

Messen von Fähigkeiten und Einstellungen mittels Item Response Theorie

Parameterschätzmethoden für das logistische
Zweiparameter-Modell

Masterarbeit der Philosophisch-naturwissenschaftlichen Fakultät der
Universität Bern

vorgelegt von

Paul Ruppen

2015

Leiter der Arbeit

Prof. Dr. Lutz Dümbgen

Inhaltsverzeichnis

1	Einleitung und Problemstellung	1
2	Ein deterministisches Messmodell	2
3	Probabilistische Mess-Modelle	5
4	Das logistische Zwei-Parameter-Modell	7
4.1	Schätzen der Itemparameter bei gegebenen Fähigkeiten	9
4.2	Schätzen der Fähigkeiten bei gegebenen Item-Parametern	10
4.3	Gemeinsames Schätzen der Fähigkeiten und der Item-Parameter	12
4.4	Marginal Maximum Likelihood-Schätzmethode	13
5	Schlusswort	35
6	Anhang	37
6.1	Skalen	37
6.2	Eigenschaften logistischer <i>ICCs</i>	41
6.3	Newton-Raphson-Verfahren	43
6.4	Gauss-Hermite-Integration	46
6.5	Wahrscheinlichkeitstheoretischer Hintergrund	47
6.6	Eine suffiziente Statistik für Fähigkeiten im dichotomen, logistischen Zwei-Parameter-Modell	48
6.7	EM-Algorithmus	51
6.8	Herleitung der Likelihood-Gleichungen für die MMLE-Methode	52
6.9	Regressionsbeispiel	55
6.10	R-Befehle	58
6.11	R-Paket IRT	61
7	Liste der Symbole	91
	Literatur	93

1 Einleitung und Problemstellung

Ziel der Item Response Theorie (IRT) ist die Analyse von mehreren i. A. dichotomen oder polytomen Variablen oder Items auf dem Hintergrund von Modellen mit latenten, nicht direkt beobachtbaren Variablen. Im Prinzip werden dabei multivariate Regressionsmodelle verwendet, die erhebbare, manifeste Antwortvariablen mit nicht beobachtbaren, latenten Variablen verknüpfen: eine kleine Menge von latenten Variablen soll die Zusammenhänge zwischen einer Vielzahl beobachtbarer Antwortvariablen erklären. Dadurch kann Information zusammengefasst werden: Korrelierte oder gruppenweise korrelierte Variablen können in eine oder mehrere unkorrelierte Variablen verdichtet werden, was für die Anwendung von statistischen Verfahren von Vorteil sein kann. Diesbezüglich kann man die IRT als eine Art Faktorenanalyse betrachten, die für nominal, dichotom oder polytom ordinal skalierte Variablen geeignet ist (Rizopoulos, 2006, S.1).

Die meisten Modelle gehen dabei von einer einzigen latenten Variable aus, die etliche manifeste dichotome oder ordinale Variablen zusammenfasst. Liegen verschiedene latente Dimensionen vor, werden vorgängig Gruppen von manifesten Variablen gebildet, auf Grund inhaltlicher Überlegungen oder bei ordinalskalierten Variablen auch mittels der Berechnung der paarweisen Korrelationskoeffizienten wie Gamma (Goodman & Kruskal, 1954) oder Kendalls τ (Kendall, 1938). Die Kohärenz einer Batterie ordinalskaliertter Variablen wird zusätzlich oft mit Hilfe von Cronbachs Alpha (Cronbach, 1951) überprüft. Letzteres Vorgehen ist nicht über alle Zweifel erhaben, da die Berechnung dieser Kennzahl Varianzen verwendet, die für ordinalskalierte Variablen kaum sinnvoll interpretierbar sind. Neben Modellen für die Zusammenfassung einer Gruppe von Variablen durch eine einzige latente Variable gibt es aber auch Modelle, welche es erlauben, eine vorgegebene Menge von manifesten Variablen auf einmal in mehr als eine latente Dimension zu zerlegen.

Gemäss IRT-Theorie ist die konstruierte latente Variable intervallskaliert - siehe für Erläuterungen zu den verschiedenen Skalenniveaus Anhang 6.1, S. 37. Dies hat den Vorteil, dass Verfahren, welche mindestens intervallskalierte Variablen voraussetzen, auf inhaltliche Bereiche angewendet werden können, die traditionell solchen Verfahren nicht zugänglich waren - wie z. B. Leistungsmessung in der Schule, Stärke der Einstellung zu politischen Themen und zu Produkten oder Betroffenheitsstärke bei Mobbing in der Schule. Man kann dann z. B. den Zusammenhang zwischen der Einstellung zu Mathematik, die mit 5 dichotomen oder ordinalskalierten Variablen gemessen wurde, mit Mathematikfähigkeiten, die ihrerseits mit 10 Aufgaben gemessen wurden, mit einer klassischen, linearen Regression zu erfassen versuchen. Entsprechend häufig wird die IRT in den Erziehungswissenschaften, etwa in internationalen Vergleichsstudien wie TIMSS (s. z. B. International Association for the Evaluation of Educational Achievement (IEA) (2009)) und PISA (s. z. B. OECD (2009)) verwendet.

Gegenüber der traditionellen psychologischen Testtheorie (s. z. B. Lord und Novick (1968)) hat die IRT den Vorteil, dass das Schätzen der Parameter von den spezifischen Items oder von den untersuchten Personen unabhängig ist, sofern die Items dieselbe verborgene Eigenschaft messen und die Personen zur selben Untersuchungsgruppe gehören. Entsprechend erübrigt sich die traditionelle, aufwendige Kalibrierung von Items. Zudem kann man die Parameter verschiedener Tests auf eine identische Skala abbilden, wenn die Tests mindestens ein gemeinsames Item enthalten. Damit kann man z. B. die Leistung von Schülern einer bestimmten Altersklasse über die Zeit hinweg vergleichbar machen, aber auch die Entwicklung einer Kohorte von Schülern messen. Während in der klassischen Testtheorie bei der Verfolgung einer Kohorte die Tests genau gleich schwer sein mussten, ist dies bei der IRT nicht der Fall. Man kann später einen schwierigeren Test anwenden und die Resultate trotzdem vergleichen. Dies ist auch deshalb von Vorteil, weil die Trennschärfe der Tests bezüglich unterschiedlicher Fähigkeitsniveaus nicht überall dieselbe ist (Baker, 2001).

Da die IRT mehrere Items verwendet, um dieselbe verborgene Eigenschaft zu messen, kann man die IRT auch als Variante der verschiedenen Methoden wiederholter Messung auffassen - siehe für eine entsprechende Darstellung bezüglich des Raschmodells Perline, Wright und Wainer (1979) und für eine systematische Übersicht über derartige Verfahren Agresti (2002, Kapitel 11 und 12).

Es gibt verschiedene IRT-Modelle, welche mehrere manifeste diskrete Items auf eine latente

Variable zurückführen:

- Zweiparameter-Modelle: Sie weisen einen Schwierigkeits- und Trennschärfeparameter für jedes Items auf.
- Rasch-Modelle oder Einparameter-Modelle: Diese weisen für jedes Item einen Schwierigkeitsparameter auf. Der Trennschärfeparameter wird damit für alle Items auf 1 gesetzt (Rasch, 1980, Kapitel 5).
- Dreiparameter-Modelle: Neben einem Schwierigkeits- und einem Trennschärfeparameter wird noch ein Rate-Parameter eingefügt, der die Wahrscheinlichkeit zufälligerweise richtiger Antworten ausdrückt.

Jedes dieser Modelle weist Unterarten auf, je nach Anzahl Ausprägungen der Variablen (dichotom oder polytom) und je nach dem Skalenniveau der Daten, die analysiert werden (nominal oder ordinal). Für eine Darstellung der Fülle von Modellen s. van der Linden und Hambleton (1997).

Ziel der vorliegenden Arbeit ist es, ein einfaches Zweiparameter-Modell für dichotome Variablen, welches diese auf eine einzige latente Variable zurückführt, darzulegen. Vorgängig wird an Hand der Guttman-Skala diskutiert, wieso deterministische Messmodelle für die Erfassung von latenten Variablen nicht geeignet sind. Anschliessend erfolgt eine Darlegung des allgemeinen Rahmens probabilistischer Messmodelle. Auf diesem Hintergrund wird das einfache Zweiparameter-Modell für dichotome Variablen von Birnbaum (1968, S. 399, ff.) eingeführt. Für dieses Modell wird die Problematik der Schätzung der Parameter eingehend diskutiert. Bei der von Birnbaum vorgeschlagenen Schätzmethode ergibt sich das Problem, dass die Parameter nicht konsistent schätzbar sind. Zudem ist die Berechnungsmethode nicht stabil. Deshalb schlugen Bock und Aitkin (1981) vor, die Itemparameter in der Randverteilung der gemeinsamen Verteilung von Fähigkeitszufallsvariablen und Item-Parameterstatistiken zu schätzen: die Fähigkeitsparameter werden wegintegriert (Marginal Maximum Likelihood-Schätzmethode). Dazu betrachteten sie die Fähigkeitsparameter als Werte von unabhängigen standardnormalverteilten Zufallsvariablen. Bei Einstellungs- oder Betroffenheitsparametern ist eine symmetrische Verteilung nicht unbedingt angebracht. Entsprechend wird es darum gehen, zu untersuchen, ob es möglich ist, mit empirischen Verteilungen statt der Standardnormalverteilung zu arbeiten. Die diskutierten Schätzmethoden wurden in R (R Development Core Team, 2006) implementiert, um selber Erfahrungen mit den verschiedenen Algorithmen sammeln zu können (R-Paket irt, s. S. 61 ff.).

2 Ein deterministisches Messmodell

In der ersten Hälfte des 20. Jahrhunderts wurden von Psychologen, Sozialpsychologen und Soziologen die Frage der Messung von unsichtbaren Eigenschaften von Personen (latent traits; Einstellungen, Fähigkeiten, Charakterzüge), denen man gewöhnlich mittels Fragebogen näher kommen wollte, zunehmend diskutiert. Diese Diskussionen mündeten in den 40er Jahren unter anderem in die Konstruktion der Guttman-Skala (Guttman scaling, cumulative scale analysis, implicational scale analysis)(Guttman, 1950a, 1950c, 1950b). Eine kurze Darlegung einiger Grundideen hinter der Guttman-Skala beleuchtet die Notwendigkeit der Einführung von probabilistischen Methoden für solche Messungen. In der Folge werden wir i. A. nur von Fähigkeiten sprechen, um die sprachlichen Formulierungen zu vereinfachen. Die jeweiligen Aussagen gelten auch für die Messung von Einstellungsstärken oder von beliebigen Grössen, die irgendwie nach Stärke geordnet werden können.

Bei der Konstruktion einer Guttman-Skala geht man von mehreren dichotomen Items aus, die dieselbe Fähigkeit oder Einstellung erheben. Mit Hilfe der Items sollte eine Skala konstruiert werden, welche die Fähigkeit oder die Einstellung misst. Die Items sind statistische Variablen mit zwei Ausprägungen. Diese werden oft mittels Fragebogen erhoben. Bei den zwei möglichen Antworten auf eine Frage kennzeichnen wir eine der Antworten als die ausgezeichnete. Der ausgezeichneten Antwort wird der Wert 1 zugewiesen, der anderen Antwort der Wert 0. Geht es um

Leistungsmessung, ist die ausgezeichnete Antwort die, welche eine grössere Fähigkeit der Person zum Ausdruck bringt. Geht es um Einstellungsstärken, ist die ausgezeichnete Antwort die, welche eine stärkere Einstellung der Person ausdrückt. Jeder Person i ist ein Fähigkeitsparameter θ_i und jedem Item j ein Schwierigkeitsparameter δ_j ($\theta_i, \delta_j \in \mathbb{R}$) zuzuordnen, die damit auf der Zahlengerade zu liegen kommen. Mit Hilfe der verschiedenen Items muss entsprechend dieselbe, eindimensionale Fähigkeitstärke gemessen werden können. Der Fähigkeitsparameter θ_i einer Person i mit grösseren Fähigkeiten sollten rechts des Fähigkeitsparameters θ_l einer Person l mit weniger Fähigkeiten liegen. Der Schwierigkeitsparameter δ_j des schwierigeren Items j sollten rechts des Schwierigkeitsparameters δ_m des leichteren Items m liegen. Mit

$$\mathbb{N}_n^* := \{m | m \in \mathbb{N} \setminus \{0\} \text{ und } m \leq n\}$$

kann man festlegen:

Definition 1. Seien $n \geq 2$ Items mit paarweise unterschiedlicher Schwierigkeit und N Untersuchungspersonen gegeben, sowie $\mathbb{N}_n^* := \{m | m \in \mathbb{N} \setminus \{0\} \text{ und } m \leq n\}$.

