G. Rohwer, C. Dudel Wintersemester 2010/11

Methoden der Datenrepräsentation und Klassifikation

Kapitel 3: Multidimensionale Skalierung

3 Multidimensionale Skalierung

3.1 Konfigurationen
1. Konfigurationen und Abstände
2. Ein zweidimensionales Beispiel
3. Transformation von Konfigurationen
4. Prokrustes-Rotation

3.2 Metrische MDS-Verfahren
1. Die Problemstellung
2. Alternative Problemformulierungen
3. Rechentechnische Probleme
4. Illustration mit Klausurdaten
5. Metrische MDS mit R
6. Illustration mit Berufsstrukturdaten
7. Das Shepard-Diagramm

3.3 Nichtmetrische MDS-Verfahren
1. Die Problemstellung
2. Berechnungsmethoden
3. Nichtmetrische MDS mit R
4. Illustration mit Berufsstrukturdaten

3.4 Informationsgehalt von MDS-Bildern
1. Ergänzungen der MDS-Bilder
2. Illustration mit Schulabschlüssen
3. Konstruktion ergänzender Achsen

In diesem Kapitel beginnen wir, uns mit Methoden der Datenrepräsentation zu beschäftigen, die räumliche Darstellungen intendieren. Damit sind Darstellungen gemeint, bei denen sich durch räumliche Abstände in einem Bild Hinweise auf Eigenschaften der für die Darstellung verwendeten Daten gewinnen lassen. Die Methoden, um solche räumlichen Darstellungen zu erzeugen, sind sehr vielfältig. In diesem Kapitel besprechen wir Methoden, die von einer gegebenen Abstandsmatrix ausgehen. *Multidimensionale Skalierung* (MDS) wird als Sammelbegriff für diese Verfahren verwendet, die allgemein dem Zweck dienen, Abstände, die durch eine Abstandsmatrix gegeben sind, durch räumliche Abstände zu repräsentieren.[1] Meistens wird ein Zahlenraum mit einer euklidischen Metrik verwendet; indem man sich dann auf zwei (oder maximal drei) Dimensionen beschränkt, können räumliche Bilder erzeugt werden. In diesem Kapitel besprechen wir zwei Ansätze:

[1] Es gibt eine umfangreiche Literatur, u.a. Kruskal und Wish (1978); Young und Hamer (1987); Borg und Lingoes (1987); Cox und Cox (1994). Speziell mit Marketing-Anwendungen beschäftigen sich Green, Carmone und Smith (1989).

- Metrische MDS, bei der versucht wird, die Punkte im Zahlenraum so zu bestimmen, dass ihre Abstände möglichst weitgehend der vorausgesetzten Abstandsmatrix entsprechen.
- Nichtmetrische MDS, bei der versucht wird, Abstände so zu konstruieren, dass ihre ordinalen Beziehungen möglichst weitgehend denjenigen in der vorausgesetzten Abstandsmatrix entsprechen.

Eine verwandte Methode ist die multidimensionale Skalierung mit Hauptkoordinaten, die auf Überlegungen der linearen Algebra zur Zerlegung von Matrizen beruht. Diesen Ansatz besprechen wir in Abschnitt **??**.

Da in allen Ansätzen der multidimensionalen Skalierung das Ziel darin besteht, eine *Konfiguration* von Punkten in einem Zahlenraum zu finden, beginnen wir mit einigen Bemerkungen zu diesem Begriff. In den dann folgenden Abschnitten orientieren wir uns an der Frage, wie zweidimensionale Bilder erzeugt werden können. Eindimensionale Skalierung bildet einen Sonderfall und wird erst im nächsten Kapitel behandelt.

3.1 Konfigurationen

1. Konfigurationen und Abstände

Unter einer *Konfiguration* verstehen wir eine Menge von n Punkten in einem Zahlenraum. Als Zahlenraum wird im Folgenden stets $\mathbf{R}^p$ mit einer beliebigen Dimension p verwendet ($\mathbf{R}$ bezeichnet die Menge der reellen Zahlen). Eine Konfiguration besteht dann aus n Vektoren: $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbf{R}^p$. Wir folgen der Konvention, Vektoren stets als Spaltenvektoren aufzufassen, also kann jeder Vektor in der Form $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})'$ geschrieben werden. Die aus den n Vektoren bestehende Konfiguration kann auch in einer (n, p)-Matrix

$$\mathbf{X} = \begin{pmatrix} x_{11} & \ldots & x_{1p} \\ \vdots & & \vdots \\ x_{n1} & \ldots & x_{np} \end{pmatrix} = \begin{pmatrix} \mathbf{x}_1' \\ \vdots \\ \mathbf{x}_n' \end{pmatrix}$$

dargestellt werden. Die Zeilen enthalten die Punkte der Konfiguration.

Für die Punkte einer Konfiguration können auf unterschiedliche Weisen Abstände definiert werden. Insbesondere können euklidische Abstände verwendet werden:

$$\| \mathbf{x}_i - \mathbf{x}_j \| = \left(\sum_{k=1}^{p} (x_{ik} - x_{jk})^2 \right)^{1/2}$$

Für die quadrierten Abstände gilt:

$$\| \mathbf{x}_i - \mathbf{x}_j \|^2 = (\mathbf{x}_i - \mathbf{x}_j)'(\mathbf{x}_i - \mathbf{x}_j) = \mathbf{x}_i'\mathbf{x}_i + \mathbf{x}_j'\mathbf{x}_j - 2\mathbf{x}_i'\mathbf{x}_j$$

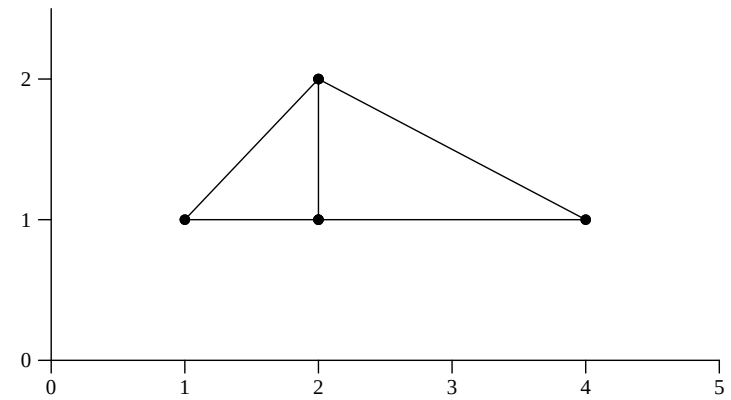

Abb. 3.1-1 Vier Punkte in einem zweidimensionalen Koordinatensystem.

2. Ein zweidimensionales Beispiel

Abbildung 3.1-1 zeigt zur Illustration eine Konfiguration, die aus vier Punkten in einem zweidimensionalen Zahlenraum besteht. Sie kann durch eine Matrix

$$\mathbf{X} = \begin{pmatrix} 1 & 1 \\ 2 & 1 \\ 4 & 1 \\ 2 & 2 \end{pmatrix} \tag{3.1}$$

erfasst werden. Verwendet man euklidische Abstände, erhält man die Abstandsmatrix

$$\mathbf{D} = \begin{pmatrix} 0 & 1 & 3 & \sqrt{2} \\ 1 & 0 & 2 & 1 \\ 3 & 2 & 0 & \sqrt{5} \\ \sqrt{2} & 1 & \sqrt{5} & 0 \end{pmatrix} \tag{3.2}$$

Die einzelnen Einträge entsprechen in diesem Fall den Längen der in Abbildung 3.1-1 eingezeichneten Verbindungslinien zwischen den Punkten.

3. Transformation von Konfigurationen

Eine Konfiguration kann einer Reihe von Transformationen unterzogen werden, ohne dass sich an den euklidischen Abständen zwischen ihren Punkten etwas ändert. Würde man beispielsweise alle Punkte in Abbildung 3.1-1 um eine Einheit nach rechts verschieben, würden die Längen der eingezeichneten Verbindungslinien gleich bleiben. Man unterscheidet drei Arten von Transformationen, die auch miteinander kombiniert werden können.

- Unter einer *Translation* versteht man, dass zu allen Punkten einer Konfiguration der gleiche Vektor addiert wird. Offenbar gilt

$$\| (\mathbf{x}_i + \mathbf{a}) - (\mathbf{x}_j + \mathbf{a}) \| = \| \mathbf{x}_i - \mathbf{x}_j \|$$

  wobei $\mathbf{a}$ ein beliebiger Vektor ist.

- Wenn man alle Punkte einer Konfiguration mit einer Zahl multipliziert, gilt für die euklidischen Abstände:

$$\| (\alpha\mathbf{x}_i) - (\alpha\mathbf{x}_j) \| = |\alpha| \, \| \mathbf{x}_i - \mathbf{x}_j \|$$

  Offenbar verändern sich die Abstände nicht, wenn $\alpha = -1$ ist; man spricht dann von einer *Reflexion*.

- Schließlich verändern sich die euklidischen Abstände auch dann nicht, wenn man eine Konfiguration rotiert. Mathematisch kann dass durch die Multiplikation der Punkte mit einer orthogonalen Matrix ausgedrückt werden: $\mathbf{x}_i^* := \mathbf{S}\mathbf{x}_i$.[2] Man erkennt:

$$\begin{aligned} \| \mathbf{x}_i^* - \mathbf{x}_j^* \|^2 = \| \mathbf{S} (\mathbf{x}_i - \mathbf{x}_j) \|^2 &= (\mathbf{x}_i - \mathbf{x}_j)' \mathbf{S}' \mathbf{S} (\mathbf{x}_i - \mathbf{x}_j) \\ &= (\mathbf{x}_i - \mathbf{x}_j)' (\mathbf{x}_i - \mathbf{x}_j) = \| \mathbf{x}_i - \mathbf{x}_j \|^2 \end{aligned}$$

Zur Illustration transformieren wir die vier Punkte der Konfiguration aus § 2 durch $\mathbf{x}_i^* = \mathbf{S}\mathbf{x}_i + \mathbf{a}$, wobei

$$\mathbf{S} = \begin{pmatrix} 0.866 & 0.500 \\ -0.500 & 0.866 \end{pmatrix} \quad \text{und} \quad \mathbf{a} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

ist. Durch Nachrechnen erkennt man, dass $\mathbf{S}$ orthogonal ist; diese Matrix entspricht einer Drehung um 30°.[3] Abbildung 3.1-2 zeigt gestrichelt die transformierte Konfiguration. Hier ist leicht ersichtlich, dass die Abstandsmatrix (3.2) auch für die transformierte Konfiguration gilt.

4. Prokrustes-Rotation

Als Ergebnis kann festgehalten werden, dass Konfigurationen, die ausgehend von Abständen konstruiert werden, nicht eindeutig bestimmt sind. Orientiert man sich an euklidischen Abständen, können die konstruierten Konfigurationen beliebigen Translationen, Rotationen und Reflexionen unterzogen werden.

