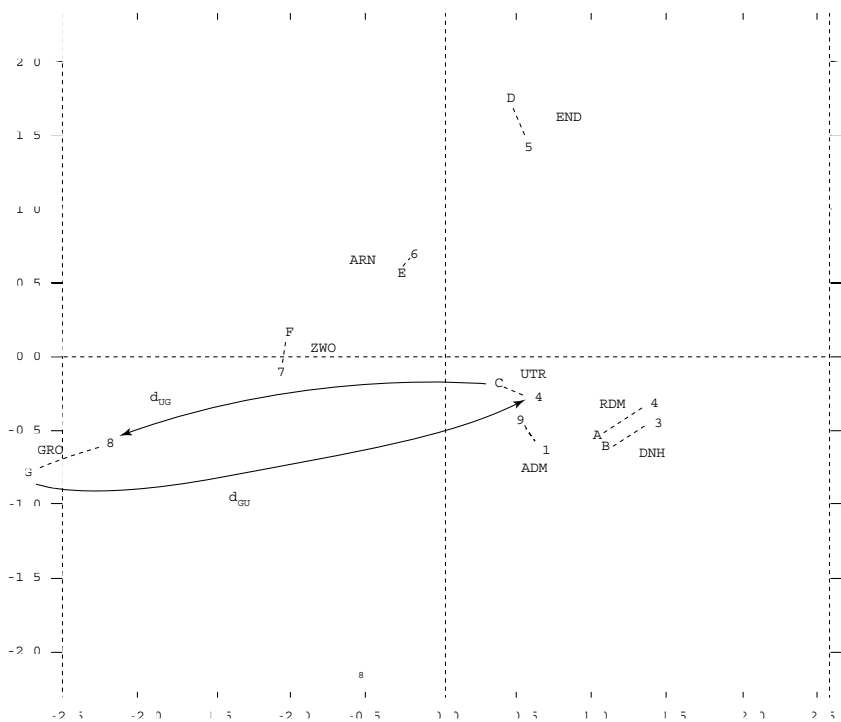
STIMULUS	STRESS	RSQ	STIMULUS	STRESS	RSQ	STIMULUS	STRESS	RSQ
4	0.062	0.996	5	0.072	0.995	6	0.041	0.998

STIMULUS	STRESS	RSQ	STIMULUS	STRESS	RSQ
7	0.065	0.996	8	0.119	0.988

AVERAGED (RMS) OVER STIMULI: STRESS = 0.071 RSQ = 0.995

Ook in dit geval is het algoritme na honderd iteraties nog niet geconvergeerd. Ook nu zien we echter dat, in termen van s-stress, de oplossing al na zo'n vijftig iteraties zeer goed te noemen is. Het gebrek aan convergentie heeft in dit geval dus meer te maken met ons strenge convergentie criterium (aan de andere kant: het algoritme zou op weg kunnen zijn naar een gedegenerende oplossing). Omdat de analyse rij-conditioneel is uitgevoerd, drukt ALSICAL ook stresswaarden voor de afzonderlijke rijen af. We zien dat vooral Groningen en Eindhoven een slechtere *fit* hebben dan de overige steden. Waarschijnlijk is dat het gevolg van het feit dat de wegen van Groningen naar het westen met een omweg langs het IJsselmeer lopen en dat de wegen vanuit het westen naar Eindhoven met een omweg via

CONFIGURATION DERIVED IN 2 DIMENSIONS									
COLUMN (de steden als bestemming)					ROW (de steden als vertrekpunt)				
STIMULUS	STIMULUS	PLOT	DIMENSION		STIMULUS	PLOT	DIMENSION		
NUMBER	NAME	SYMBOL	1	2	NAME	SYMBOL	1	2	
1	ADM	1	0.6333	-0.620	AM	9	0.4853	-0.3948	
2	RDM	2	1.3989	-0.4247	RDM	A	1.0119	-0.5006	
3	DNH	3	1.3979	-0.4372	DNH	B	1.0488	-0.5949	
4	UTR	4	0.6178	-0.209 9	UIR	C	0.3461	-0.1768	
5	END	5	0.5442	1.3807	END	D	0.43553	1.7249	
6	ARN	6	-0.2284	0.6837	ARN	E	-0.29623	0.5477	
7	ZWO	7	-1.1037	-0.0560	ZWO	F	-1.06033	0.1871	
8	GRO	8	-2.2397	-0.5471	GRO	G	-2.99153	-0.5621	



Rotterdam (eventueel via Utrecht) gaan. De corresponderende afstanden zijn dus geen afstanden in vogelvlucht en passen daarom minder goed in de ruimte. Het meest interessante aan deze oplossing is de manier waarop de asymmetrie wordt afgebeeld. Onderstaande figuur en de bijbehorende coördinaten laten zien dat iedere stad door twee punten is afgebeeld: een vertrekpunt en een bestemmingspunt.

We zien nu dat een stad-als-bestemming wel steeds dicht bij dezelfde stad-als-vertrekpunt ligt, maar dat de vertrekpunten bijna allemaal meer naar

links liggen dan de bestemmingspunten. Dat betekent dat als men van links naar rechts gaat de afstanden langer zijn, terwijl van rechts naar links de afstanden korter zijn. Vergelijk bijvoorbeeld de afstand van Utrecht naar Groningen en de afstand van Groningen naar Utrecht. Deze afbeelding klopt weer met het tegenwindmodel, omdat de horizontale as nagenoeg samenvalt met de richting van zuidwest (rechts) naar noordoost (links), de richting waaruit meestal de wind waait.