$f : \{\delta_j | j \in \mathbb{N}_n^* \text{ und } \delta_j \in \mathbb{R}\} \cup \{\theta_i | i \in \mathbb{N}_N^* \text{ und } \theta_i \in \mathbb{R}\} \rightarrow \mathbb{R}$ ist eine Guttman-Skala genau dann, wenn

1. die Person i mit dem Fähigkeitsparameter θ_i die ausgezeichnete Antwort auf das Item i mit dem Schwierigkeitsparameter δ_i genau dann gibt, wenn $\theta_i \geq \delta_j$,
2. $f(\theta_i) = f(\delta_j)$ genau dann, wenn $\delta_j \leq \theta_i < \delta_{j+1}$,
3. $f(\theta_i) > f(\delta_j)$ genau dann, wenn $\theta_i \geq \delta_{j+1}$

Die Guttman-Skala ist damit eine Ordinalskala (s. S. 37). Findet man genügend Items mit geeigneten Schwierigkeitsparametern, können mit solchen Skalen Personen mit paarweise verschiedenen Fähigkeiten strikt geordnet werden: jeder Person wird eine andere Zahl zugeordnet und Personen mit grösseren Fähigkeiten werden höhere Zahlen zugeordnet. Das Verfahren läuft darauf hinaus, den Personen die Summe der erreichten Punkte als Skalenwert zuzuordnen. Wegen des ordinalen Charakters der Skala ist jede andere Skala f , welche die Ordnung dieser Summen respektiert, mit einer solchen Punkteskala äquivalent. Insbesondere kann man auch $f(\delta_j) := \delta_j$ festlegen, wodurch bei dieser Festlegung $f(\theta_i) = \delta_j$ genau dann, gilt wenn $\delta_j \leq \theta_i < \delta_{j+1}$.

Auf Grund der Definition liegt ein deterministisches Modell für das Antwortverhalten von Personen vor. Die Person i gibt auf das Item j die ausgezeichnete Antwort mit Wahrscheinlichkeit 1, wenn $\theta_i \geq \delta_j$ und mit Wahrscheinlichkeit 0 wenn $\theta_i < \delta_j$. Liegt eine Guttman-Skala vor und ordnen wir die n Items aufsteigend gemäss Schwierigkeit, so erhalten wir für eine Person i mit Fähigkeitsparameter θ_i den Antwortvektor

$$\begin{cases} \mathbf{0}_n & \text{für } \theta_i < \delta_1 \\ (\mathbf{1}_j, \mathbf{0}_{n-j})^\top & \text{für } \delta_j \leq \theta_i < \delta_{j+1} \\ \mathbf{1}_n & \text{für } \theta_i \geq \delta_n \end{cases} \quad (1)$$

mit $\mathbf{0}_n \in \{0\}^n$, $\mathbf{1}_n \in \{1\}^n$ und $(\mathbf{1}_j, \mathbf{0}_{n-j})^\top$ zusammengesetzt aus den Vektoren $\mathbf{1}_j \in \{1\}^j$ und $\mathbf{0}_{n-j} \in \{0\}^{n-j}$. Wir nennen solche Vektoren „Guttman-Vektoren“. Vektoren \mathbf{v} werden durchgängig als einspaltige Matrizen aufgefasst, wobei statt $\mathbf{v} \in \mathbb{R}^{n \times 1}$ jeweils $\mathbf{v} \in \mathbb{R}^n$ geschrieben wird.

Sei $x_{ij} \in \{0, 1\}$ die Antwort der befragten Person i auf das Item j und \mathbf{x} die (N, n) -Matrix der Antworten von N befragten Personen auf n Items. Für Items l und j mit den Schwierigkeitsparametern $\delta_l > \delta_j$ gilt bei einer Guttman-Skala

$$\sum_{i=1}^N x_{ij} > \sum_{i=1}^N x_{il},$$

wenn sich Untersuchungspersonen entsprechend den Erfordernissen der Skala verhalten und es Personen mit Fähigkeitsparameter θ gibt, so dass $\delta_l > \theta \geq \delta_j$. Ist die Rangordnung der Schwierigkeitsparameter der Items nicht bekannt, kann diese mit Hilfe der Spaltensummen von \mathbf{x} empirisch bestimmt werden.

Werden von Daten die Erfordernisse einer Guttmanskala erfüllt und erstellt man paarweise von den Items Kreuztabellen, so dass das schwierigere Item die Spalten beschriftet, erhält man Kreuztabellen der folgenden Art (s. Tabelle 1):

Antworten auf Items		Item mit δ_j ($\delta_j > \delta_i$)		Randsummen
		0	1	
Item mit δ_i	0	k_{11}	0	k_{11}
	1	k_{21}	k_{22}	$k_{21} + k_{22}$
Randsummen		$k_{11} + k_{21}$	k_{22}	$k_{11} + k_{21} + k_{22}$

Tabelle 1: Gemeinsame Häufigkeitsverteilung zweier Items bei Daten, welche die Erfordernisse einer Guttmanskala erfüllen

In der Forschungspraxis ist das für die Konstruktion einer Guttmanskala erforderliche Verhalten eher selten. Dabei kann man das Problem ungeeigneten Items zuschreiben oder dem inkonsistenten Verhalten der befragten Personen. Sich geeignete Personen auszusuchen, macht wenig Sinn, da man ja vorgegebene Personengruppen untersuchen will. Nach günstigen Items zu suchen, kann - unter der Voraussetzung dass es solche überhaupt gibt - zu langwierig oder zu aufwendig sein. Will man die Fähigkeiten der beteiligten Personen bei vorliegenden Items und gegebenen Antwortvektoren messen, muss man deshalb Verfahren entwickeln, die trotzdem zu vertretbaren Zuschreibungen von Skalenwerten führen. Dazu wurden in der Literatur verschiedene Vorschläge gemacht, die Zuordnungsvorschriften beinhalten, um „fehlerhafte“ Vektoren in Guttman-Vektoren zu verwandeln (Mokken, 1971, S. 41 ff.):

- So wurde etwa versucht, die Zuordnung mittels einer minimalen Anzahl von Änderungen zu definieren, welche den „fehlerhaften“ Vektor in einen Guttman-Vektor überführen. Einerseits bleibt dadurch die Summe der korrekten Antworten nicht unbedingt erhalten - eine vermutlich nicht harmlose Manipulation der Ergebnisse. Andererseits ist es schwierig, zu eindeutigen Zuordnungen zu gelangen (Mokken, 1971, S. 45).
- Man könnte die Einsen im „fehlerhaften“ Vektor nach links schieben. Allerdings werden dadurch Vektoren als gleichwertig behandelt, die es vermutlich nicht wirklich sind. Der Vorschlag läuft darauf hinaus, den Individuen die Summe $\sum_{i=1}^n x_{ij}$ als Skalenwert zuzuordnen, unabhängig davon, wie die Summe zu Stande kommt.

Das Hauptproblem der verschiedenen möglichen Zuordnungsvorschriften ist, dass die Vorschläge ad hoc entwickelt werden, nicht im Mess-Modell abgestützt und damit ziemlich willkürlich sind. Von gewöhnlichen Zuordnungsverfahren (z. B. Zusammenzählen von Punkten) unterscheidet sich die Guttmanskala allerdings durch die Möglichkeit, die Eignung von Items für die Skalierung von Untersuchungsobjekten zu überprüfen, wie die folgenden Überlegungen zeigen.

Items sollten einerseits unterschiedlich schwierig sein, um eine Bandbreite von unterschiedlichen Fähigkeiten abzudecken. Die Bandbreite kann mit den oben erwähnten Spaltensummen der Matrix \mathbf{x} überprüft werden. Andererseits sollten sie dieselbe Fähigkeitsdimension messen, also genügend „homogen“ sein. Um die Homogenität von Items zu überprüfen, wurden verschiedene Verfahren vorgeschlagen. Guttman (1950a, S. 77) definierte den folgenden Homogenitäts-Index: es wird die Anzahl der „Fehler“ in den Antwortvektoren an der möglichen Anzahl Fehler gemessen und dieser Quotient von 1 abgezählt. Es stellt sich das Problem, wie Fehler zu definieren sind. Dazu wurden ebenfalls verschiedene Vorschläge gemacht (Mokken, 1971, S. 41 ff.). Guttman (1950a, S. 77) definiert einen Fehler als eine Nichtübereinstimmung von Komponenten im Antwortvektor \mathbf{x} , in dem die Items der Schwierigkeit gemäss aufsteigend geordnet sind, mit Komponenten im entsprechenden Guttman-Vektor \mathbf{x}^g , der durch Verschieben der Einsen nach links erreicht wird. Die Anzahl der Fehler pro Antwortvektor der Person i bei n Items ist damit

$$\sum_{j=1}^n 1\{x_{ij} \neq x_{ij}^g\}$$

Die grösstmögliche Anzahl Fehler beträgt Nn (N Anzahl befragter Personen) und wir erhalten den von Guttman mit „rep“ bezeichneten Index:

$$rep := 1 - \frac{\sum_{i=1}^N \sum_{j=1}^n 1\{x_{ij} \neq x_{ij}^g\}}{Nn}$$

„rep“ steht dabei für „reproducibility“. Es geht darum, in welchem Ausmass man auf Grund der Skalenwerte die Antwortvektoren reproduzieren kann. Eine völlige Reproduzierbarkeit besteht, wenn nur Guttmanvektoren vorkommen und damit eine Eins-zu-Eins-Beziehung zwischen Klassen von Personen mit dem selben Guttman-Antwortvektor und dem den Personen der Klasse zugeordneten Skalenwert besteht.

Wird die Schwierigkeits-Ordnung der Items mit Hilfe der Stichprobe durch die oben erwähnten Spaltensummen festgelegt, so ist ersichtlich, dass die grösstmögliche Anzahl Fehler Nn nicht erreicht werden kann. rep ist dann nicht ein Index, der alle Werte $\frac{i}{Nn}$, $i \in \mathbb{N}_{Nn}$, zwischen 0 und 1 annehmen kann, und die grössere Nähe zu 1 als zu 0 drückt nicht unbedingt aus, dass die Itematterie gut gewählt ist. Deshalb wurde ziemlich viel Aufwand betrieben, um einen Index zu entwickeln, der den Bereich ausschöpft (Mokken, 1971, S. 49 ff.). Da die Ersetzung von Nn in der obigen Formel durch eine kleinere Zahl eine monotone Transformation der Indexwerte bewirkt, ist dieser Aufwand allerdings nicht wirklich verständlich. Guttman selbst legte auf Grund praktischer Erfahrungen fest, dass für eine akzeptable Itematterie rep grösser als 0.9 zu sein hat, legte aber noch weitere Kriterien für die Güte einer Itematterie fest, auf die hier nicht eingegangen wird (Guttman, 1950a, S. 79).

3 Probabilistische Mess-Modelle

Auf Grund der unbefriedigenden Behandlung von Antwortverhalten, das von Guttman-Vektoren abweicht, drängen sich Modelle auf, die nicht von einem deterministischen Antwortmodell ausgehen, sondern von einem probabilistischen. Damit verschwindet das Problem der „fehlerhaften“ Antwortvektoren. Die Antwortvektoren werden als Werte von Zufallsvektoren betrachtet, die unter der Annahme unbekannter Itemparameter zustande kommen. Auf Grund der beobachtbaren, manifesten Antwortvektoren geht es darum, die unsichtbaren, latenten Parameter zu schätzen - die Terminologie von „manifest“ und „latent“ wurde von Lazarsfeld (1950, S. 363) im Rahmen seiner „Latent Structure Analysis“ eingeführt. Probabilistische Modelle fürs Messen von latenten Eigenschaften wurden bereits vor der Publikation der Guttman-Skala veröffentlicht (Lawley, 1944). Im Sammelband, in dem die Guttman-Skala vorgestellt wird, wird auch die mit IRT-Methoden verwandte „Latent Structure Analysis“ von Lazarsfeld (1950) dargelegt.