[2] Eine quadratische Matrix $\mathbf{S}$ wird *orthogonal* genannt, wenn gilt: $\mathbf{S}'\mathbf{S} = \mathbf{I}$, wenn also die transponierte gleich der inversen Matrix ist.

[3] $\mathbf{S}$ ist ein Beispiel für eine zweidimensionale Rotationsmatrix, die allgemein die Form

$$\mathbf{S} = \begin{pmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{pmatrix}$$

hat. In unserem Beispiel ist $\phi = 30°$.

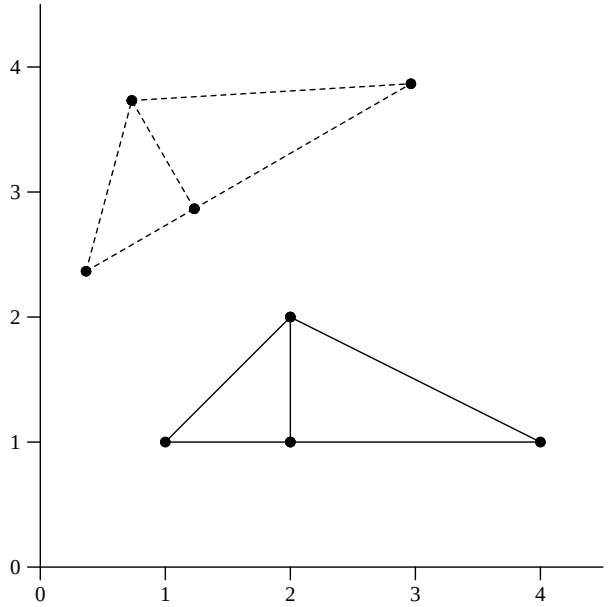

Abb. 3.1-2 Gedrehte und verschobene Konfiguration aus Abb. 3.1-1.

Umgekehrt kann man fragen, wie man eine Konfiguration durch Translationen und Rotationen zu einer vorgegebenen Konfiguration möglichst ähnlich machen kann. Diesem Zweck dient die sogenannte *Prokrustes-Rotation*. Ausgangspunkt sind zwei Konfigurationen, $\mathbf{X}$ und $\mathbf{Y}$, mit jeweils n Zeilen und p Spalten.

Es gibt verschiedene Varianten des Verfahrens.[4] In einer ersten Variante sind eine orthogonale (p, p)-Matrix $\mathbf{S}$ und eine (n, p)-Translationsmatrix $\mathbf{T}$ gesucht, so dass sich $\mathbf{X}$ und

$$\mathbf{Y}_{(\mathbf{S},\mathbf{T})} := \mathbf{Y}\mathbf{S} + \mathbf{T}$$

möglichst ähnlich sind.[5] Als Kriterium wird

$$\| \mathbf{X} - \mathbf{Y}_{(\mathbf{S},\mathbf{T})} \| \longrightarrow \min$$

verwendet.[6] In einer zweiten Variante wird außerdem eine variable Skalie-

[4] Eine ausführliche Diskussion gibt Commandeur (1991).

[5] Unter einer *Translationsmatrix* verstehen wir eine Matrix, deren Zeilen identisch (= dem oben so genannten Translationsvektor) sind.

[6] Für eine beliebige Matrix $\mathbf{A} = (a_{ij})$ ist der Ausdruck $\| \mathbf{A} \|$ durch

$$\| \mathbf{A} \| = (\Sigma_i \Sigma_j a_{ij}^2)^{1/2}$$

rung zugelassen, d.h. es werden eine orthogonale Matrix $\mathbf{S}$, eine Translationsmatrix $\mathbf{T}$ und ein Skalierungsfaktor α gesucht, so dass

$$\| \mathbf{X} - \mathbf{Y}_{(\alpha,\mathbf{S},\mathbf{T})} \| \longrightarrow \min$$

wobei jetzt $\mathbf{Y}_{(\alpha,\mathbf{S},\mathbf{T})} := \alpha\mathbf{YS} + \mathbf{T}$ ist.[7]

Zur Illustration sei angenommen, dass man die in (3.1) definierte Matrix $\mathbf{X}$ kennt und die Koordinaten der in Abbildung 3.1-2 gestrichelt gezeichneten Konfiguration:

$$\mathbf{Y} = \begin{pmatrix} 0.3660 & 2.3660 \\ 1.2320 & 2.8660 \\ 2.9640 & 3.8660 \\ 0.7320 & 3.7320 \end{pmatrix}$$

Gesucht sind jetzt α, $\mathbf{S}$ und $\mathbf{T}$, so dass $\| \mathbf{Y} - (\alpha\mathbf{XS} + \mathbf{T}) \|$ minimal wird. Man findet:[8]

$$\alpha = 1.0 \qquad \mathbf{S} = \begin{pmatrix} 0.8660 & 0.5000 \\ -0.5000 & 0.8660 \end{pmatrix} \qquad \mathbf{T} = \begin{pmatrix} 0 & 1 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{pmatrix}$$

In diesem Beispiel lässt sich eine perfekte Übereinstimmung erzielen, d.h. der Ausdruck $\| \mathbf{Y} - (\alpha\mathbf{XS} + \mathbf{T}) \|$ wird Null. Im Allgemeinen ist nur eine Annäherung möglich; Beispiele folgen in späteren Abschnitten.

3.2 Metrische MDS-Verfahren

1. Die Problemstellung

Ausgangspunkt ist eine Abstandsmatrix $\mathbf{D} = (d_{ij})$ für n Objekte. Außerdem wird der Zahlenraum für die räumliche Darstellung vorgegeben. Grundsätzlich kann ein Zahlenraum $\mathbf{R}^p$ mit einer beliebigen Dimension p verwendet werden. Wenn man an graphischen Darstellungen interessiert ist, verwendet man meistens den zweidimensionalen Zahlenraum ($p = 2$); das wird im Folgenden angenommen.

Gesucht ist nun eine Konfiguration $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbf{R}^p$, so dass die durch ihre Punkte gegebenen Abstände $\| \mathbf{x}_i - \mathbf{x}_j \|$ möglichst den vorgegebenen Abständen d_{ij} entsprechen. Dafür kann folgendes Kriterium verwendet werden:

$$s(\mathbf{x}_1, \ldots, \mathbf{x}_n) := \sum\nolimits_{j<i} w_{ij} \, (d_{ij} - \| \mathbf{x}_i - \mathbf{x}_j \|)^2 \tag{3.3}$$

definiert. Bei Vektoren entspricht er ihrer euklidischen Länge.

[7] Die mathematischen Hintergründe des Verfahrens sind kompliziert und sollen hier nicht besprochen werden; man vgl. beispielsweise Mardia, Kent und Bibby (1979: 416ff.).

[8] Verwendet wurde das TDA-Skript `pr1.cf`.

Dabei sind w_{ij} nichtnegative Gewichte, die dem Zweck dienen, eine etwas allgemeinere Problemformulierung zu erreichen. Man kann beispielsweise $w_{ij} = 0$ setzen, wenn ein Abstandswert d_{ij} nicht bekannt ist. Bei vollständig bekannten Abstandsmatrizen wird man meistens für alle Gewichte $w_{ij} = 1$ annehmen, so dass man sie ignorieren kann. Die Funktion s wird *Stressfunktion* genannt. Die Aufgabe besteht darin, eine Konfiguration zu finden, die den Wert dieser Funktion minimal macht.

Um einen von der Skalierung der als Input verwendeten Abstandsmatrix unabhängigen Stresswert anzugeben, wird anstelle oder ergänzend zu (3.3) auch folgende Definition verwendet:

$$s_r(\mathbf{x}_1, \ldots, \mathbf{x}_n) := \frac{\sum_{j<i} w_{ij} \, (d_{ij} - \| \mathbf{x}_i - \mathbf{x}_j \|)^2}{\sum_{j<i} w_{ij} \, d_{ij}^2} \tag{3.4}$$

Es sei angemerkt, dass manchmal auch die Quadratwurzel dieses Ausdrucks als Definition der Stressfunktion verwendet wird.

2. Alternative Problemformulierungen

Für die metrische MDS wird meistens die Stressfunktion (3.3) verwendet. Varianten können bei zwei Aspekten ansetzen. Einerseits kann man anstelle der euklidischen Abstände $\| \mathbf{x}_i - \mathbf{x}_j \|$ andere Abstandsdefinitionen verwenden. Vorgeschlagen wurde insbesondere die City-Block-Metrik; die Stressfunktion nimmt dann folgende Form an:

$$s^a(\mathbf{x}_1, \ldots, \mathbf{x}_n) := \sum\nolimits_{j<i} w_{ij} \, (d_{ij} - d^a(\mathbf{x}_i, \mathbf{x}_j))^2 \tag{3.5}$$

wobei $d^a(\mathbf{x}_i, \mathbf{x}_j) = \sum_{k=1}^{p} |x_{ik} - x_{jk}|$ ist.[9] Andererseits kann man die Art des Vergleichs zwischen den vorgegebenen und den zu konstruierenden Abständen verändern. Man kann beispielsweise das Kriterium

$$a(\mathbf{x}_1, \ldots, \mathbf{x}_n) := \sum\nolimits_{j<i} w_{ij} \left| \, d_{ij} - \| \mathbf{x}_i - \mathbf{x}_j \| \, \right| \tag{3.6}$$

verwenden, also absolute anstelle der quadrierten Abweichungen (vgl. Heiser 1988). Oder man kann das Kriterium

$$m(\mathbf{x}_1, \ldots, \mathbf{x}_n) := \max\{ \left| \, d_{ij} - \| \mathbf{x}_i - \mathbf{x}_j \| \, \right| \Big| \, 1 \le i < j \le n \} \tag{3.7}$$

verwenden, d.h. man versucht, die maximale Abweichung zwischen den vorgegebenen und den zu konstruierenden Abständen zu minimieren.

[9] Vgl. Pliner (1986); Hubert, Arabie und Hesson-Mcinnis (1992); Groenen, Heiser und Meulman (1998).

3. Rechentechnische Probleme

In allen Varianten treten rechentechnische Probleme auf. Dies gilt auch für die Standardvariante mit der Stressfunktion (3.3), an der wir uns im Folgenden orientieren. Das Hauptproblem besteht darin, dass die als Kriterium verwendete Funktion meistens zahlreiche (und unter Umständen sehr viele) lokale Minima aufweist und alle üblichen Minimierungsalgorithmen nur solche lokalen Minima finden können. Infolgedessen findet man nicht unbedingt auch ein globales Minimum der Zielfunktion.

Bei der praktischen Durchführung einer metrischen MDS ist es deshalb sinnvoll, den Minimierungsalgorithmus ausgehend von unterschiedlichen Startkonfigurationen sehr oft zu wiederholen. So kann man sich einen gewissen Überblick über lokale Minima verschaffen und schließlich das beste der bisher gefundenen Ergebnisse auswählen.