Um die Parameter zu schätzen, braucht man Wahrscheinlichkeitsmodelle, die man allgemein wie folgt beschreiben kann:

- Man betrachtet die $(\Omega, \mathfrak{A}) \rightarrow (\mathbb{R}, \mathfrak{B})$ -messbaren Zufallsvariablen X_{ij} , die als Funktionswerte (= Realisationen) die Antworten der Person i bezüglich des Items j annehmen, wobei \mathfrak{A} eine σ -Algebra auf Ω und \mathfrak{B} die Borel- σ -Algebra auf \mathbb{R} ist. In dieser Arbeit wird $X_{ij} : (\Omega, \mathfrak{A}) \rightarrow (\{0, 1\}, \mathcal{P}(\{0, 1\}))$ gesetzt ($\mathcal{P}(A)$ für die Potenzmenge von A). Weiterhin sei $\mathbf{X}_i := (X_{i1}, \dots, X_{in})^\top$ der Zufallsvektor, welcher als Realisationen die Antworten der Person i auf n Items annimmt.
- Die Wahrscheinlichkeitsverteilung der Zufallsvektoren \mathbf{X}_i hängt von den Fähigkeits- und Itemparametern ab. Dabei kann man den Fähigkeitsparameter θ_i der i -ten Person als Realisation der $\mathfrak{A} - \mathfrak{B}$ -messbaren Zufallsvariable $\Theta_i : (\Omega, \mathfrak{A}) \rightarrow (\mathbb{R}, \mathfrak{B})$ betrachten - z. B. wenn man eine Zufalls-Stichprobe aus einer Population zieht. Für manche Überlegungen ist es zudem nützlich, die Items als Stichprobe aus einer Sammlung von Items zu betrachten. Damit sind die Itemparameter δ_j des Items j als Realisationen von $(\Omega, \mathfrak{A}) \rightarrow (\mathbb{R}^k, \mathfrak{B}^k)$ -messbaren Zufallsvektoren Δ_j für k Typen von Itemparametern zu betrachten. Diese Sichtweise der

Parameter als Werte von Zufallsgrößen wird auch bei der Verwendung Bayesscher Schätz-Methoden benötigt. Hier werden zwei Typen von Item-Parametern berücksichtigt werden: Trennschärfe- und Schwierigkeitsparameter, d.h. $k = 2$.

- Sei \mathbb{P} die gemeinsame Verteilung von $(\mathbf{X}, \Theta, \Delta)$ mit $\mathbf{X} = (\mathbf{X}_1, \dots, \mathbf{X}_N)^\top$, $\Theta = (\Theta_1, \dots, \Theta_N)^\top$ und $\Delta = (\Delta_1, \dots, \Delta_n)^\top$, wobei N die Anzahl der Personen ist. Man betrachtet die bedingte Verteilung $\mathbb{P}(\mathbf{X}|\Theta, \Delta)$ der Zufallsmatrix \mathbf{X} bezüglich \mathbb{P} unter Θ und Δ .
- Man geht davon aus, dass die Zufallsvektoren Θ und die Zufallsmatrizen Δ voneinander unabhängig sind, dass die Komponenten dieser Zufallsgrößen voneinander unabhängig sind und dass die Zufallsvariablen X_{i1}, \dots, X_{in} , welche als Werte die Antwort der Person i auf die verschiedenen Items $j \in \mathbb{N}_n^*$ annehmen, für gegebene θ_i und $\delta \in \mathbb{R}^{n \times k}$ unabhängig sind. Dies wird lokale oder bedingte Unabhängigkeit genannt. Entsprechend gilt

$$\mathbb{P}(\mathbf{X}_i = \mathbf{x}_i | \Theta_i = \theta_i, \Delta = \delta) = \prod_{j=1}^n \mathbb{P}(X_{ij} = x_{ij} | \Theta_i = \theta_i, \Delta_j = \delta_j),$$

wobei $\mathbb{P}(X_{ij} = x_{ij} | \Theta_i = \theta_i, \Delta_j = \delta_j)$ die Wahrscheinlichkeit ist, dass die Zufallsvariable X_{ij} für gegebene θ_i und δ_j den Wert x_{ij} annimmt.

- Zudem geht man davon aus, dass das Antwortverhalten der verschiedenen Personen voneinander unabhängig ist. Entsprechend gilt:

$$\mathbb{P}(\mathbf{X} = \mathbf{x} | \Theta = \theta, \Delta = \delta) = \prod_{i=1}^N \mathbb{P}(\mathbf{X}_i = \mathbf{x}_i | \Theta_i = \theta_i, \Delta = \delta)$$

Auf diesem allgemeinen Hintergrund möchte man bei gegebener Antwortdatenmatrix \mathbf{x} (Realisation von \mathbf{X}) mit den Zeilenvektoren $\mathbf{x}_i = (x_{i1}, \dots, x_{in})$ die Parameter $(\theta_1, \dots, \theta_N)^\top$ und $(\delta_1, \dots, \delta_n)^\top$ schätzen.

Betrachtet man die Funktion

$$ICC(\theta) := \mathbb{E}(X | \Theta = \theta, \Delta = \delta),$$

so erhält man für jedes $\theta \in \mathbb{R}$ und $\delta \in \mathbb{R}^k$ den Erwartungswert der Zufallsvariable X unter der Bedingung, dass $\Theta = \theta$ und $\Delta = \delta$, wobei X mit $X(\omega) \in \mathbb{R}$ eine Zufallsvariable ist, welche die möglichen Antworten auf das Item j als Werte annimmt. Diese Regressions-Funktionen werden „Charakteristische Item-Kurven“ (*ICC*, item characteristic curve, item characteristic function) oder „trace function“ (Mokken, 1971, S. 79) genannt. Jedem Item entspricht eine solche Funktion. Die *ICC*s sollten monoton steigende Funktionen sein, wenn die Antwortmöglichkeiten aufsteigend gemäss Güte der möglichen Antworten codiert sind. Erfüllt eine Item-Batterie diese Eigenschaft für jedes Item, wird sie „schwach homogen“ genannt (Mokken, 1971, S. 115 ff). Neben der Forderung nach schwacher Homogenität scheint eine solche nach stärkerer Homogenität sinnvoll zu sein: *ICC*s verschiedener Items sollten sich nicht schneiden. Schneiden sich die *ICC*s zweier Items i und l , gibt es ein Intervall, wo $ICC_j(\theta) < ICC_l(\theta)$ und ein Intervall, wo $ICC_j(\theta) > ICC_l(\theta)$, d.h. für manche Personen ist das erste Item schwieriger als das zweite und für andere ist das erste Item einfacher als das zweite - je nach dem Bereich, wo die Fähigkeitsparameter der Personen liegen. Auszuschliessen ist ein solcher Fall allerdings nicht: es ist z. B. möglich, dass eine fähigere Person bei einer Aufgabe mehr Probleme sieht als eine andere und es für diese Person schwieriger ist, die korrekte Antwort zu finden. Erfüllt eine Itembatterie die Bedingung, dass sich die *ICC*s paarweise nicht schneiden, so spricht man von einer „stark homogenen“ oder „holomorphen“ (Mokken, 1971, S. 115 ff) Itembatterie.

Die Steigung einer *ICC* in θ drückt die Trennschärfe des entsprechenden Items in θ aus: ist die Steigung in θ gross, so wirken sich kleine Unterschiede in einer Umgebung von θ in relativ grossen Unterschieden bezüglich der erwarteten Güte der Antwort aus. Solche Items unterscheiden entsprechend in einer Umgebung von θ , wo die *ICC* eine grosse Steigung hat, gut zwischen Individuen mit nahe beieinander liegenden Fähigkeiten.

Definition 2. Sei die ICC_j des Items j differenzierbar in θ . Die Trennschärfe des Items j in θ ist die Steigung der ICC_j in θ .

Auf diesem recht allgemeinen Hintergrund werden nun spezifischere Modelle betrachtet, welche unter die obige Darstellung subsumiert werden können. Es gilt, die Antwortmöglichkeiten (d. h. das Bild der Zufallsvariablen X_{ij}), konkrete ICC s und eventuell die Wahrscheinlichkeitsverteilungen von Θ und Δ festzulegen.

In der Folge werden wir uns mit ICC s beschäftigen, für die $0 \leq ICC(\theta) \leq 1$ gilt. Im Falle dichotomer Antwortmöglichkeiten, die mit 0 und 1 kodiert werden, drückt der Wert $ICC_j(\theta)$ die Wahrscheinlichkeit aus, dass eine Person mit dem Fähigkeitsparameter θ auf das Item mit δ_j die korrekte Antwort gibt. Je grösser die Fähigkeit einer Person, desto höher sollte die Wahrscheinlichkeit sein, dass sie die korrekte Antwort gibt. Die ICC hat gemäss dieser Anforderung für den dichotomen Fall die Form der Verteilungsfunktion stetig verteilter Zufallsvariablen. Da ICC s mit $0 \leq ICC(\theta) \leq 1$ zwischen 0 und 1 liegen und monoton steigend sind, wird eine starke Trennschärfe in einem Intervall durch eine schwache in anderen Intervallen erkauft. Für solche ICC s ist es sinnvoll, die Schwierigkeit des Items mit dem Median der entsprechenden Verteilungsfunktion zu identifizieren:

Definition 3. $\beta_j \in \mathbb{R}$ ist die Schwierigkeit des Items j mit $0 \leq ICC_j(\theta) \leq 1$ (für $\theta, \beta_j, \beta \in \mathbb{R}$) genau dann, wenn $\beta_j = \inf\{\beta | ICC_j(\beta) \geq 0.5\}$.

Unter den vielen möglichen Modellen, die sich durch entsprechende Spezifikationen ergeben, betrachten wir das logistische Zwei-Parameter-Modell von Birnbaum (1968) mit dichotomen Antwortmöglichkeiten auf die Items sowie mit fixen aber unbekanntem Schwierigkeits- und Fähigkeitsparametern. Es sind passende Wahrscheinlichkeitsmodelle und Schätzmethoden einzuführen sowie allfällige Schätz-Probleme zu diskutieren.

4 Das logistische Zwei-Parameter-Modell

Das Zwei-Parameter-Modell von Birnbaum (1968, S. 399, ff.) geht von dichotomen Antwortmöglichkeiten auf die Items aus und legt für jedes Item eine logistische Funktion als ICC zu Grunde. Bis auf die Konstante 1.702 weist sie die folgende Form auf, die künftig „logistische ICC “ genannt wird (s. Abbildung 1):

Definition 4.

$$p : \mathbb{R} \rightarrow]0, 1[\tag{2}$$

$$p(\theta) := \frac{\exp(\alpha(\theta - \beta))}{1 + \exp(\alpha(\theta - \beta))} \text{ mit } \alpha, \beta \in \mathbb{R} \text{ und } \alpha > 0$$

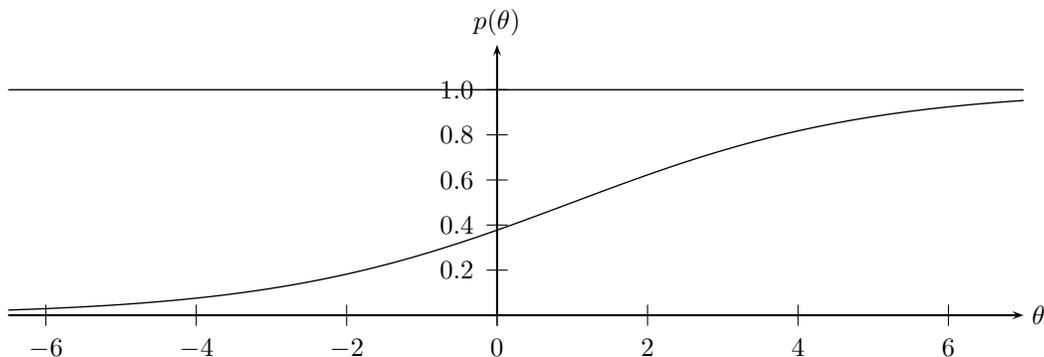


Abbildung 1: Logistische ICC mit den Parametern $\alpha = 0.5$ und $\beta = 1$

Die Verwendung der oben erwähnten und hier nicht berücksichtigten Konstante hat vor allem historische Gründe: ursprünglich wurden für *ICCs* zur Beschreibung dichotomer Antwortmöglichkeiten Verteilungsfunktionen normalverteilter Zufallsvariablen gewählt. Durch die Konstante werden die logistischen Funktionen den Verteilungsfunktionen solcher Zufallsvariablen angeglichen: es gilt z. B. für alle $\theta \in \mathbb{R} : |\Phi(\theta) - p(1.702 \cdot \theta)| < 0.01$ mit der Verteilungsfunktion der Standardnormalverteilung Φ und p von (2) mit $\alpha = 1$ und $\beta = 0$ (s. Birnbaum (1968, S. 399)).