4. Illustration mit Klausurdaten

Zur Illustration des Verfahrens verwenden wir die Klausurdaten. Ausgangspunkt ist die in Abschnitt 2.3 (§ 7) mit dem Dissimilaritätsindex gebildete Abstandsmatrix (vgl. Tabelle 2.3-5)

$$\mathbf{D} = \begin{pmatrix} 0.0000 & 0.1087 & 0.3913 & 0.3913 & 0.2826 \\ 0.1087 & 0.0000 & 0.3913 & 0.4130 & 0.2826 \\ 0.3913 & 0.3913 & 0.0000 & 0.3696 & 0.3043 \\ 0.3913 & 0.4130 & 0.3696 & 0.0000 & 0.1522 \\ 0.2826 & 0.2826 & 0.3043 & 0.1522 & 0.0000 \end{pmatrix} \tag{3.8}$$

Sie bildet die Eingabedaten für eine metrische MDS mit dem Kriterium (3.3) ohne Gewichte (alle $w_{ij} = 1$). Bei 100 Wiederholungen eines Minimierungsalgorithmus für dieses Kriterium, die mit der TDA-Prozedur `mdsm` durchgeführt wurden,[10] wurde in 41 Fällen das relativ beste lokale Minimum gefunden. Die Stresswerte sind $s = 0.002334$ bzw. $s_r = 0.002215$. Die entsprechende Konfiguration ist

$$\mathbf{X} = \begin{pmatrix} 0.0000 & 0.0000 \\ 0.0666 & 0.0605 \\ -0.0425 & 0.4151 \\ -0.3394 & 0.1919 \\ -0.2030 & 0.1670 \end{pmatrix} \tag{3.9}$$

Um eine eindeutige Formulierung zu erreichen, erhält der erste Punkt die Koordinaten $(0, 0)$. Folgende Matrix zeigt die euklidischen Abstände zwischen ihren Punkten:

$$\mathbf{D}_x = \begin{pmatrix} 0.0000 & 0.0900 & 0.4172 & 0.3899 & 0.2629 \\ 0.0900 & 0.0000 & 0.3709 & 0.4267 & 0.2899 \\ 0.4172 & 0.3709 & 0.0000 & 0.3714 & 0.2954 \\ 0.3899 & 0.4267 & 0.3714 & 0.0000 & 0.1387 \\ 0.2629 & 0.2899 & 0.2954 & 0.1387 & 0.0000 \end{pmatrix} \tag{3.10}$$

[10] Das Skript ist `mdsm4.cf`.

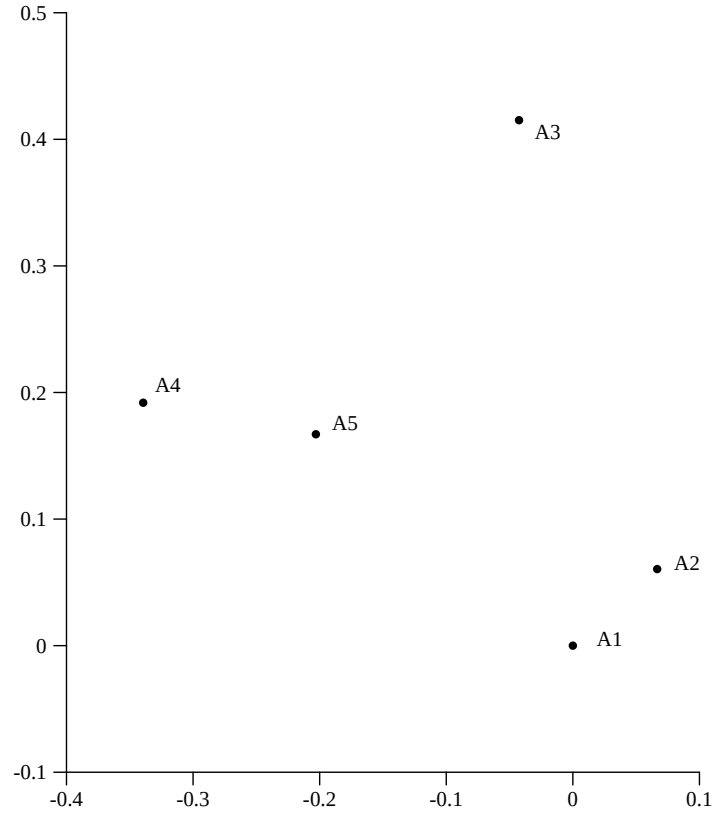

Abb. 3.2-1 Metrische MDS mit der Abstandsmatrix (3.8) für die Klausurdaten. Die Label beziehen sich auf die Aufgaben (= Zeilen der Matrix).

Offenbar ist diese Abstandsmatrix sehr ähnlich zu $\mathbf{X}$.

Abbildung 3.2-1 veranschaulicht die gefundene Konfiguration.[11] Bei der Interpretation dieser Abbildung muss zunächst berücksichtigt werden, dass lediglich die Abstände zwischen den einzelnen Punkten gedeutet werden können, nicht jedoch die Werte der Koordinaten. Dann lässt sich festhalten, dass die Klausuraufgaben 1 und 2 sowie die Klausuraufgaben 4 und 5 bezogen auf die Verteilung von Leistungen relativ ähnlich sind, dagegen Aufgabe 3 etwas „abseits“ liegt. Betrachtet man die in der Matrix (2.1) in Abschnitt 2.2 (§ 6) angegebenen Ausgangsdaten, wird dies leicht ersichtlich: die Aufgaben 1 und 2 wurden von fast allen Teilnehmern sehr gut gelöst; Aufgabe 4 und Aufgabe 5 weisen vergleichsweise viele Leistungen

[11] Die Abbildung wurde mit dem Skript `mdsplot6b.cf` erzeugt.

im mittleren Bereich auf; und Aufgabe 3 wurde von etlichen Teilnehmern sehr schlecht bearbeitet.

5. Metrische MDS mit R

Zur Durchführung einer metrischen MDS stehen in R mehrere Prozeduren zur Verfügung, u.a. ein Verfahren, das bereits 1969 von J. W. Sammon vorgeschlagen wurde,[12] Hier verwenden wir zunächst den Befehl `smacofSym` des Pakets `SMACOF`. Der Name dieses Pakets ist eine Abkürzung für *Scaling by Majorizing a Complicated Function*, was auf die verwendeten Lösungsverfahren hinweist. Die Funktionen des Pakets werden von J. de Leeuw und P. Mair (2009) beschrieben, ebenso wie die genutzten Algorithmen, wobei diese relativ kompliziert sind und hier nicht weiter besprochen werden. Es sei allerdings darauf hingewiesen, dass auch diese Algorithmen eventuell nur ein lokales Minimum finden und die resultierende Lösung somit nicht optimal sein muss.

Dem Befehl muss eine Abstandsmatrix übergeben werden. Werden wie in Box 3.2-1 keine weiteren Optionen spezifiziert, wird eine zweidimensionale Skalierung durchgeführt. Soll eine Konfiguration mit mehr oder weniger Dimensionen erzeugt werden, kann die Option `ndim` verwendet werden. Beispielsweise erhält man mit dem Aufruf `smacofSym(d,ndim=3)` eine dreidimensionale Konfiguration. Weitere Optionen umfassen unter anderem `itmax` und `eps`. Mit der erstgenannten Option kann die Zahl der maximal durchzuführenden Iterationen bestimmt werden (Standardwert: 1000), mit `eps` lässt sich das Konvergenzkriterium bestimmen. Weitere Optionen finden sich in der Hilfe zu `smacofSym`.

Die Ergebnisse des Aufrufs `smacofSym(d)` sind im Objekt `mdsfit` abgespeichert. Wird dieses aufgerufen, wird unter dem Punkt `Call` der Befehlsaufruf angezeigt – in diesem Beispiel ist das Argument `delta`, welches der dem Befehl übergebenen Abstandsmatrix entspricht, gleich `d`, also der Abstandsmatrix der Klausuraufgaben. Anschließend wird angegeben, dass der SMACOF-Algorithmus mit einer symmetrischen Abstandsmatrix ausgeführt wurde, die fünf Objekte enthält. Schließlich sind der Stresswert und die Zahl der Iterationsschritte, die zum Finden der Lösung verwendet wurden, aufgelistet.

Das Objekt `mdsfit` enthält noch weitere Ergebnisse, die allerdings explizit aufgerufen werden müssen. Dabei entspricht jedes Ergebnis einem Eintrag im Objekt `mdsfit`. Die Namen dieser Einträge lassen sich mit `names(mdsfit)` aufrufen und mit `mdsfit$name` direkt anzeigen, wobei für `name` der Name des Eintrags eingesetzt wird. Die aus der MDS resultierende Konfiguration lässt sich beispielsweise mit `mdsfit$conf` anzeigen. Alle Einträge lassen sich mit `mdsfit[]` auf einmal aufrufen. Eine allgemeine Beschreibung der Namen und Inhalte aller Einträge findet sich in der Hilfe zum Befehl `smacofSym`.

[12] Die Prozedur heißt `sammon` und ist im Paket `MASS` enthalten.

Box 3.2-1 R-Skript: Metrische MDS mit Klausurdaten.

```
# Paket SMACOF fuer metrische MDS laden
library(smacof)

# Datenmatrix eingeben
dat <- matrix(c(39, 4, 0, 1, 2,
  40, 1, 4, 0, 1,
  25, 0, 2, 2,17,
  21, 6, 9, 6, 4,
  27, 6, 8, 0, 5),nrow=5,byrow=T)

# Dissimilarittsindex berechnen
dat <- dat/rowSums(dat)
d <- dist(dat,method="manhattan")*0.5

# metrische MDS durchfhren
mdsfit <- smacofSym(d)

# Teil der Ergebnisse anzeigen lassen
mdsfit

# Output des letzten Aufrufs:
Call: smacofSym(delta = d)

Model: Symmetric SMACOF
Number of objects: 5

Metric stress: 0.002214244
Number of iterations: 14

# Grafische Darstellung
plot(mdsfit$conf,main="",xlab="",ylab="",bty="l")
text(mdsfit$conf, labels=c("A1","A2","A3","A4","A5"), adj=1.2)
```

Box 3.2-2 zeigt einen Teil des durch `mdsfit[]` erzeugten Outputs. Zunächst die für die Berechnung verwendete Abstandsmatrix. Es handelt sich um eine skalierte Variante der in (3.8) angegebenen Matrix $\mathbf{D}$. Die Skalierung beruht darauf, dass ausgehend von der Stressfunktion (3.3) angenommen wird, dass die für die Berechnung verwendeten Abstände folgende Normierungsbedingung erfüllen:

$$\sum\nolimits_{j<i} w_{ij}\, d_{ij}^2 = \frac{n\,(n-1)}{2} \tag{3.11}$$

Wenn der Anwender keine unterschiedlichen Gewichte vorgibt, läuft dies darauf hinaus, dass anstelle der als Input angegebenen Abstandsmatrix $\mathbf{D} = (d_{ij})$ eine Abstandsmatrix $\mathbf{D}^* = (\gamma\, d_{ij})$ verwendet wird, wobei

$$\gamma := \sqrt{\frac{n\,(n-1)/2}{\sum_{j<i} d_{ij}^2}} \tag{3.12}$$

Box 3.2-2 Einige Ergebnisse des `mdsfit[]`-Aufrufs.

```
#Beobachtete Abstaende; nur Eintraege unter der Hauptdiagonalen
mdsfit$obsdiss
1 2 3 4
2 0.3348248
3 1.2053692 1.2053692
4 1.2053692 1.2723341 1.1384042
5 0.8705444 0.8705444 0.9375093 0.4687547

#Konfiguration
mdsfit$conf
D1 D2
1 -0.5689632 -0.20538404
2 -0.6149081  0.06775106
3  0.3256921  0.71677596
4  0.6176932 -0.38967859
5  0.2404860 -0.18946439

#Abstaende der Konfiguration
mdsfit$confdiss
1 2 3 4
2 0.2772796
3 1.2862545 1.1440543
4 1.2022138 1.3162005 1.1456057
5 0.8105035 0.8942200 0.9112466 0.4275228

#Metrischer Stress (Wert der Stressfunktion)
mdsfit$stress.m
[1] 0.002214244
```

verwendet wird. In unserem Beispiel ist $\gamma \approx 3.0803$. Ausgewiesen wird jedoch der in (3.4) definierte relative Stresswert (der in diesem Beispiel näherungsweise mit dem in § 4 angegebenen Wert übereinstimmt).

Infolge der Skalierung der für die Rechnung verwendeten Abstandsmatrix ist auch die in Box 3.2-2 angegebene Konfiguration

$$\mathbf{X}^* = \begin{pmatrix} -0.5689632 & -0.20538404 \\ -0.6149081 & 0.06775106 \\ 0.3256921 & 0.71677596 \\ 0.6176932 & -0.38967859 \\ 0.2404860 & -0.18946439 \end{pmatrix} \tag{3.13}$$

skaliert. Abbildung 3.2-2 veranschaulicht diese Konfiguration. Die interpretierbaren relativen Abstände entsprechen offenbar denjenigen in Abbildung 3.2-1. Verwendet man die in Abschnitt 3.1 (§ 4) besprochene Methode der Prokrustes-Rotation, um $\mathbf{X}^*$ mit der in (3.9) angegebenen Konfiguration $\mathbf{X}$ zu vergleichen, findet man, dass näherungsweise $\mathbf{X}^* = \gamma\,\mathbf{X}\,\mathbf{S} + \mathbf{T}$

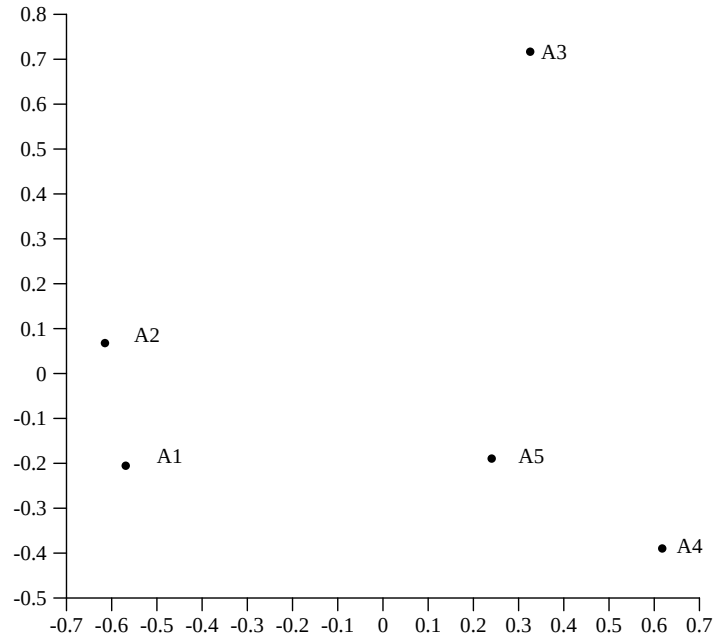

Abb. 3.2-2 Darstellung der Konfiguration $\mathbf{X}^*$ für die Klausurdaten.

gilt,[13] wobei

$$\gamma = 3.0803 \qquad \mathbf{S} = \begin{pmatrix} -0.7849 & 0.6197 \\ 0.6197 & 0.7849 \end{pmatrix} \qquad \mathbf{T} = \begin{pmatrix} -0.5692 & -0.2056 \\ -0.5692 & -0.2056 \\ -0.5692 & -0.2056 \\ -0.5692 & -0.2056 \\ -0.5692 & -0.2056 \end{pmatrix}$$

6. Illustration mit Berufsstrukturdaten

Als zweites Beispiel verwenden wir die Berufsstrukturdaten aus Abschnitt 2.3. Zunächst wird mittels des Dissimilaritätsindex eine Abstandsmatrix aus Tabelle 2.3-4 gebildet.[14] Abbildung 3.2-3 zeigt eine mit dem Stresskriterium (3.3) an diese Abstandsmatrix angepasste Konfiguration.[15] Der Wert der relativen Stressfunktion für diese Konfiguration beträgt $s_r = 0.008588$. Dieser relativ beste Wert wurde in 31 von 100 Wiederholungen gefunden.

[13] Für die Berechnung wurde das TDA-Skript `pr6.cf` verwendet.

[14] Das Datenfile wird `bs4b.dat` genannt.

[15] Sie wurde mit der TDA-Prozedur `mdsm` erzeugt, das Skript ist `mdsm5.cf`.

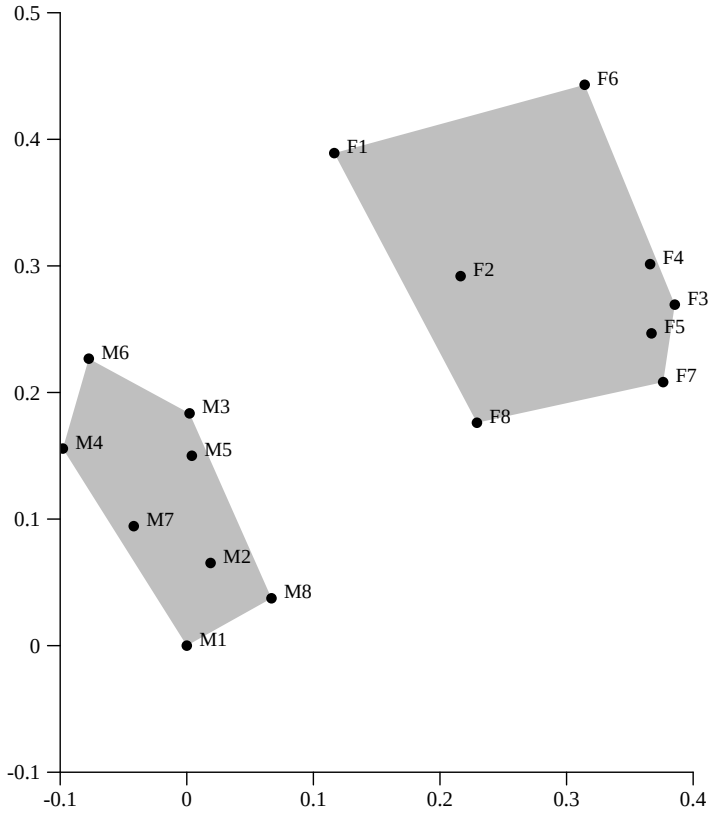

Abb. 3.2-3 Mit einer MDS erzeugte Konfiguration für die Dissimilaritätsabstände zwischen den Zeilen der Tabelle 2.3-4.

Um die Berechnungen mit R durchzuführen, kann das in Box 3.2-3 angegebene Skript verwendet werden. Es zeigt zunächst (wie bereits in Abschnitt 2.3 erläutert wurde) die Erzeugung der Abstandsmatrix. Dann folgt der Aufruf des `smacofSym`-Befehls. Der nach XXX Iterationen erreichte Stresswert entspricht dem oben angegebenen Wert, so dass man annehmen kann, dass eine optimale Konfiguration gefunden wurde.

Schließlich wird mittels des Befehls `plot` eine Grafik erstellt, bei der die einzelnen Koordinaten der Konfiguration als Punkte eingezeichnet werden. Über die Argumente `main`, `xlab` und `ylab` werden der Titel der Grafik, die Beschriftung der x-Achse und die Beschriftung der y-Achse gesteuert. In diesem Beispiel wird keine Beschriftung verwendet. Über das Argument `pch` werden ausgefüllte Punkte als Symbole für die Koordinaten ausgewählt, und über `bty` wird festgelegt, dass um die Grafik eine L-frmige

Box 3.2-3 R-Skript: Metrische MDS mit Berufsstrukturdaten.

```
# Paket SMACOF laden
library(smacof)

# Daten aufbereiten, s. Box 2.3-3
dat <- read.table("bs1.dat")
names(dat) <- c("X","Y","Z","h")
tab1 <- xtabs(h~.,dat)
tab2 <- ftable(tab1, row.vars=c("Z","X"), col.vars="Y")
tab3 <- prop.table(tab2,1)

# Dissimilaritaetsindex berechnen
d <- dist(tab3,method="manhattan")

# Metrische MDS durchfuehren
mdsfit <- smacofSym(d)

# Konfiguration anzeigen
mdsfit$conf

# Grafik erstellen (s. Abb. 3.2-4)

plot(mdsfit$conf,main="",xlab="",ylab="",bty="l",col="red")
text(mdsfit$conf, labels=c(
+ "M1","M2","M3","M4","M5","M6","M7","M8",
+ "F1","F2","F3","F4","F5","F6","F7","F8"), adj=1.2)
segments(mdsfit$conf[1:8,1],mdsfit$conf[1:8,2],
+          mdsfit$conf[9:16,1],mdsfit$conf[9:16,2],lty=3)
```

Box gezeichnet wird. Der Befehl `text` wird dazu verwendet, um eine Beschriftung der Punkte einzufügen, der Befehl `segments` dient dazu, die gestrichelten Linien zwischen den Punktpaaren der einzelnen Länder zu ergänzen. Abbildung 3.2-4 zeigt das Ergebnis.

Bei der Betrachtung der Abbildungen 3.2-3 bzw. 3.24 fällt auf, dass diese relativ klar in zwei Hälften geteilt ist, wobei die eingezeichneten Punkte für die beiden Geschlechter jeweils eine dieser Hälften bilden. Andererseits gibt es auch innerhalb der beiden Geschlechtergruppen und zwischen den einzelnen Ländern deutliche Unterschiede. Zum Beispiel ist der Punkt für japanische Frauen näher am Punkt der japanischen Männer als am Punkt der türkischen Frauen. Gleichzeitig ist der Abstand zwischen M1 und F1 vergleichsweise groß.