Es werden nun ein paar Eigenschaften logistischer *ICCs* angeführt - für Beweise einiger dieser Aussagen s. Seite 41.

- $\frac{\exp(\alpha(\theta-\beta))}{1+\exp(\alpha(\theta-\beta))} = \frac{1}{1+\exp(\alpha(\beta-\theta))}$
- Logistische *ICCs* sind strikt monoton steigend für $\alpha > 0$.
- $p(\theta) = 0.5$ genau dann, wenn $\theta = \beta$. Damit drückt der Parameter β die Schwierigkeit des Items aus.
- Die *ICC* nimmt die grösste Steigung in $\theta = \beta$ an. Die logistische *ICC* weist die höchste Trennschärfe also in ihrem Schwierigkeitsparameter auf. Sie hat dort die Steigung $\frac{\alpha}{4}$. Damit ist α proportional zur grössten Trennschärfe, die das Item annehmen kann. Wir nennen deshalb künftig α den Trennschärfeparameter des Items.
- Die Kurve ist punktsymmetrisch in $(\beta, 0.5)$.
- Es gilt $0 < p(\theta) < 1$ für $\theta \in \mathbb{R}$ und $\lim_{\theta \rightarrow \infty} p(\theta) = 1$ und $\lim_{\theta \rightarrow -\infty} p(\theta) = 0$ für $\alpha > 0$.
- Es gilt

$$\ln\left(\frac{p(\theta)}{1-p(\theta)}\right) = \alpha(\theta - \beta), \quad (3)$$

d.h. die logarithmierte Chance, eine korrekte Antwort zu liefern, ist eine affine Funktion der Fähigkeiten. Die Steigung der affinen Funktion ist die Trennschärfe und der Ordinatenabschnitt das mit -1 multiplizierte Produkt von Schwierigkeit und Trennschärfe des Items. Die logarithmierte Chance wird auch „Log-Odds“ oder „Logit“ genannt.

- Logistische *ICCs* von Items i und j schneiden sich genau dann, wenn $\alpha_i \neq \alpha_j$ für $i \neq j$. Damit sind Itembatterien mit unterschiedlichen α_j nicht stark homogen. Weisen die Items ähnliche α_j auf oder liegen deren Schwierigkeitsparameter genügend weit auseinander, ergeben sich für die Praxis keine Probleme. Bereiche, wo sich die Schwierigkeiten für Probanden umkehren, liegen dann so, dass die entsprechenden Wahrscheinlichkeitsunterschiede zwischen den Items für die Praxis vernachlässigbar werden (s. Abbildung 2).

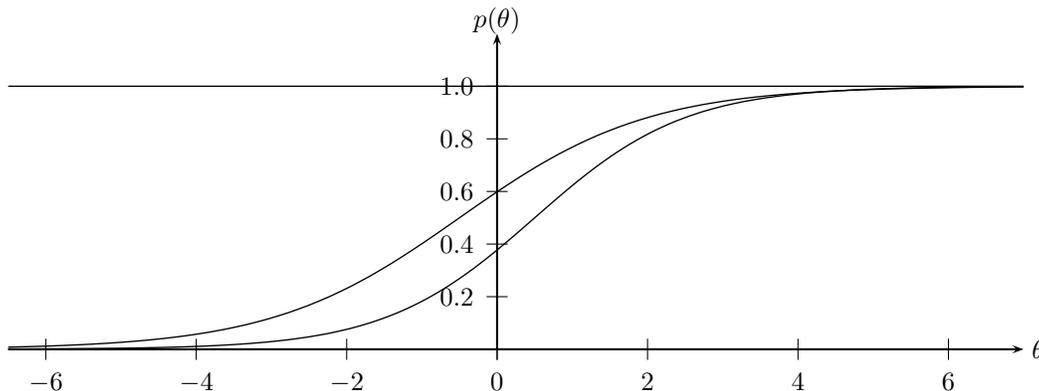


Abbildung 2: Logistische *ICCs* mit den Parametern $\alpha = 1$ und $\beta = -0.5$ sowie $\beta = 0.5$ und $\alpha = 0.8$

Bei logistischen *ICCs* wird gemäss Definition vorausgesetzt, dass die Fähigkeitsparameter von Personen reelle Zahlen sind und damit auf der reellen Zahlengerade liegen. Damit entspricht die

logistische *ICC* einer logistischen Regressionskurve. Im Gegensatz zur logistischen Regression (s. (Hosmer & Lemeshow, 2000)) wird man aber in der Anwendung die Fähigkeiten von Personen nicht kennen. Es wird in der Folge darum gehen, Methoden zu entwickeln, um die Schwierigkeitsparameter der Items zusammen mit den Fähigkeitsparametern der Antwortvektoren von Personen zu schätzen. In diesem Zusammenhang sind geeignete Wahrscheinlichkeitsmodelle für die Zufallsvariablen X_{ij} , die als Werte die Antworten der i -ten Person auf das j -te Item annehmen, zu entwickeln.

Dazu starten wir mit den einfacheren Problemen der Schätzung der Itemparameter bei gegebenen Fähigkeitsparametern und der Schätzung der Fähigkeitsparameter bei gegebenen Itemparametern. Anschliessend wird eine Schätzmethode eingeführt, die beide Parametertypen gemeinsam schätzt: die *Joint Maximum Likelihood-Schätzmethode* (*JMLE* - „E“ für „Estimation“). Deren Probleme werden zur *Marginal Maximum Likelihood-Schätzmethode* (*MMLE*) führen, welche die Itemparameter unter der Voraussetzung einer spezifischen Verteilung der Fähigkeitsparameter schätzt. Diese Verteilung nennen wir „a priori Verteilung der Fähigkeitsparameter“. Die *MMLE* Methode stellt heute weitgehend die Standardschätzmethode in der IRT dar. Es stellt sich die Frage, wie die a priori Verteilungen zu bestimmen sind. Während bei Fähigkeiten symmetrische Verteilungen i. A. sinnvoll sind und entsprechend die Voraussetzung einer Normalverteilung bei genügender Dominanz der Daten ziemlich problemlos ist, müssen Einstellungs- oder Betroffenheitsstärken in einer Population keineswegs symmetrisch verteilt sein. Es stellt sich die Frage, wie man beim Vorliegen asymmetrischer Verteilungen eine günstigere Verteilung aus den Daten schätzen könnte.

Das *MMLE*-Verfahren wird in der IRT-Literatur nicht den bayesschen Methoden zugerechnet, obwohl die Schätzungen mit Hilfe einer a priori Verteilung vorgenommen werden und der Satz von Bayes eine Rolle spielt. Von bayesschen Schätzverfahren wird erst gesprochen, wenn auch für die „strukturellen“ Parameter (hier die Item-Parameter) und nicht nur für die „Stör-Parameter“ („nuisance-parameter“, - hier die Fähigkeitsparameter) a priori Verteilungen vorausgesetzt werden. Bayessche Schätzmethoden wurden in der IRT etwa von Swaminathan und Gifford (1982), Mislevy (1986) und Rigdon und Tsutakawa (1983) eingeführt und liegen mittlerweile in verschiedenen Varianten vor (s. z. B. Baker und Kim (2004, Kapitel 7, S. 177 ff.) und Fox (2010)). Computereperimente scheinen zu zeigen, dass Bayessche Schätzmethoden stabil und vor allem für kleine Itembatterien präziser als die *MMLE*-Methode sind, sofern die a priori Verteilungen gut gewählt sind. Liegen Methoden vor, mit Hilfe derer man nachträglich die Güte des Modells oder die Verträglichkeit der faktischen Antwortematrix mit - mit Hilfe der geschätzten Parameter - simulierten Antwortematrizen zeigen kann, lässt sich die Wahl der a priori Verteilungen im Nachhinein rechtfertigen. Hier wird aber eine Konzentration auf einige *MMLE*-Verfahren erfolgen.

4.1 Schätzen der Itemparameter bei gegebenen Fähigkeiten

Sind die Fähigkeiten von Personen bekannt und als auf einer Intervallskala messbar vorausgesetzt, so kann man die Parameter α und β der logistischen *ICC* mit den bekannten Maximum-Likelihood-Methoden (*ML*-Methoden) für die logistische Regression schätzen (s. für entsprechende Methoden z. B. Agresti (1990, S. 112 ff.) und Baker und Kim (2004, S. 38 ff.)). Setzt man nämlich

$$\begin{aligned} a &:= \alpha \\ b &:= -\alpha\beta, \end{aligned} \tag{4}$$

so wird aus

$$\begin{aligned} \alpha(\theta - \beta) &= \alpha\theta - \alpha\beta \\ &= a\theta + b, \end{aligned}$$

wobei die Form der logistischen Regressionsfunktion durch

$$\begin{aligned} \pi &: \mathbb{R} \rightarrow]0, 1[\\ \pi(\theta) &:= \frac{\exp(a\theta + b)}{1 + \exp(a\theta + b)} \end{aligned} \tag{5}$$

gegeben ist. Als Ausgangspunkt für das bei ML-Methoden benötigte Wahrscheinlichkeitsmodell betrachtet man relative Häufigkeiten, mit denen Personen mit spezifischen Fähigkeiten θ_i die Frage richtig beantworten - $i \in \mathbb{N}_m^*$ mit der Anzahl m von Klassen von Personen mit unterschiedlichen Fähigkeitsparametern. Geeignet ist entsprechend die Betrachtung der binomialverteilten Statistik X_i , welche die Anzahl x_i der korrekten Antworten pro Fähigkeitsklasse θ_i als Werte annimmt. Die Parameter der Verteilung sind N_i (Anzahl der Personen mit dem Fähigkeitsparameter θ_i) und $\pi(\theta_i)$ (Wahrscheinlichkeit, dass eine Person mit dem Fähigkeitsparameter θ_i die korrekte Antwort gibt). Die Wahrscheinlichkeitsfunktion (W-Funktion) f_w der gemeinsamen Verteilung der m binomialverteilten Statistiken X_i ist dann unter der Voraussetzung der Unabhängigkeiten dieser Statistiken

$$f_w : \prod_{i=1}^m \mathbb{N}_{N_i} \rightarrow [0, 1]$$

$$f_w(x_1, \dots, x_m | \boldsymbol{\theta}) := \prod_{i=1}^m \binom{N_i}{x_i} \pi(\theta_i)^{x_i} (1 - \pi(\theta_i))^{N_i - x_i}.$$

mit $\boldsymbol{\theta} = (\theta_1, \dots, \theta_m)^\top$. Dies ergibt mit (5) die folgende Likelihood-Funktion f in den Variablen a und b , wobei die Binomialkoeffizienten $\binom{N_i}{x_i}$ weggelassen werden. Sie sind bei gegebenen N_i und x_i konstant und fallen deshalb bei der üblichen Suche nach Maxima weg.

$$f : \mathbb{R}^2 \rightarrow]0, 1]$$

$$f(a, b) := \prod_{i=1}^m \pi(a, b)^{x_i} (1 - \pi(a, b))^{N_i - x_i}$$

$$= \prod_{i=1}^m \left(\frac{\exp(a\theta_i + b)}{1 + \exp(a\theta_i + b)} \right)^{x_i} \left(1 - \frac{\exp(a\theta_i + b)}{1 + \exp(a\theta_i + b)} \right)^{N_i - x_i}$$

Durch Logarithmieren erhält man die Log-Likelihood-Funktion (LL-Funktion):

$$\ln f : \mathbb{R}^2 \rightarrow [0, -\infty[$$

$$\ln f(a, b) := \sum_{i=1}^m x_i \ln \pi(a, b) + (N_i - x_i) \ln(1 - \pi(a, b))$$

$$= \sum_{i=1}^m x_i \ln \frac{\exp(a\theta_i + b)}{1 + \exp(a\theta_i + b)} + \sum_{i=1}^m (N_i - x_i) \ln \left(1 - \frac{\exp(a\theta_i + b)}{1 + \exp(a\theta_i + b)} \right)$$

Zu suchen sind die Parameter \hat{a} und \hat{b} , die diese Funktion bei gegebenen x_i und N_i maximieren. Die Maximierung erfolgt mit üblichen iterativen Näherungsmethoden wie etwa dem Newton-Raphson-Verfahren (s. Anhang, S. 43 und Agresti (1990, S. 112 ff.), bezüglich Existenz und Eindeutigkeit der ML-Schätzer s. Dümbgen (2009, S. 109))

Es ergeben sich die geschätzten Werte für die Parameter a und b in

$$\ln \left(\frac{\pi(\theta)}{1 - \pi(\theta)} \right) = a\theta + b.$$

Die gesuchten Parameter der logistischen ICC können nun zurückgerechnet werden. Die Parameter sind nicht schätzbar, wenn ein Item nur korrekte oder nur falsche Antworten erhält. Solche Items sind aus den Daten zu streichen.