7. Das Shepard-Diagramm

Ein Hilfsmittel, um die Güte der Anpassung zu visualisieren, ist ein von Shepard (1962) vorgeschlagenes und nach ihm benanntes Diagramm, welches sowohl in Kombination mit metrischer als auch nichtmetrischer MDS verwendet werden kann (Kruskal und Wish 1978: 19). Bei diesem wer-

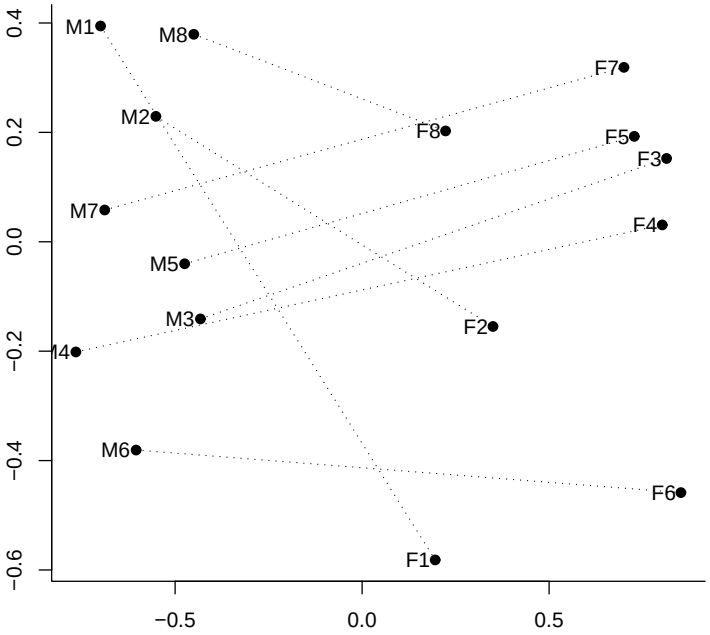

Abb. 3.2-4 Mit den in Box 3.2-3 angegebenen R-Befehlen erzeugte Darstellung der MDS-Konfiguration für die Berufsstrukturdaten.

den die mittels MDS berechneten Abstände d^*_{ij} (Abstände zwischen den Punkten der ermittelten Konfiguration) gegen die ursprünglichen (als Input der MDS verwendeten) Abstände d_{ij} abgetragen. Bei einer optimalen Anpassung sollte $d^*_{ij} = d_{ij}$ gelten und sollten alle Punkte auf der Winkelhalbierenden liegen. Zur Illustration zeigt Abbildung 3.2-5 ein mit TDA erzeugtes Shepard-Diagramm für die MDS mit den Berufsstrukturdaten.

Dieses Diagramm kann in `R` mittels der Befehle in Box 3.2-4 erstellt werden. Zunächst wird das Paket `MASS` geladen, welches den Befehl `Shepard` enthält. Diesem werden als Argument `d` die als Input verwendeten Abstände und als Argument `x` eine aus einer MDS resultierende Konfiguration übergeben. Die durch diese Konfiguration implizierten Abstände werden dann beim Befehlsaufruf berechnet. Da bei der Verwendung des Befehls `smacofSym` die ursprüngliche Abstandsmatrix normiert wird, ist darauf zu achten, dass diese normierte Abstandsmatrix fr `d` angegeben wird. Das über den Befehl `Shepard` erzeugte Objekt `sh` enthält dann als Eintrag `x` die normierten Abstnde und als Eintrag `y` die durch die Konfigu-

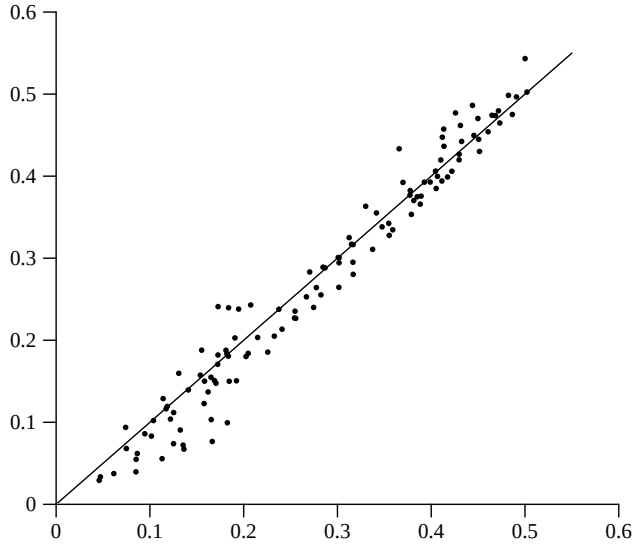

Abb. 3.2-5 Shepard-Diagramm für die MDS mit den Berufsstrukturdaten (unter Verwendung der Konfiguration aus Abbildung 3.2-3).

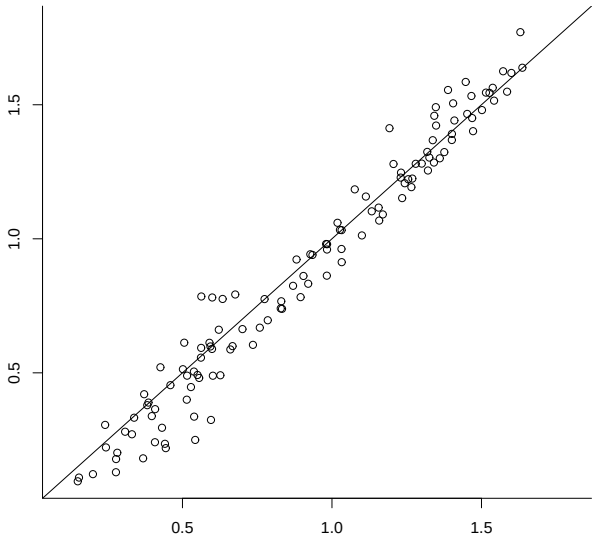

Abb. 3.2-6 Analog zu Abb. 3.2-5 mit den R-Befehlen in Box 3.2-4 erzeugtes Shepard-Diagramm für die Berufsstrukturdaten.

Box 3.2-4 R-Befehle zur Erzeugung eines Shepard-Diagramms.

```
# Paket MASS laden
library(MASS)

# implizierte Abstnde berechnen
sh <- Shepard(d=mdsfit$obsdiss,x=mdsfit$conf)

# Shepard-Diagramm erzeugen
plot(sh$x,sh$y,xlab="",ylab="")
abline(0,1) # Winkelhalbierende einfuegen

# Koorelationskoeffizient berechnen
cor(sh$x,sh$y)^2
```

ration der MDS implizierten Abstände. Zur Visualisierung kann der Befehl `plot` benutzt werden. Über den Aufruf von `abline(0,1)` wird zusätzlich eine Gerade mit Achsenabschnitt 0 und Steigung 1 in das Diagramm eingezeichnet. Abbildung 3.2-6 zeigt das Ergebnis.

Mittels des Befehls `cor` wird der Korrelationskoeffizient zwischen den als Input verwendeten *normierten* und den aus der Konfiguration resultierenden Abständen berechnet. Wird dieser quadriert, erhält man ein Maß für die Güte der Anpassung, welches Werte zwischen 0 und 1 annehmen kann. In unserem Beispiel ergibt sich ein Wert von 0.9651, was auf eine insgesamt gute Anpassung schließen lässt.[16]

3.3 Nichtmetrische MDS-Verfahren

1. Die Problemstellung

In diesem Abschnitt besprechen wir das Verfahren der *nichtmetrischen MDS*. Das Verfahren wurde Mitte der 1960er Jahre von J. B. Kruskal vorgeschlagen (Kruskal 1964a, 1864b), seitdem gibt es viele weitere Beiträge.[17] Ausgangspunkt ist wiederum eine Abstandsmatrix $\mathbf{D} = (d_{ij})$, und gesucht ist eine Konfiguration $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbf{R}^p$, deren Abstände die vorgegebenen Abstände d_{ij} möglichst gut repräsentieren. Wir beschränken uns auch in diesem Abschnitt auf den Fall $p = 2$ (eindimensionale Skalierung wird im nächsten Kapitel besprochen) und nehmen an, dass für die Konfiguration euklidische Abstände verwendet werden. Im Unterschied zum Vorgehen bei der metrischen MDS wird jetzt jedoch nur gefordert, dass die Ordnungen der vorgegebenen und der durch die Konfiguration gebildeten Abstände sich möglichst gut entsprechen sollen.

[16] Denselben Wert findet man mit den Punkten in Abbildung 3.2-5, da der Korrelationskoeffizient von der Skalierung unabhängig ist.

[17] Wir beziehen uns u.a. auf Kruskal und Wish (1978); Cox und Cox (1994: 42ff.).

Um das zu präzisieren, nehmen wir zunächst an, dass es in der Abstandsmatrix $\mathbf{D}$ keine Bindungen gibt. Dann können die Abstände d_{ij} in eine streng aufsteigende Reihenfolge gebracht werden, und mit passend gewählten Indizes kann man einen Vektor

$$\mathbf{d} = (d_1, \ldots, d_q)' \quad \text{mit} \quad d_1 < d_2 < \cdots < d_q$$

bilden, wobei $q := n\,(n-1)/2$ die Anzahl der Abstände im unteren Dreieck der Abstandsmatrix $\mathbf{D}$ ist (da $\mathbf{D}$ symmetrisch ist, genügt es, diese Abstände zu betrachten). Ist nun eine Konfiguration $\mathbf{X}$ gegeben, gibt es korrespondierend zu jedem Abstand d_{ij} einen Abstand zwischen $\mathbf{x}_i$ und $\mathbf{x}_j$, den wir mit d^x_{ij} bezeichnen. Diese Abstände werden analog zu $\mathbf{d}$, also in derselben Reihenfolge, zu einem Vektor

$$\mathbf{d}^x = (d^x_1, \ldots, d^x_q)'$$

zusammengefasst. Gesucht ist schließlich eine Konfiguration, die möglichst gut folgender Bedingung genügt:

$$d_j < d_k \implies d^x_j \le d^x_k \tag{3.14}$$

und natürlich soll auch $d^x_1 < d^x_q$ sein. Zwar ist nicht sicher, dass man eine solche Konfiguration finden kann; aber man kann jedenfalls die Menge der Vektoren angeben, die zu einer im Sinne des Kriteriums (3.14) perfekten Lösung führen würden, nämlich

$$\mathcal{R}_q := \{(r_1, \ldots, r_q)' \,|\, r_1 \le r_2 \le \cdots \le r_q, r_1 < r_q\} \tag{3.15}$$

Wenn man eine Konfiguration $\mathbf{X}$ finden kann, so dass $\mathbf{d}^x \in \mathcal{R}_q$ ist, hat man eine perfekte Lösung gefunden. Ansonsten ist eine möglichst gute Annäherung gesucht, was durch folgende Forderung präzisiert wird: Gesucht ist eine Konfiguration $\mathbf{X}^*$, so dass

$$\min_{\mathbf{r}\in\mathcal{R}_q} \|\,\mathbf{d}^{x^*} - \mathbf{r}\,\| \;\le\; \min_{\mathbf{r}\in\mathcal{R}_q} \|\,\mathbf{d}^x - \mathbf{r}\,\| \tag{3.16}$$

für alle möglichen Konfigurationen $\mathbf{X}$ ist. Eine äquivalente Formulierung verwendet eine explizite Bezeichnung eines Vektors $\check{\mathbf{d}}^x \in \mathcal{R}_q$, der zu $\mathbf{d}^x$ einen minimalen Abstand hat, für den also

$$\|\,\check{\mathbf{d}}^x - \mathbf{d}^x\,\| \le \|\,\mathbf{r} - \mathbf{d}^x\,\| \quad \text{(für alle } \mathbf{r} \in \mathcal{R}_q)$$

gilt.[18] Mit dieser Bezeichnung kann folgende *Stressfunktion* definiert werden:

$$s(\mathbf{X}) := \frac{\|\,\check{\mathbf{d}}^x - \mathbf{d}^x\,\|}{\|\,\mathbf{d}^x\,\|} = \sqrt{\frac{\sum_{k=1}^q (\check{d}^x_k - d^x_k)^2}{\sum_{k=1}^q (d^x_k)^2}} \tag{3.17}$$

[18] Dieser Vektor ist eindeutig bestimmt und wird auch als Projektion von $\mathbf{d}^x$ auf $\mathcal{R}_q$ bezeichnet; vgl. Rohwer und Pötter (2002a: 179).

und die Aufgabe besteht darin, eine Konfiguration $\mathbf{X}$ zu finden, die diese Stressfunktion minimiert.[19]

2. Berechnungsmethoden

Um Lösungen für das nichtmetrische MDS-Problem zu finden, wurde von Kruskal (1964b) ein iteratives Verfahren vorgeschlagen. Eine genaue Beschreibung findet man bei Cox und Cox (1994: 50ff.).[20] Hier besprechen wir nur einige allgemeinere Punkte.

Zunächst ist zu bemerken, dass der in § 1 skizzierte Ansatz noch nicht zu einer eindeutig bestimmten Konfiguration führt. Es werden deshalb *normalisierte* Konfigurationen verwendet, die durch zwei Bedingungen definiert sind: ihre Spalten haben den Mittelwert 0, und die Summe der quadrierten euklidischen Abstände zwischen ihren n Punkten ist gleich n^2.[21] Gleichwohl sind unterschiedliche Lösungen möglich. Man betrachte zum Beispiel die Abstandsmatrix

$$\mathbf{D} = \begin{pmatrix} 0 & 1 & 2 & 4 \\ 1 & 0 & 3 & 5 \\ 2 & 3 & 0 & 6 \\ 4 & 5 & 6 & 0 \end{pmatrix} \tag{3.18}$$

Hier sind zwei Konfigurationen (in normalisierter Form) und zugehörige Abstandsmatrizen:

$$\mathbf{X}' = \begin{pmatrix} -0.3043 & -0.0460 \\ -0.4622 & 0.2381 \\ 0.4249 & 0.5440 \\ 0.3415 & -0.7361 \end{pmatrix} \quad \mathbf{D}' = \begin{pmatrix} 0.0000 & 0.3251 & 0.9380 & 0.9451 \\ 0.3251 & 0.0000 & 0.9384 & 1.2629 \\ 0.9380 & 0.9384 & 0.0000 & 1.2828 \\ 0.9451 & 1.2629 & 1.2828 & 0.0000 \end{pmatrix}$$

und

$$\mathbf{X}'' = \begin{pmatrix} 0.1164 & -0.2735 \\ 0.5282 & -0.1463 \\ -0.4123 & -0.4223 \\ -0.2323 & 0.8420 \end{pmatrix} \quad \mathbf{D}'' = \begin{pmatrix} 0.0000 & 0.4310 & 0.5492 & 1.1687 \\ 0.4310 & 0.0000 & 0.9802 & 1.2470 \\ 0.5492 & 0.9802 & 0.0000 & 1.2770 \\ 1.1687 & 1.2470 & 1.2770 & 0.0000 \end{pmatrix}$$

Beide Konfigurationen liefern eine ordinale Repräsentation der Abstände in $\mathbf{D}$ (die Stressfunktion (3.17) hat in beiden Fällen den Wert 0), sie können jedoch durch eine Prokrustes-Rotation nicht in Übereinstimmung gebracht werden.

[19] In der Literatur findet man auch noch andere Varianten der Stressfunktion; vgl. Kruskal und Wish (1978: 26).

[20] Varianten dieses Verfahren werden in vielen Statistikprogrammen verwendet, u.a. auch in der TDA-Prozedur `mdsn`, die hier für einige Illustrationen verwendet wird.

[21] Oder anders formuliert: der durchschnittliche quadrierte Abstand der Punkte vom Nullpunkt $(0, 0)$ soll gleich 1 sein.

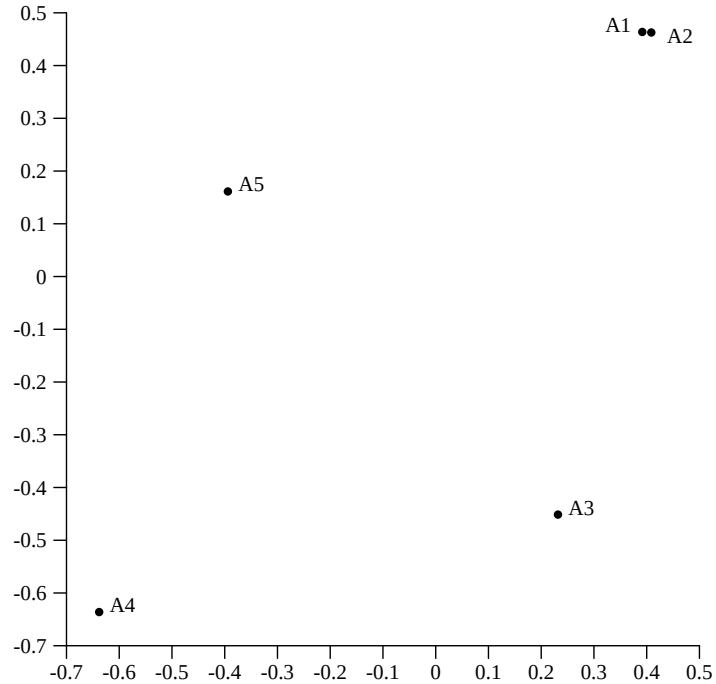

Abb. 3.3-1 Darstellung der Konfiguration $\mathbf{X}'$ für die Klausurdaten.

Eine weiteres Problem betrifft Bindungen, denn um die Stressfunktion (3.17) zu minimieren, ist es insbesondere erforderlich, den Projektionsvektor $\check{\mathbf{d}}$ zu berechnen. Wenn es in der Abstandsmatrix keine Bindungen gibt, wie bisher angenommen wurde, kann das dadurch erreicht werden, dass man die Projektion von $\mathbf{d}^x$ auf $\mathcal{R}_q$ berechnet. Wenn Bindungen auftreten, kann man jedoch auf zwei unterschiedliche Weisen vorgehen (vgl. Kruskal 1964a: 22). Man kann sich auf das Kriterium (3.14) beschränken, also offen lassen, in welchem Verhältnis d_j^x und d_k^x stehen, wenn $d_j = d_k$ ist (dies ist Kruskals „primary approach“). Oder man kann zusätzlich fordern:

$$d_j = d_k \implies d_j^x = d_k^x \tag{3.19}$$

Dies ist Kruskals „secondary approach“. Die praktische Umsetzung erfolgt so, dass man die Definition der Menge $\mathcal{R}_q$ entsprechend modifiziert, bevor man die Projektion berechnet.

Zur Illustration der beiden Methoden verwenden wir die in Abschnitt 2.3 (§ 7) mit dem Dissimilaritätsindex gebildete Abstandsmatrix für die Klausurdaten (Tabelle 2.3-5). Sie weist zahlreiche Bindungen auf. Verwendet man natürliche Zahlen, um die ordinalen Beziehungen zwischen

den Abständen zu beschreiben, sieht die Abstandsmatrix folgendermaßen aus:

$$\mathbf{D} = \begin{pmatrix} 0 & 1 & 6 & 6 & 3 \\ 1 & 0 & 6 & 7 & 3 \\ 6 & 6 & 0 & 5 & 4 \\ 6 & 7 & 5 & 0 & 2 \\ 3 & 3 & 4 & 2 & 0 \end{pmatrix} \tag{3.20}$$

Die erste Methode für Bindungen führt zu der folgenden Konfiguration $\dot{X}'$, die zweite zur Konfiguration $\mathbf{X}''$.

$$\mathbf{X}' = \begin{pmatrix} 0.3916 & 0.4637 \\ 0.4087 & 0.4627 \\ 0.2318 & -0.4514 \\ -0.6380 & -0.6362 \\ -0.3940 & 0.1613 \end{pmatrix} \quad \mathbf{X}'' = \begin{pmatrix} -0.4292 & 0.3146 \\ -0.4293 & 0.3145 \\ -0.1300 & -0.8617 \\ 0.7342 & -0.0311 \\ 0.2543 & 0.2637 \end{pmatrix}$$

Beide Konfigurationen sind normalisiert und liefern eine vollständige Stressreduktion (d.h. die Stressfunktion hat den Wert 0). Die zugehörigen euklidischen Abstandsmatrizen sind:

$$\mathbf{D}' = \begin{pmatrix} 0.0000 & 0.0171 & 0.9289 & 1.5066 & 0.8418 \\ 0.0171 & 0.0000 & 0.9311 & 1.5176 & 0.8574 \\ 0.9289 & 0.9311 & 0.0000 & 0.8892 & 0.8758 \\ 1.5066 & 1.5176 & 0.8892 & 0.0000 & 0.8340 \\ 0.8418 & 0.8574 & 0.8758 & 0.8340 & 0.0000 \end{pmatrix}$$

und

$$\mathbf{D}'' = \begin{pmatrix} 0.0000 & 0.0001 & 1.2137 & 1.2137 & 0.6854 \\ 0.0001 & 0.0000 & 1.2137 & 1.2138 & 0.6854 \\ 1.2137 & 1.2137 & 0.0000 & 1.1986 & 1.1891 \\ 1.2137 & 1.2138 & 1.1986 & 0.0000 & 0.5633 \\ 0.6854 & 0.6854 & 1.1891 & 0.5633 & 0.0000 \end{pmatrix}$$

Man erkennt, dass beide Abstandsmatrizen die Bedingung (3.14) erfüllen, die zweite Abstandsmatrix außerdem die Bedingung (3.19).