4.2 Schätzen der Fähigkeiten bei gegebenen Item-Parametern

Unter der Voraussetzung, dass die Item-Parameter von $j \in \mathbb{N}_n^*$ Items mit logistischen ICCs bekannt sind, möchten man nun die Fähigkeitsparameter von $i \in \mathbb{N}_N^*$ Personen auf dem Hintergrund deren Antworteverhaltens bezüglich der n Items schätzen. Die Antworten ergeben für die Person

i einen Antwortvektor (x_{i1}, \dots, x_{in}) , mit $x_{ij} \in \{0, 1\}$ und 0 für „falsch“ sowie 1 für „wahr“. Die Ergebnisse x_{ij} betrachten wir dabei als Werte von bernoulli-verteilten Zufallsvariablen X_{ij} . Die Fähigkeitsparameter θ_i einer jeden Person i werden separat geschätzt - unter der Voraussetzung, dass das Antwortverhalten der verschiedenen Personen voneinander unabhängig ist. Die Wahrscheinlichkeit, dass die Person i auf das Item j bei einer Fähigkeit von θ_i die Antwort x_{ij} gibt, ist

$$\mathbb{P}(X_{ij} = x_{ij} | \Theta_i = \theta_i, \Delta_j = \delta_j) = (p_j(\theta_i))^{x_{ij}} (1 - p_j(\theta_i))^{1-x_{ij}}$$

mit $\delta_j = (\alpha_j, \beta_j)^\top$ und

$$p_j(\theta_i) = \frac{\exp(\alpha_j(\theta_i - \beta_j))}{1 + \exp(\alpha_j(\theta_i - \beta_j))}$$

(s. Seite 8, Formel (2)). Die gemeinsame Verteilung unter der Voraussetzung, dass das Antwortverhalten der Person i auf die n verschiedenen Items unabhängig ist, ist dann mit $\delta = ((\alpha_1, \beta_1)^\top, \dots, (\alpha_n, \beta_n)^\top)^\top$:

$$\mathbb{P}(\mathbf{X}_i = \mathbf{x}_i | \Theta_i = \theta_i, \Delta = \delta) = \prod_{j=1}^n (p_j(\theta_i))^{x_{ij}} (1 - p_j(\theta_i))^{1-x_{ij}}$$

Damit erhält man die Likelihood-Funktion bezüglich des Parameters θ_i :

$$f(\theta_i) = \prod_{j=1}^n (p_j(\theta_i))^{x_{ij}} (1 - p_j(\theta_i))^{1-x_{ij}}$$

und die LL-Funktion

$$\ln f(\theta_i) = \sum_{j=1}^n [x_{ij} \ln(p_j(\theta_i)) + (1 - x_{ij}) \ln(1 - p_j(\theta_i))].$$

Das näherungsweise Maximum der LL-Funktion wird wiederum mit den üblichen Näherungsverfahren berechnet. Ähnlich wie bei der Schätzung der Item-Parameter ist zu beachten, dass bei diesem Verfahren Personen, die alle Fragen richtig oder alle Fragen falsch beantworten, aus dem Datensatz gestrichen werden müssen. Während dies bei Items kein Problem darstellt - Items die nicht diskriminieren, sind für einen Test nicht geeignet -, ist die Entfernung bei Antwortvektoren problematischer. Man möchte ja nicht darauf verzichten, die Fähigkeiten von Personen zu messen, die sehr gut oder sehr schlecht abschneiden. Verfahren, die diese Streichung nicht notwendig machen, wären vorzuziehen. Zu beachten ist zudem, dass die Schätzung auf Grund relativ weniger Daten vorgenommen wird. Bei n Items liegen jeweils n Daten und die Parameter für die Schätzung der Fähigkeiten vor. Entsprechend wird die Schätzung der Fähigkeitsparameter durch die Hinzunahme von Personen allein nicht präziser - abgesehen von der Verminderung der Varianz der Parameter-Zufallsvektoren, die bei der Fähigkeitsparameterschätzung verwendet werden.

Man kann zeigen, dass bei logistischen ICCs

$$\sum_{j=1}^n \alpha_j x_{ij}$$

eine suffiziente Statistik für θ_i ist (s. Seite 50). Die Statistik ist allerdings von den zu schätzenden Trennschärfeparametern α_j abhängig. Entsprechend kann man Fähigkeiten einer Person auch mit Hilfe des Antwortvektors der Person schätzen: die Fähigkeit ist die Summe der mit den Trennschärfeparametern gewichteten Antworten auf die Items. Die Lage der Schwierigkeitsparameter spielt entsprechend keine Rolle, wirkt sich aber indirekt mittels der Wahrscheinlichkeit aus, dass x_{ij} 0 oder 1 ist.

4.3 Gemeinsames Schätzen der Fähigkeiten und der Item-Parameter

Gewöhnlich sind weder die Item- noch die Fähigkeitsparameter bekannt. Sie müssen entsprechend alle geschätzt werden. Es wird zuerst die Joint Maximum Likelihood-Schätzmethode (JMLE) eingeführt, die von Birnbaum (1968, S. 420 ff.) vorgeschlagen wurde. Es liegen $i \in \mathbb{N}_N^*$ Personen und $j \in \mathbb{N}_n^*$ Items mit den Antwortmöglichkeiten $x_{ij} \in \{0, 1\}$ vor, wobei die x_{ij} als Werte von $N \cdot n$ bernoulli-verteiltern Zufallsvariablen X_{ij} zu betrachten sind. Die gemeinsame Verteilung unter der Voraussetzung der Unabhängigkeit dieser Zufallsvariablen ist

$$\prod_{i=1}^N \mathbb{P}(\mathbf{X}_i = \mathbf{x}_i | \Theta_i = \theta_i, \mathbf{\Delta} = \mathbf{\delta}) = \prod_{i=1}^N \prod_{j=1}^n (p_j(\theta_i))^{x_{ij}} (1 - p_j(\theta_i))^{1-x_{ij}},$$

wobei $p_j(\theta_i)$ die Wahrscheinlichkeit einer korrekten Antwort der Person i beim Vorliegen der logistischen ICC des Items j ist. Daraus erhält man die LL-Funktion in $2n + N$ Parametern

$$\ln f(\alpha_1, \dots, \alpha_n, \beta_1, \dots, \beta_n, \theta_1, \dots, \theta_N) = \sum_{i=1}^N \sum_{j=1}^n x_{ij} \ln p_j(\alpha_j, \beta_j, \theta_i) + (1 - x_{ij}) \ln(1 - p_j(\alpha_j, \beta_j, \theta_i)).$$

Wegen der vorausgesetzten Unabhängigkeit der Fähigkeits-Parameter bei bekannten Item-Parametern kann man die Fähigkeits-Parameter einzeln schätzen. Ebenso kann man die Item-Parameter bei bekannten Fähigkeits-Parametern pro Item schätzen. Diese Überlegungen führen zum folgenden, von Birnbaum vorgeschlagenen iterativen Verfahren, um die Parameter zu schätzen (Birnbaum, 1968, S. 420). Man startet mit einem grob geschätzten Fähigkeitsparametervektor der untersuchten Personen. Man berechnet die Itemparameter, dann wiederum die Fähigkeitsparameter. Mit Hilfe der Fähigkeitsparameter werden erneut die Itemparameter berechnet, etc., bis sich die Änderungen bei den Iterationen höchstens um einen festzulegenden Betrag unterscheiden. Als Start-Fähigkeitsvektor könnte man z. B. für jede Person i die Summe

$$x_i := \sum_{j=1}^n x_{ij}$$

der erreichten Punkte x_{ij} der n Items berechnen und diese Summen standardisieren. Unter Standardisierung des Datums x_i wird die Berechnung von $z_i := \frac{x_i - \bar{x}}{s}$ mit dem Mittelwert \bar{x} und der Standardabweichung s eines Datenvektors $\mathbf{x} := (x_1, \dots, x_i, \dots, x_k)^\top$ verstanden. Am Schluss des Iterationsverfahrens werden die Fähigkeitsparameter standardisiert und die Itemparameter auf die derart gebildete Skala umgerechnet.

Die zu schätzenden Parameter sind ohne Verankerung auf der reellen Skala nicht identifizierbar, da die Fähigkeitsparameter nur bis auf affine Transformationen bestimmt sind (s. z.B. Baker und Kim (2004, S. 90)). Es gilt nämlich für $\tilde{\theta}_i := c + d\theta_i$ mit $d > 0$

$$\alpha_j(\theta_i - \beta_j) = \alpha_j \left(\frac{\tilde{\theta}_i - c}{d} - \beta_j \right) = \alpha_j \left(\frac{\tilde{\theta}_i - c - d\beta_j}{d} \right) = \frac{\alpha_j}{d} (\tilde{\theta}_i - c - d\beta_j) = \tilde{\alpha}_j (\tilde{\theta}_i - \tilde{\beta}_j)$$

mit $\tilde{\alpha}_j := \frac{\alpha_j}{d}$ und $\tilde{\beta}_j := c + d\beta_j$. Für die Verankerung sind verschiedene Verfahren möglich. Gebräuchlich ist die jeweilige Standardisierung der Fähigkeitsparameter, wodurch der Mittelwert der Fähigkeiten auf 0 und die Standardabweichung auf 1 gesetzt wird.

Die JMLE-Methode ist nicht ein konsistentes Schätzverfahren. In der Literatur wird die Tatsache gewöhnlich mit einer Argumentation von Neyman und Scott (1948) begründet. Neyman und Scott unterscheiden bei Schätzproblemen zwischen Parametern, die in der Verteilung aller Zufallsvariablen einer unendlichen Folge solcher auftauchen und Parametern, die nur in der Verteilung mancher, endlich vieler Zufallsvariablen auftauchen. Erstere nennen sie „strukturell“, letztere „inzidentell“. Während die Anzahl der strukturellen Parameter endlich ist, ist die der inzidentellen Parameter abzählbar unendlich. Beim Vorliegen inzidenteller Parameter ist die Konsistenz der

Schätzung struktureller Parameter nicht gewährleistet, wie sie mit einschlägigen Beispielen zeigen. Auf die IRT angewendet, werden die Itemparameter als die strukturellen betrachtet und die Fähigkeitsparameter als die inzidentellen. Für eine entsprechende Diskussion im Rahmen der IRT s. z. B. Baker und Kim (2004, S. 100) sowie Swaminathan und Gifford (1985, S. 349 f).

Ein weiteres Problem, das allerdings alle IRT-Schätzmethoden mit sich bringen, liegt darin, dass die Schätzung der Fähigkeitsparameter von Personen mit identischen Antwortvektoren identisch ist, auch wenn diese „faktisch“ unterschiedliche Fähigkeitsparameter aufweisen. Es können nicht mehr Fähigkeitsparameter geschätzt werden als es Typen von Antwortvektoren gibt und bei n dichotomen Items können maximal 2^n Fähigkeitsparameter geschätzt werden. Durch die Vergrößerung der Stichprobe bei konstanter Anzahl Items kann sich die Schätzung der Fähigkeitsparameter der einzelnen Person entsprechend nicht verbessern. Dies wirkt auf die Schätzung der Item-Parameter zurück. Dort spielt zwar die Häufigkeit der Antwortvektoren eines spezifischen Typs durchaus eine Rolle, auf Grund der relativ geringen Zahl von Antwortvektortypen und derer bei genügend grosser Stichprobe konstanter Zahl bleibt die Basis für die Schätzung doch recht klein und kann bei fixem n auch nicht vergrössert werden.

Im praktischen Rechengang ergeben sich bei der Methode zusätzliche Probleme. Die geschätzten Parameter tendieren bei der Iteration dazu, sich vom vernünftigen Bereich wegzuentwickeln, wie in der Literatur wiederholt erwähnt. Durch eigene Computer-Experimente hat sich gezeigt, dass mit dieser Methode Parameterschätzungen erzeugt werden, die sich von den zur Produktion der Daten verwendeten Parametern schnell entfernen und gewöhnlich nach wenigen Runden die Parameter nicht mehr schätzbar sind. Die in der Folge skizzierte MMLE-Schätzmethode führt einerseits bei der Voraussetzung stetig verteilter Zufallsvariablen, welche als Werte Fähigkeitsparameter annehmen, zu konsistenten Schätzern der Itemparameter und löst manchmal auch das Problem des Weggleitens von Schätzern im konkreten Rechengang.

4.4 Marginal Maximum Likelihood-Schätzmethode

Bei der *Marginal Maximum Likelihood-Schätzmethode (MMLE)* von Bock und Lieberman (1970) geht man - wie der Name sagt - von einer Randverteilung aus. Die Fähigkeitsparameter θ_i werden als Wert unabhängiger, stetig verteilter Zufallsvariable Θ_i betrachtet, deren Verteilung durch eine Dichte g gegeben ist. Die Hinzunahme von Personen in die Stichprobe hat unter diesen Voraussetzungen keinen Einfluss auf die Konsistenz der ML-Schätzung der Itemparameter.

Man betrachtet die Verteilung

$$P(\mathbf{x}_i|\theta_i, \boldsymbol{\delta}) := \mathbb{P}(\mathbf{X}_i = \mathbf{x}_i \mid \Theta_i = \theta_i, \boldsymbol{\Delta} = \boldsymbol{\delta})$$

des Zufallsvektors $\mathbf{X}_i = (X_{i1}, \dots, X_{in})^\top$ mit den Werten $\mathbf{x}_i = (x_{i1}, \dots, x_{in})^\top$, unter der Bedingung, dass die Fähigkeits-Zufallsvariable Θ_i den Wert θ_i und die Item-Parameter-Zufallsmatrix $\boldsymbol{\Delta}$ die Matrix $\boldsymbol{\delta} = (\delta_1, \dots, \delta_n)^\top$ annimmt. Bei gegebenem θ_i ist der Zufallsvektor $(X_{i1}, \dots, X_{in})^\top$ von der Zufallsvariable Θ_i unabhängig (lokale Unabhängigkeit, s. S. 6). Entsprechend ist in θ_i die gemeinsame Verteilung von $(X_{i1}, \dots, X_{in})^\top$ und Θ_i bestimmt durch das Produkt von g und der bedingten Dichtefunktion, die durch

$$P(\mathbf{x}_i|\theta_i, \boldsymbol{\delta})$$

gegeben ist. Damit kann man die Abhängigkeit von den Fähigkeitsparametern wie folgt wegintegrieren:

$$P(\mathbf{x}_i|\boldsymbol{\delta}) := \mathbb{P}(\mathbf{X}_i = \mathbf{x}_i|\boldsymbol{\Delta} = \boldsymbol{\delta}) = \int_{-\infty}^{+\infty} P(\mathbf{x}_i|\theta_i, \boldsymbol{\delta}) g(\theta_i) d\theta_i.$$

Mit

$$P(\mathbf{x}_i|\theta_i, \boldsymbol{\delta}) = \prod_{j=1}^n (p_j(\theta_i))^{x_{ij}} (1 - p_j(\theta_i))^{1-x_{ij}} g(\theta_i)$$

erhält man fürs logistische Modell mit

$$\boldsymbol{\delta} = (\delta_1, \dots, \delta_n)^\top,$$

$$\boldsymbol{\delta}_j := (\alpha_j, \beta_j)^\top$$

und

$$p_j(\theta_i) = \frac{\exp(\alpha_j(\theta_i - \beta_j))}{1 + \exp(\alpha_j(\theta_i - \beta_j))}$$

die folgende Likelihood-Funktion in den Parametervektoren $\boldsymbol{\alpha} := (\alpha_1, \dots, \alpha_n)^\top$ und $\boldsymbol{\beta} := (\beta_1, \dots, \beta_n)^\top$ - da über θ_i integriert wird, lassen wir künftig den Index in θ_i weg -

$$f_i(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \int_{-\infty}^{+\infty} P(\mathbf{x}_i|\theta, \boldsymbol{\delta}) g(\theta) d\theta.$$

Dies ergibt die gemeinsame LL-Funktion aller Personen

$$\ln L = \ln \prod_{i=1}^N f_i(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \sum_{i=1}^N \ln \int_{-\infty}^{+\infty} P(\mathbf{x}_i|\theta, \boldsymbol{\delta}) g(\theta) d\theta. \quad (6)$$

Es gilt nun wiederum, die Parameter so zu bestimmen, dass die LL-Funktion maximal wird.

Für das Modell (6) ergeben sich die folgenden Likelihood-Gleichungen - die mit 0 identifizierten Ableitungen von $\ln L$ nach $a_j := \alpha_j$ und $b_j := -\alpha_j \beta_j$:

Satz 5. Bei $a_j := \alpha_j$ und $b_j := -\alpha_j \beta_j$ ergibt sich für a_j die Likelihood-Gleichung

$$\frac{\partial \ln L}{\partial a_j} = \sum_{i=1}^N \int (x_{ij} - p_j(\theta)) P(\theta|\mathbf{x}_i, \boldsymbol{\delta}) \theta d\theta = 0 \quad (7)$$

und für b_j :

$$\frac{\partial \ln L}{\partial b_j} = \sum_{i=1}^N \int (x_{ij} - p_j(\theta)) P(\theta|\mathbf{x}_i, \boldsymbol{\delta}) d\theta = 0 \quad (8)$$

mit

$$P(\theta|\mathbf{x}_i, \boldsymbol{\delta}) = \frac{P(\mathbf{x}_i|\theta, \boldsymbol{\delta}) g(\theta)}{\int P(\mathbf{x}_i|\theta, \boldsymbol{\delta}) g(\theta) d\theta} \quad (9)$$

(Satz von Bayes) und

$$p_j(\theta) = \frac{\exp(a_j \theta + b_j)}{1 + \exp(a_j \theta + b_j)}.$$

Siehe für einen Beweis S. 52. Die MMLE wurde zuerst von Bock und Lieberman (1970) für ICCs vorgeschlagen, welche Verteilungsfunktionen von Normalverteilungen entsprechen. Für die Verteilung der Fähigkeitsparameter wird dort die Standardnormalverteilung gewählt. Für die Berechnung der Integrale wird von Bock und Lieberman (1970) die Gauss-Hermite-Quadratur mit q Knoten oder Stützpunkten Z_k und den entsprechenden Gewichten w_k verwendet (s. S. 46). Es ergeben sich aus (7) und (8) die Gleichungen

$$\begin{aligned} & \sum_{i=1}^N \sum_{k=1}^q (x_{ij} - p_j(Z_k)) P(Z_k|\mathbf{x}_i, \boldsymbol{\delta}) Z_k = \\ & \sum_{k=1}^q \left[\sum_{i=1}^N x_{ij} P(Z_k|\mathbf{x}_i, \boldsymbol{\delta}) - \sum_{i=1}^N p_j(Z_k) P(Z_k|\mathbf{x}_i, \boldsymbol{\delta}) \right] Z_k = 0 \end{aligned} \quad (10)$$

und

$$\sum_{i=1}^N \sum_{k=1}^q (x_{ij} - p_j(Z_k)) P(Z_k|\mathbf{x}_i, \boldsymbol{\delta}) = 0 \quad (11)$$

mit

$$P(Z_k|\mathbf{x}_i, \boldsymbol{\delta}) = \frac{\prod_{j=1}^n p_j(Z_k)^{x_{ij}} (1 - p_j(Z_k))^{1-x_{ij}} w_k}{\sum_{k=1}^q \prod_{j=1}^n p_j(Z_k)^{x_{ij}} (1 - p_j(Z_k))^{1-x_{ij}} w_k} \quad (12)$$

und

$$p_j(Z_k) = \frac{\exp(a_j Z_k + b_j)}{1 + \exp(a_j Z_k + b_j)},$$

wobei offensichtlich $\sum_{k=1}^q P(Z_k | \mathbf{x}_i, \boldsymbol{\delta}) = 1$ für jedes \mathbf{x}_i .

Für die Lösung der Gleichungen wurden von Bock und Lieberman (1970) das Newton-Raphson-Verfahren verwendet, was zu Berechnungs-Problemen bei einer grösseren Anzahl von Items führt. Deshalb wurde von Bock und Aitkin (1981) ein weiteres Berechnungsverfahren vorgeschlagen: eine Variante des EM-Algorithmus (Expectation-Maximisation, s. Dempster, Laird und Rubin (1977)). Statt dem Erwartungswert der Log-Likelihood der fehlenden Daten setzen sie die erwarteten Werte der fehlenden Daten in die Log-Likelihood ein (Bock & Aitkin, 1981, S. 448).

4.4.1 Die Bock-Aitkin-Schätzmethode

$P(Z_k | \mathbf{x}_i, \boldsymbol{\delta})$ in (12) ist die a posteriori Wahrscheinlichkeit, dass eine befragte Person mit dem Antwortvektor \mathbf{x}_i bei $\boldsymbol{\delta}$ die Fähigkeit Z_k aufweist. Für den E-Schritt im EM-Algorithmus kann man in (10) mit $q_j(Z_k) := 1 - p_j(Z_k)$ die folgenden Terme definieren:

$$\begin{aligned} f_k &:= \sum_{i=1}^N P(Z_k | \mathbf{x}_i, \boldsymbol{\delta}) \\ &= \sum_{i=1}^N \frac{\prod_{j=1}^n p_j(Z_k)^{x_{ij}} q_j(Z_k)^{1-x_{ij}} w_k}{\sum_{k=1}^q \prod_{j=1}^n p_j(Z_k)^{x_{ij}} q_j(Z_k)^{1-x_{ij}} w_k}. \end{aligned} \quad (13)$$

f_k ist die Summe der a posteriori Wahrscheinlichkeiten der N untersuchten Personen, die Fähigkeit Z_k aufzuweisen, also die erwartete Anzahl Personen mit der Fähigkeit Z_k . Es gilt:

$$\sum_{k=1}^q f_k = N.$$

Zudem kann man den Ausdruck

$$\begin{aligned} r_{kj} &:= \sum_{i=1}^N x_{ij} P(Z_k | \mathbf{x}_i, \boldsymbol{\delta}) \\ &= \sum_{i=1}^N \frac{\prod_{j=1}^n x_{ij} p_j(Z_k)^{x_{ij}} q_j(Z_k)^{1-x_{ij}} w_k}{\sum_{k=1}^q \prod_{j=1}^n p_j(Z_k)^{x_{ij}} q_j(Z_k)^{1-x_{ij}} w_k} \end{aligned} \quad (14)$$

definieren. Es handelt sich um die erwartete Anzahl korrekter Antworten von Personen mit der Fähigkeit Z_k auf das Item j . Es gilt

$$\sum_{k=1}^q r_{kj} = \sum_{i=1}^N x_{ij}.$$

$\sum_{k=1}^q r_{kj}$ ist also mit der Summe der korrekten Antworten pro Item j identisch und entsprechend gilt

$$\sum_{j=1}^n \sum_{k=1}^q r_{kj} = \sum_{j=1}^n \sum_{i=1}^N x_{ij},$$

d.h. $\sum_{j=1}^n \sum_{k=1}^q r_{kj}$ ist mit der Summe der korrekten Antworten in der Antwortematrix identisch.