Abbildung 3.3-1 veranschaulicht die Konfiguration $\mathbf{X}'$, Abbildung 3.3-2 ist das zugehörige Shepard-Diagramm. Die vollständige Stressreduktion zeigt sich im monotonen Verlauf der Funktion. Die graphische Darstellung in Abbildung 3.3-1 verdeutlicht auch ein Problem der nichtmetrischen MDS: dass das Ausmaß der sichtbaren Abstände nicht interpretiert werden kann, weil das Konstruktionsverfahren nur auf den ordinalen Beziehungen beruht.

3. Nichtmetrische MDS mit R

Zur Durchführung einer nichtmetrischen MDS in `R` kann wieder der Befehl `smacofSym` benutzt werden, wobei eine von dem von Kruskal vorgeschla-

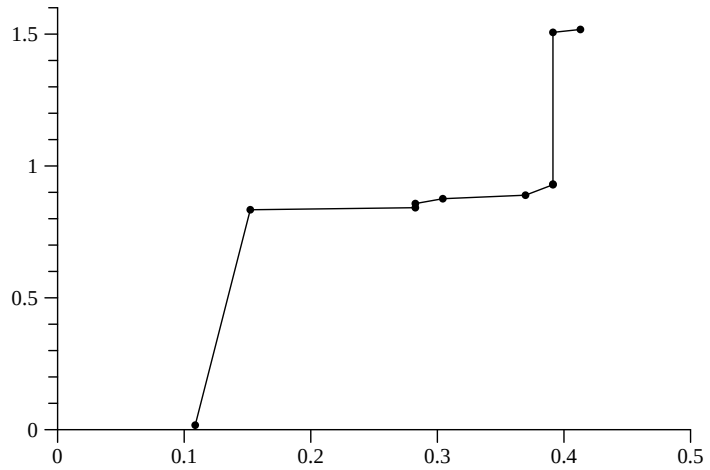

Abb. 3.3-2 Shepard-Diagramm bei der Konfiguration $\mathbf{X}'$ für die Klausurdaten.

genen Verfahren abweichende Prozedur zum Einsatz kommt[22]. Box 3.3-1 illustriert die Vorgehensweise anhand der Klausurdaten. Im Wesentlichen kann wie bei den vorausgegangenen Beispielen der metrischen MDS vorgegangen werden; allerdings muss nun das bisher nicht verwendete Argument `metric` auf den Wert `FALSE` gesetzt werden. Ferner kann über das Argument `ties` angegeben werden, wie Bindungen behandelt werden sollen. Durch die Angabe von `primary` wie in Box 3.3-1 wird der im letzten Abschnitt besprochene *primary approach* benutzt. Der *secondary approach* kann durch `secondary` ausgewählt werden.[23]

Ergebnisse können nach der Berechnung wie bei der metrischen MDS aufgerufen werden, wobei nun allerdings der nichtmetrische Stress ausgewiesen wird (s. Box 3.3-1). In unserem Beispiel gibt es folgende Konfiguration:

$$\mathbf{X} = \begin{pmatrix} 0.5387800 & 0.114258806 \\ 0.6360755 & -0.006632482 \\ -0.3050978 & -0.735748020 \\ -0.6139564 & 0.408849812 \\ -0.2558013 & 0.219271885 \end{pmatrix} \tag{3.21}$$

[22] Als Alternativen können die Befehle `isoMDS` aus dem Paket `MASS` und der Befehl `metaMDS` aus dem Paket `vegan` benutzt werden. Beide verwenden Varianten des von Kruskal vorgeschlagene Verfahrens, wobei `metaMDS` dieses wiederholt mit zufälligen Startkonfigurationen durchführt.

[23] Darüber hinaus kann durch den Parameter `tertiary` noch eine dritte Variante gewählt werden.

Box 3.3-1 R-Skript: Nichtmetrische MDS mit Klausurdaten.

```
# Paket SMACOF laden
library(smacof)

# Klausurdaten
dat <- matrix(c(39, 4, 0, 1, 2,
40, 1, 4, 0, 1,
25, 0, 2, 2,17,
21, 6, 9, 6, 4,
27, 6, 8, 0, 5),nrow=5,byrow=T)

dat <- dat/rowSums(dat)

d <- dist(dat,method="manhattan")*0.5

# MDS durchfuehren
mdsfit <- smacofSym(d,metric=FALSE,ties="primary")

mdsfit

# Output des letzten Aufrufs
Call: smacofSym(delta = d, metric = F, ties = "primary")

Model: Symmetric SMACOF
Number of objects: 5

Nonmetric stress: 2.960011e-08
Number of iterations: 7

# Grafische Darstellung
plot(mdsfit$conf,main="",xlab="",ylab="",bty="l",ylim=c(-0.8,0.5))
text(mdsfit$conf, labels=c("A1","A2","A3","A4","A5"), adj=1.2,pos=1)
```

Sie ist in Abbildung 3.3-3 dargestellt. Vergleicht man sie mit den Konfigurationen aus dem letzten Abschnitt, zeigen sich zwar deutliche Unterschiede bei den implizierten Abständen, allerdings sind diese nicht direkt interpretierbar, da die Konfigurationen nur die Rangfolge der ursprünglichen Abstände wiedergeben. Ein Vergleich von Abbildung 3.3-3 mit Abbildung 3.3-2 zeigt dann beispielsweise, dass der Abstand zwischen den Punkten A1 und A2 in beiden Fällen am kleinsten ist.

Wird wie in Box 3.3-1 der *primary approach* verwendet, ergibt sich ein relativ niedriger Wert der Stressfunktion, wobei allerdings keine vollständige Stressreduktion erreicht wird. Wird hingegen der *secondary approach* benutzt, beträgt der Wert der nichtmetrischen Stressfunktion etwa 0.15 und ist somit deutlich höher, obwohl im vorherigen Abschnitt ein Beispiel vorgestellt wurde, bei dem unter Verwendung dieses Ansatzes eine vollständige Stressreduktion erreicht wird. Das liegt vermutlich daran, dass die Prozedur `smacofSym` in diesem Beispiel nur ein lokales Minimum

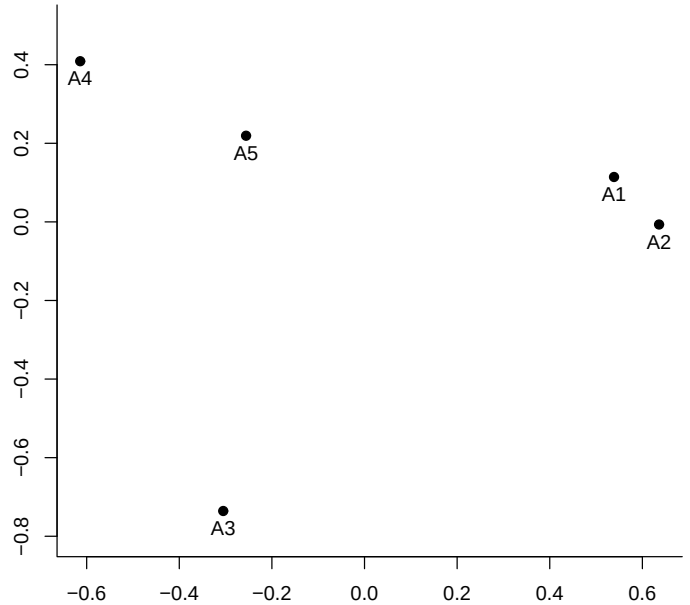

Abb. 3.3-3 Darstellung der in (3.21) angegebenen Konfiguration $\mathbf{X}$ für die Klausurdaten.

der Stressfunktion gefunden hat.

4. Illustration mit Berufsstrukturdaten

Für eine weitere Illustration verwenden wir wieder die Berufsstrukturdaten. Box 3.3-2 zeigt das R-Skript. Die verwendeten Befehle sind aus den vorausgegangenen Abschnitten bekannt. Die einzige Besonderheit liegt in der Angabe von `ties="secondary"` beim Befehl `smacofSym`, d.h. es wird in diesem Beispiel der *secondary approach* zur Behandlung von Bindungen verwendet.

Eine grafische Darstellung der resultierenden Konfiguration findet man in Abbildung 3.3-4. Vergleicht man sie mit der Konfiguration, die sich bei der metrischen MDS ergab (Abbildung 3.2-4), zeigen sich nur geringfügige Unterschiede. Bei diesem Beispiel liefert die Prozedur `smacofSym` auch bei Verwendung des *primary approach* sehr ähnliche Ergebnisse.

Wie bereits anhand der Klausurdaten besprochen wurde, kann auch bei einer nichtmetrischen MDS ein Shepard-Diagramm verwendet werden. Um für das gegenwärige Beispiel ein Shepard-Diagramm zu erzeugen, können

Box 3.3-2 R-Skript: Nichtmetrische MDS mit Berufsstrukturdaten.

```
# Paket SMACOF laden
library(smacof)

# Daten laden und aufbereiten
dat <- read.table("bs1.dat")
names(dat) <- c("X","Y","Z","h")
tab1 <- xtabs(h~.,dat)
tab2 <- ftable(tab1, row.vars=c("Z","X"), col.vars="Y")
tab3 <- prop.table(tab2,1)
d <- dist(tab3,method="manhattan")

# Nichtmetrische MDS durchfuehren
mdsfit <- smacofSym(d,metric=F,ties="secondary")

# Grafische Darstellung
plot(mdsfit$conf,main="",xlab="",ylab="",bty="l",pch=19,xlim=c(-0.8,1))
text(mdsfit$conf, labels=
c("M1","M2","M3","M4","M5","M6","M7","M8",
"F1","F2","F3","F4","F5","F6","F7","F8"), adj=1.2)
segments(mdsfit$conf[1:8,1],mdsfit$conf[1:8,2],
         mdsfit$conf[9:16,1],mdsfit$conf[9:16,2],lty=3)
```

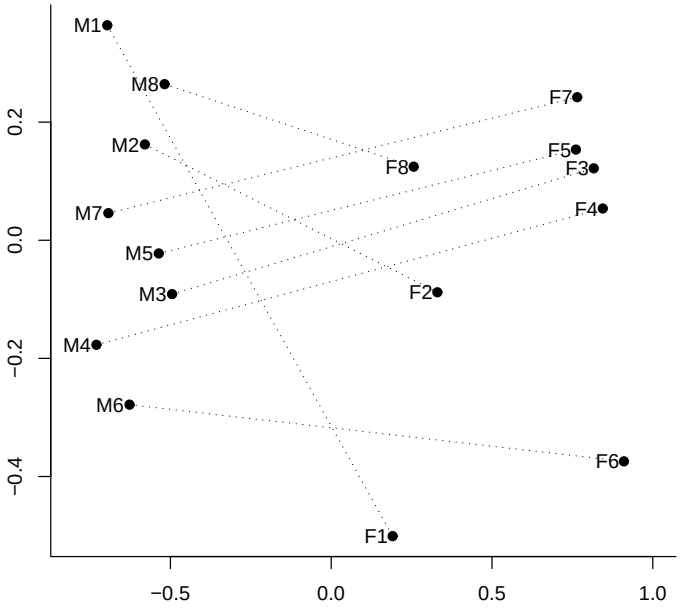

Abb. 3.3-4 Darstellung der mit dem R-Skript in Box 3.3-2 gefundenen Konfiguration für die Berufsstrukturdaten.