Aus (10) und (11) wird mit den Definitionen von f_k und r_{kj}

$$\begin{aligned} \sum_{k=1}^q (r_{kj} - f_k p_j(Z_k)) Z_k &= 0 \\ \sum_{k=1}^q (r_{kj} - f_k p_j(Z_k)) &= 0. \end{aligned}$$

Dies entspricht den Likelihoodgleichungen der klassischen logistischen Regression, wobei r_{kj} und f_k nicht unbedingt ganze Zahlen sind (s. Seite 45 Gleichungen (21) und (22) oder Agresti (1990, S. 113)). Bezüglich der erwarteten Häufigkeiten r_{jk} und f_k kann man deshalb eine klassische, logistische Regression berechnen, um die Parameterschätzungen \hat{a}_j und \hat{b}_j zu erhalten. Dies entspricht dem Maximierungsschritt des EM-Algorithmus. Mit den neu geschätzten Parametern $\hat{\mathbf{a}}$ und $\hat{\mathbf{b}}$ wird wiederum (f_k, r_{jk}) berechnet, etc.

Hinweise zur Implementierung der Bock-Aitkin-Schätzmethode in R Aus rechenökonomischen Gründen verwendet man eine reduzierte Antwortematrix \mathbf{x} . Sie enthält jeden vorkommenden Antwortvektor nur einmal als Zeile. Statt einer (N, n) -Matrix erhält man eine (s, n) -Matrix mit $s \leq 2^n$. Sei zudem $\mathbf{Z} \in \mathbb{R}^{q \times 2}$, wobei q die Anzahl der Knoten ist, die in der Gauss-Hermite-Integration (s. Anhang S. 46) verwendet werden, die erste Spalte von \mathbf{Z} sei $\mathbf{1}_q \in \{1\}^q$, die zweite Spalte enthalte die Gauss-Hermite-Knoten (unter GH-Knoten und GH-Gewichten sind jeweils die an die Standardnormalverteilung angepassten Varianten gemeint, s. Anhang S. 46). Sei $\mathbf{B} \in \mathbb{R}^{n \times 2}$ die Matrix mit den vorläufigen Schwierigkeits- und Trennschärfe-Parametern in dieser Reihenfolge als Spalten, wobei n die Anzahl der Items ist.

$$\mathbf{L} := \mathbf{Z}\mathbf{B}^\top \in \mathbb{R}^{q \times n}$$

ist damit die Matrix der vorläufigen Werte von $b_j + a_j Z_i$ in $\frac{\exp(a_j Z_i + b_j)}{1 + \exp(a_j Z_i + b_j)}$ für alle GH-Knoten Z_i und für die vorläufigen Parameter der n Items j . Dann werden für die Komponenten der Matrix \mathbf{L} die Werte der logistischen Verteilungsfunktion berechnet (in R mit „plogis(\mathbf{L})“, d.h. es wird $\frac{\exp(a_j Z_i + b_j)}{1 + \exp(a_j Z_i + b_j)}$ an allen GH-Knoten Z_i für alle j Items berechnet. Gemäss R-Hilfe berechnet plogis($a_j Z_i + b_j$) den Wert der logistischen Verteilungsfunktion

$$\frac{1}{1 + \exp\left(-\frac{a_j Z_i + b_j}{1}\right)} = \frac{\exp\left(-\frac{0 - (a_j Z_i + b_j)}{1}\right)}{1 + \exp\left(-\frac{0 - (a_j Z_i + b_j)}{1}\right)} = \frac{\exp(a_j Z_i + b_j)}{1 + \exp(a_j Z_i + b_j)}.$$

Man erhält die Matrix

$$\mathbf{P} \in \mathbb{R}^{q \times n}.$$

Um die Berechenbarkeit zu gewährleisten, werden Werte, die durch das Programm mittels Runden mit 0 oder 1 gleichgesetzt werden, von 0 und 1 etwas weggeschoben, indem z. B. 1 durch $1 - \sqrt{x}$ sowie 0 durch \sqrt{x} ersetzt wird, mit der computerabhängigen kleinsten positiven Gleit-Komma-Zahl x , so dass $1 + x$ im Programm als von 1 verschieden behandelt wird. \mathbf{P} sei in der Folge entsprechend korrigiert. Es wird

$$\mathbf{P}^{xz} := \exp\left(\mathbf{x} \ln(\mathbf{P})^\top + (\mathbf{E} - \mathbf{x}) \ln(\mathbf{E} - \mathbf{P})^\top\right) \in \mathbb{R}^{s \times q}$$

mit $\ln \mathbf{P} := (\ln p_{kj})_{k,j}^{q,n}$ und $\exp(\mathbf{A}) := (\exp a_{kj})_{i,k}^{s,q}$ berechnet. Für eine einzelne Komponente von \mathbf{P}^{xz} gilt damit

$$p_{ik}^{xz} = \exp\left(\sum_{j=1}^n \left(\ln p_{kj}^{x_{ij}} + \ln(1 - p_{kj})^{1-x_{ij}}\right)\right) = \prod_{j=1}^n p_{kj}^{x_{ij}} (1 - p_{kj})^{1-x_{ij}}.$$

Sei weiterhin

$$\mathbf{w} \in \mathbb{R}^q$$

der Vektor der Gewichte, die in der Gauss-Hermite-Integration verwendet werden (GH-Gewichte). Damit wird

$$\mathbf{P}^{xzw} := (p_{ik}^{xz} w_i)_{i,k}^{s,q}$$

berechnet. Es handelt sich um die Matrix der $P(\mathbf{x}_i | Z_k, \boldsymbol{\delta}) w_k$ für die vorkommenden Antwortvektoren \mathbf{x}_i und die Gewichte der GH-Knoten Z_k . Mit dem Spaltensummenvektor

$$\mathbf{p}^{xzw\Sigma} \in \mathbb{R}^q$$

von \mathbf{P}^{xzw} erhält man

$$\mathbf{P}^{zx} := \left(\frac{p_{ik}^{xzw}}{p_i^{xzw\Sigma}} \right)_{i,k}^{s,q},$$

die Matrix der Wahrscheinlichkeiten, dass eine Person mit dem Antwortvektor \mathbf{x}_i bei den gegebenen Itemparametern die Fähigkeit Z_k aufweist. Dies entspricht (9) von Seite 14.

Sei

$$\mathbf{o} \in \mathbb{R}^s$$

der Vektor der Häufigkeiten der s verschiedenen Antwortvektoren. Man berechnet den Spaltensummenvektor $\mathbf{f} \in \mathbb{R}^q$ von

$$(p_{ik}^{zx} o_i)_{i,k}^{s,q},$$

was dem Vektor $(f_k)_k^q$ der erwarteten Anzahl Personen mit der Fähigkeit Z_k entspricht (s. (13), S. 15). Schliesslich wird die Matrix

$$\mathbf{R} := (\mathbf{P}^{zx})^\top (x_{ij} o_i)_{i,j}^{s,n} \in \mathbb{R}^{q \times n}$$

berechnet. Sie entspricht der Matrix mit den n Spalten $(r_{kj})_k^q$ (s. (14), S. 15), der Anzahl erwarteter korrekter Antworten pro Item und GH-Punkt. Anschliessend werden für die $(r_{kj})_k^q$ für $j \in \mathbb{N}_n^*$ logistische Regressionen berechnet, womit man die neuen Koeffizienten gewinnt (R-Befehl: `glm(cbind(R[,j], f - R[,j]) ~ Z[,2], quasibinomial=T)`), die man in der Matrix $\mathbf{B} \in \mathbb{R}^{n \times 2}$ abspeichert, womit ein neuer Durchlauf starten kann. Am Schluss werden die Ordinatenabschnitte der logistischen Regressionen mittels $-\mathbf{B}[,1]/\mathbf{B}[,2]$ in Schwierigkeitsparameter umgerechnet.

Diskussion der Ergebnisse Berechnet man die Koeffizienten für eine Datenmatrix, die gemäss Modell mit Trennschärfeparametern (1,1,1,1,1) und Schwierigkeitsparametern (-2,-1,0,1,2) sowie standardnormalverteilten Fähigkeitsparametern erzeugt wurde, ergibt sich ein stabiles Berechnungsverfahren. Die Schwierigkeits- und Trennschärfeparameter scheinen ohne Tendenz um die tatsächlichen Werte zu streuen. Der Startvektor hat keinen ersichtlichen Einfluss auf die Endwerte des Verfahrens. Die entsprechenden Ergebnisse können mit dem R-Paket `irt` mit Hilfe des Befehls

```
BockAitkin(X1,betas)
```

mit einer Matrix von Antwortvektoren `X1` und Startparametern `betas` erzeugt werden (s. Beispiel in der Hilfe zum Befehl im `irt`-Paket). Werden 200 Datensätze mit $n = 5$ und $N = 1500$ gemäss Modell produziert und die Parameter mit der Bock-Aitkin-Methode mit 15 GH-Punkten geschätzt - das Experiment kann mit den `irt`-Befehlen

```
a=MCEval(BA=T);evalMCEval(a)
```

reproduziert werden, so ergibt sich folgendes Bild (s. Tabelle 2):

Item	Schwierigkeit		Trennschärfe	
	Mittel	Standardabweichung	Mittel	Standardabweichung
1	-2.011604289	0.27632809	1.0179003	0.1721568
2	-1.01900734	0.12488228	0.9954518	0.1392822
3	-0.001264617	0.06042319	1.0021921	0.1417835
4	1.00900076	0.1355373	1.0140858	0.1584379
5	2.034866899	0.26676081	1.012706	0.1613977

Tabelle 2: Arithmetische Mittel und Standardabweichungen von 200 Sätzen von 5×2 Parametern - mit dem Bock-Aitkin-EM-Algorithmus geschätzt für künstlich erzeugte Datensätze mit je 1500 Antwortvektoren und 5 Items gemäss Trennschärfeparametern (1,1,1,1,1) und Schwierigkeitsparametern (-2,-1,0,1,2) sowie standardnormalverteilten Fähigkeitsparametern (15 GH-Punkte; 50 EM-Durchgänge)

Es fällt auf, dass die Standardabweichungen der Trennschärfen sich offenbar eher zufällig unterscheiden, während die Standardabweichungen der Items mit peripheren Schwierigkeitsparametern eine bedeutend grössere Standardabweichung als die mit zentralen Parametern aufweisen. Dies ist zu erwarten: Die Streuung der Trennschärfen der Items werden pro Item nur durch die wirklichen Trennschärfen bestimmt, während bezüglich der Schwierigkeitsparameter Items mit peripheren Schwierigkeitsparametern in einem Bereich liegen, wo die meisten ICCs einen sehr flachen Verlauf haben und dort damit praktisch keine Trennschärfe mehr aufweisen.

Für ein Modell mit den Schwierigkeitsparametern $(-2,-1,0,1,2)$ und den Trennschärfeparametern $(1,2,3,2,0.5)$ - also für ein Modell, dessen ICCs sich schneiden - erhält man für 200 Datensätze folgende Mittelwerte und Standardabweichungen (irt-Befehle:

`a=MCeval(BA=T,paramatrix=(cbind(c(-2,-1,0,1,2),c(1,2,3,2,0.5))))`;
`evalMCeval(a)`, s. Tabelle 3):

Item	Schwierigkeit		Trennschärfe	
	Mittel	Standardabweichung	Mittel	Standardabweichung
1	-1.998209378	0.18747383	1.0131402	0.11585414
2	-1.000622857	0.0625253	2.0471364	0.25183118
3	0.002017961	0.04110085	3.0737507	0.55500316
4	0.994670687	0.065512	2.0755173	0.23554797
5	2.023655588	0.34763239	0.5062993	0.08457772

Tabelle 3: Arithmetische Mittel und Standardabweichungen von 200 Sätzen von 5×2 Parametern - mit dem Bock-Aitkin-EM-Algorithmus geschätzt für künstlich erzeugte Datensätze mit je 1500 Antwortevektoren und 5 Items gemäss Trennschärfeparametern $(1,2,3,2,0.5)$ und Schwierigkeitsparametern $(-2,-1,0,1,2)$ sowie standardnormalverteilten Fähigkeitsparametern (15 GH-Punkte; 50 EM-Durchgänge)

Bezüglich Nähe zu den wirklichen Parametern ergeben sich Schätzwerte von ähnlicher Güte wie im Standardfall mit Trennschärfeparametern $(1,1,1,1,1)$. Es fällt auf, dass nun auch die Standardabweichungen der Trennschärfeschätzer variieren: je kleiner die Trennschärfe, die für die Erzeugung des Items verwendet wurde, desto kleiner ist die Standardabweichung der Trennschärfe-Schätzer. Zudem ergibt sich, dass eine kleinere Trennschärfe (d.h. grösserer Trennschärfeparameter), wie zu erwarten, zu einer grösseren Standardabweichung der Schwierigkeitsparameter führt. Der Effekt, dass die Items mit peripheren Schwierigkeitsparametern eine grössere Standardabweichung der Schwierigkeitsschätzungen aufweisen, wird damit durch die Grösse der Trennschärfen verstärkt oder abgeschwächt - für das Modell mit den Schwierigkeitsparametern $(-2,-1,0,1,2)$ und den Trennschärfeparametern $(1,2,0.5,2,3)$ ergaben sich z. B. die folgenden Standardabweichungen der Schwierigkeitsschätzer: $(0.24508308, 0.09139178, 0.11917698, 0.07829978, 0.15740853)$.