Box 3.3-3 R-Befehle zur Erzeugung eines Shepard-Diagramms.

```
# Erzeugung des Shepard-Diagramms
plot(sh$x,sh$y,xlab="",ylab="",bty="l")
lines(sh$x,sh$yf)

# Berechnung einer Rangkorrelation
cor(sh$x,sh$y,method="kendall")
```

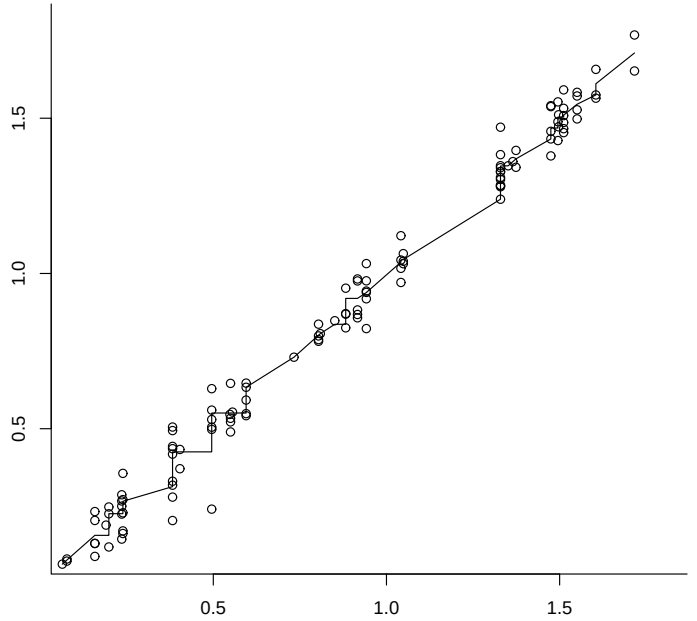

Abb. 3.3-5 Shepard-Diagramm für die nichtmetrische MDS mit den Berufsstrukturdaten, erzeugt mit den R-Befehlen in Box 3.3-3.

die in Box 3.3-3 angegebenen R-Befehle (im Anschluss an das R-Skript in Box 3.3-2) verwendet werden. Abbildung 3.3-5 zeigt das Ergebnis. Als Ergänzung haben wir den Befehl `cor` unter Angabe des Arguments `method="kendall"` benutzt, um einen Rangkorrelationskoeffizienten nach Kendall zu berechnen. Dieser weist einen Wert von etwa 0.92 auf, was auf eine relativ gute Anpassung hindeutet.

3.4 Informationsgehalt von MDS-Bildern

Bei bildlichen Darstellungen von MDS-Konfigurationen können nur Abstände zwischen den dargestellten Punkten interpretiert werden. Da die Konfigurationen beliebig verschoben und gedreht werden können, haben Richtungen keine Bedeutung. Die Frage, welche Informationen aus MDS-Bildern gewonnen werden können, hängt somit zunächst davon ab, ob die Objekte durch Label kenntlich gemacht werden können (beispielsweise durch Ländernamen, wenn man sich auf Länder bezieht). Das ist allerdings nur möglich, wenn die Anzahl der Objekte relativ klein ist.

1. Ergänzungen der MDS-Bilder

Insbesondere in denjenigen Fällen, in denen die Objekte nicht durch Label kenntlich gemacht werden können, stellt sich die Frage, ob noch weitere Informationen, die vielleicht über die dargestellten Objekte verfügbar sind, in den Bildern dargestellt werden können. Es gibt folgende Möglichkeiten:

- Eine einfache Möglichkeit entsteht, wenn es für die dargestellten Objekte eine Reihenfolge gibt, beispielsweise eine zeitliche Reihenfolge. Dann können die Punkte in der MDS-Konfiguration entsprechend ihrer bekannten Reihenfolge durch Linien verbunden werden.
- Eine andere Möglichkeit entsteht, wenn man für die dargestellten Objekte außer der Abstandsmatrix auch noch Werte einer oder mehrerer quantitativer (nicht unbedingt metrischer) Variablen kennt. Dann kann man für jede dieser Variablen eine Achse einzeichnen, so dass die Projektionen der Punkte auf diese Achse mit den Variablenwerten möglichst gut korrelieren.

Im Folgenden illustrieren wir die beiden Möglichkeiten anhand einfacher Beispiele.

2. Illustration mit Schulabschlüssen

Als erstes Beispiel verwenden wir Daten über Schulabschlüsse aus dem ALLBUS (vgl. Tabelle 2.4-1 in Abschnitt 2.4). Für die gegenwärtige Illustration verwenden wir nur die Daten für Frauen und bilden mit ihnen eine $(14, 14)$-Abstandsmatrix. Zur Berechnung von Abständen zwischen den Verteilungen wird der Dissimilaritätsindex verwendet.[24] Mit dieser Abstandsmatrix wird eine metrische MDS durchgeführt.[25] Bei 100 Wiederholungen mit zufälligen Anfangskonfigurationen beträgt der minimale Stresswert 0.0055 und wird in 28 der 100 Wiederholungen erreicht. Abbildung 3.4-1 zeigt die resultierende Konfiguration. Die Punkte wurden

[24] Berechnet mit dem Skript `bi2f.cf`; die Abstandsmatrix wird `bi2f.dat` genannt.

[25] Das Skript ist `mdsm3f.cf`.

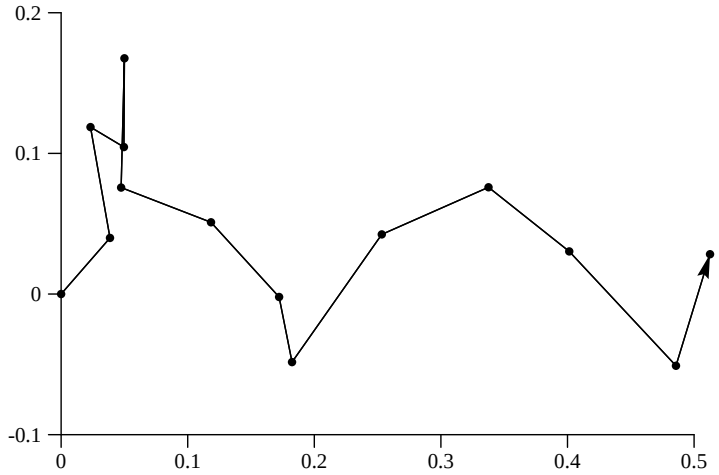

Abb. 3.4-1 MDS-Konstellation der Abstände zwischen den 14 Schulabschlussverteilungen für Frauen in Tabelle 2.4-1.

entsprechend der zeitlichen Reihenfolge der Geburtskohorten verbunden; die Richtung wird durch den Pfeil angegeben.

3. Konstruktion ergänzender Achsen

Jetzt verfolgen wir die zweite der eingangs erwähnten Möglichkeiten.[26] Die MDS-Konfiguration sei durch die Koordinaten (x_i, y_i) für $i = 1, \ldots, n$ gegeben; außerdem gebe es für die Punkte Werte $v_1, \ldots, v_n$ einer quantitativen (nicht unbedingt auch metrischen) Variablen.

Die Achsenkonstruktion verläuft folgendermaßen. Wenn man die Konfiguration um einen Winkel ϕ (im Uhrzeigersinn) dreht, gewinnt man die neuen X-Koordinaten durch

$$x_i^\phi = x_i \cos(\phi) + y_i \sin(\phi)$$

Also kann man einen Winkel ϕ bestimmen, so dass die Rangkorrelation zwischen

$$(x_1^\phi, \ldots, x_n^\phi) \quad \text{und} \quad (v_1, \ldots, v_n)$$

maximal wird.[27] Dann wird eine Achse verwendet, die mit der X-Achse

[26] Überlegungen hierzu in dem bei Green, Carmone und Smith (1989: 318ff.) abgedruckten Beitrag zum Programm PROFIT von Chang und Carroll. Hinweise geben auch Jones und Koehly (1993: 110ff.).

[27] Wir verwenden Kendalls Rangkorrelation; vgl. Rohwer und Pötter (2002a: 164). Auch andere Korrelationsmaße könnten verwendet werden, zum Beispiel der gewöhnliche Korrelationskoeffizient; vgl. Holtmann (1975).

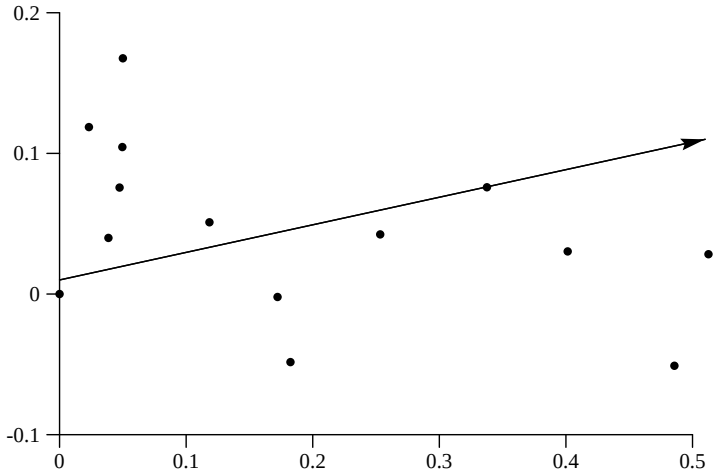

Abb. 3.4-2 MDS-Konstellation der Abstände zwischen den 14 Schulabschlussverteilungen für Frauen mit einer zusätzlichen Achse für die zeitliche Richtung der Geburtskohorten.

des Koordinatensystems den Winkel ϕ bildet. Sie hat die Eigenschaft, dass die Projektionen der Punkte der Konfiguration auf diese Achse mit den auf dieser Achse vorstellbaren Werten $v_1, \ldots, v_n$ am besten korrelieren. Natürlich entsteht eine informative Achse nur dann, wenn eine hohe Korrelation erzielt wird.

Abbildung 3.4-2 zeigt das Ergebnis für unser Beispiel, wobei als ergänzende Variable die zeitliche Reihenfolge der Geburtskohorten verwendet wird. Als optimalen Winkel findet man $\phi = 0.2$ ($= 11.5°$); die Rangkorrelation hat dann den Wert 0.956.[28]

[28] Für die Berechnungen wurde die TDA-Prozedur `mdsr` verwendet; das Skript ist `mdsr1f.cf`. Die Achse wurde so eingezeichnet, dass sie durch den Mittelpunkt der Konfiguration geht.