Gemäss Theorie ist zu erwarten, dass ML-Schätzer von grossen Stichproben näherungsweise normalverteilt sind (s. z. B. Pfanzagl (1994, S. 98 ff.)). Q-Q-Plots der Parameterschätzungen ergeben, dass dies für die Schwierigkeiten besser zutrifft als für die Trennschärfen und für Items mit zentralen Schwierigkeitsparametern etwas besser als bei peripheren Items. Anbei als Beispiel die Q-Q-Plots für die ersten und den dritten Schwierigkeits- und Trennschärfeparameter der 200 Datensätze, die für die Berechnung der Mittelwerte und Standardabweichungen der Tabelle 2 dienten (s. Abbildungen 3 bis 6, für Reproduktion analoger Graphiken s. R-irt-Befehle s. Anhang 6.10.1, S. 58).

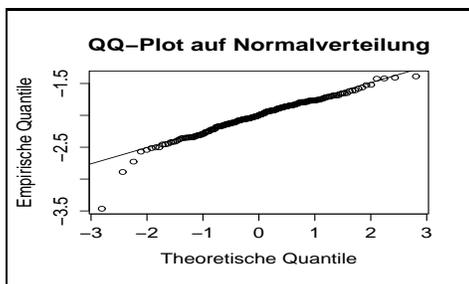


Abbildung 3: QQ-Plot auf Normalverteilung der geschätzten 200 Schwierigkeitsparameter für Item mit peripherer Schwierigkeit - erstes Item aus Tabelle 2

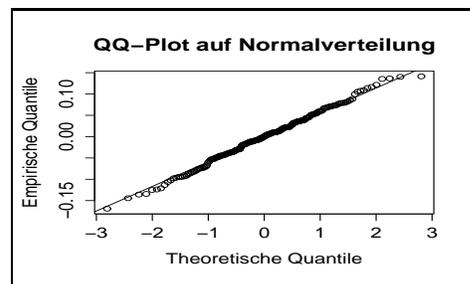


Abbildung 4: QQ-Plot auf Normalverteilung der geschätzten 200 Schwierigkeitsparameter für Item mit zentraler Schwierigkeit - drittes Item aus Tabelle 2

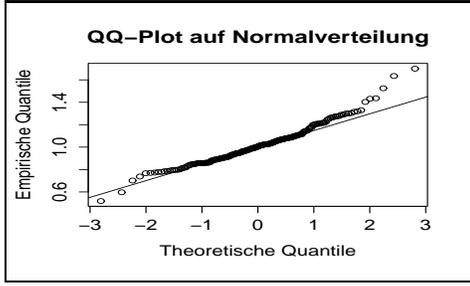


Abbildung 5: QQ-Plot auf Normalverteilung der geschätzten 200 Trennschärfeparameter für Item mit peripherer Schwierigkeit - erstes Item aus Tabelle 2

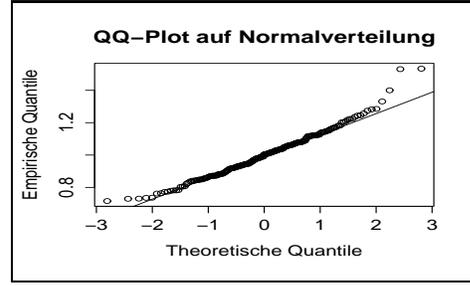


Abbildung 6: QQ-Plot auf Normalverteilung der geschätzten 200 Trennschärfeparameter für Item mit zentraler Schwierigkeit - drittes Item aus Tabelle 2

Es stellt sich die Frage, wie abhängig die Schätzwerte von der Voraussetzung der Normalverteilung der Fähigkeitsparameter sind. Zuerst werden die Schätzmethoden bezüglich uniform verteilten Fähigkeitsparametern untersucht. Es ergibt sich bei den Schwierigkeitsparametern $(-2, -1, 0, 1, 2)$, den Trennschärfeparametern $(1, 1, 1, 1, 1)$ und uniform verteilten Fähigkeitsparametern im Intervall $[-2.5, 2.5]$ folgendes Bild (s. Tabelle 4, Experiment zu reproduzieren mit: `a=MCeval(BA=T,unif=c(-2.5,2.5),distr=1);evalMCeval(a)`)

Item	Schwierigkeit		Trennschärfe	
	Mittel	Standardabweichung	Mittel	Standardabweichung
1	-1.352777245	0.11018738	1.46438	0.1716259
2	-0.630244978	0.05653031	1.599885	0.157509
3	-0.001005894	0.05136994	1.626606	0.1610123
4	0.63289311	0.05782176	1.587434	0.1703365
5	1.332069933	0.10149696	1.471994	0.1562675

Tabelle 4: Arithmetische Mittel und Standardabweichungen von 200 Sätzen von 5×2 Parametern - geschätzt für künstlich erzeugte Datensätze mit je 1500 Antwortvektoren und 5 Items gemäss Trennschärfeparametern $(1, 1, 1, 1, 1)$ und Schwierigkeitsparametern $(-2, -1, 0, 1, 2)$ sowie uniform verteilten Fähigkeitsparametern im Intervall $[-2.5, 2.5]$ (Bock-Aitkin-EM-Algorithmus; 15 GH-Punkte; 50 EM-Durchgänge)

Zu beachten ist, dass das Intervall $[-2.5, 2.5]$ die GH-Punkte nicht abdeckt. Bei 15 GH-Punkten sind die äussersten Punkte -6.363947889 und 6.363947889 . Allerdings wird die verwendete Schätzmethode bei einer Gleichverteilung auf z. B. $[7, 7]$ sehr instabil, so dass nie 100 Runden ohne Ausstieg der Routine zu Stande kamen. Der Abbruch ist jeweils im Maximierungsschritt erfolgt, und zwar bei der Berechnung der logistischen Regression. Er ist dem Umstand zuzuschreiben, dass der Vektor $(r_{kj})_k^q$ nahezu parallel zur Fähigkeitsachse liegt. In der Folge wird auf die Intervallbreite bei uniformverteilten Fähigkeiten zurückzukommen sein, sobald ein stabilerer EM-Algorithmus zur Verfügung steht. An den Ergebnissen der Tabelle 4 fällt auf, dass die Schwierigkeitsparameter die gleiche Ordnung aufweisen, wie die zur Erzeugung der Datensätze verwendeten, sie werden aber zusammengestaucht. Die Distanzgleichheit zwischen den Parametern bleibt erhalten. Die Trennschärfeparameter werden systematisch überschätzt, deren Gleichheit im Rahmen der Streuung der Schätzer scheint erhalten zu bleiben. Die Standardabweichungen der Trennschärfen verändern sich gegenüber dem Modell mit normalverteilten Fähigkeitsparametern nicht, die Standardabweichungen der Schwierigkeitsparameter werden an den Rändern kleiner. Die Phänomene verstärken sich, wenn grössere Intervalle für die uniforme Verteilung gewählt werden.

Verwendet man statt symmetrisch rechtsschief verteilte Fähigkeits-Parameter, z. B. indem man χ^2 -verteilte Daten erzeugt, die um den Erwartungswert (= Freiheitsgrade) der Verteilung

nach links verschoben werden, ergibt sich folgendes Bild (s. Tabelle 5, Experiment reproduzierbar mit:

`a=MCeval(BA=T,df=2,distr=2);evalMCeval(a):`

Item	Schwierigkeit		Trennschärfe	
	Mittel	Standardabweichung	Mittel	Standardabweichung
	Schwierigkeit	Standw	Trennschaerfe	Stabw
1	-1.3988203	0.11687922	1.172298	0.1204786
2	-0.4916467	0.0616277	1.401265	0.125505
3	0.1809378	0.04347706	1.743004	0.1745306
4	0.6732901	0.05410411	2.06391	0.2034218
5	1.0659112	0.05856634	2.457188	0.2538925

Tabelle 5: Arithmetische Mittel und Standardabweichungen von 200 Sätzen von 5×2 Parametern - geschätzt für künstlich erzeugte Datensätze mit je 1500 Antwortvektoren und 5 Items gemäss Trennschärfeparametern (1,1,1,1,1) und Schwierigkeitsparametern (-2,-1,0,1,2) sowie chiquadratverteilten Fähigkeitsparametern ($df = 2$, Fähigkeitsparameter anschliessend um 2 nach links verschoben) (Bock-Aitkin-EM-Algorithmus; 15 GH-Punkte; 50 EM-Durchgänge)

Im Beispiel werden 2 Freiheitsgrade gewählt, womit die Fähigkeitsparameter bei 15 GH-Punkten nur mit einer Wahrscheinlichkeit von 0.00172 rechts des obersten GH-Punktes zu liegen kommen. Zwar bleibt die Ordnung der Schwierigkeitsparameter gewahrt, diese werden aber zusammengestaucht, die Distanzen zwischen den Schwierigkeitsparametern sind nicht mehr - im Rahmen der Streuung - gleich und weisen eine Tendenz auf: die Differenzen zwischen den Parametern werden mit aufsteigenden Parametern kleiner. Im Beispiel ergeben sich die folgenden Differenzen: 0.9071736; 0.6725845; 0.4923523; 0.3926211. Die Trennschärfeparameter werden wiederum überschätzt und es scheint sich eine stärkere Überschätzung bei Items mit zentralen Schwierigkeitsparametern zu ergeben. Die Standardabweichungen der Trennschärfeparameter liegen im Bereich der Schätzer im Falle der normalverteilten Fähigkeitsparameter. Werden die Freiheitsgrade gross gewählt, ergibt sich eine stärkere Überschätzung der Trennschärfen. Zudem wird die Berechenbarkeit gefährdet. In diesem Fall kann man die künstlichen Daten standardisieren, damit sie im Bereich der GH-Punkte bleiben. Dadurch kann Berechenbarkeit für bedeutend grössere Freiheitsgrade gewährleistet werden. Die Parameterschätzungen werden allerdings volatil und die Berechenbarkeit ist nicht in jedem Fall gewährleistet. Die beschriebenen Tendenzen - aufsteigend stärkeres Zusammenstauchen der Schwierigkeitsparameter und mit steigenden Freiheitsgraden ein massiveres Überschätzen der Trennschärfen - ergibt sich auch in diesem Fall.

Insgesamt zeigt sich also, dass die Verteilungsannahme bezüglich der Fähigkeitsparameter eine wichtige Rolle spielt. Bock und Aitkin geben als Vorteil der MMLE an, dass man für die Item-Parameterschätzung nicht unbedingt eine Normalverteilung oder eine andere Verteilung der Fähigkeitsparameter willkürlich annehmen muss (Bock & Aitkin, 1981, S. 444). Man kann vielmehr von einer diskreten Verteilung auf endlich vielen Punkten ausgehen. Die Item-Parameter kann man schätzen, indem man mit der geschätzten Wahrscheinlichkeitsfunktion (s. Definition 6 auf Seite 47) arbeitet und damit die Methode von willkürlichen Voraussetzungen über die Verteilung der Fähigkeiten in der Population befreit. Die Wahrscheinlichkeitsfunktion wird durch das EM-Item-Parameter-Schätzverfahren in jeder Runde mitgeschätzt. Bevor entsprechende Verfahren diskutiert werden, noch ein Blick auf die EM-Schätzmethode des R-Paketes ltm, die sich als stabiler als die Bock-Aitkin-Schätzmethode erweist und die man mit den Bock-Aitkin-Vorschlägen bezüglich empirischer Verteilung kombinieren kann.

4.4.2 EM-Schätzmethode des R-Paketes ltm

Eine MMLE-Methode für logistische *ICC*'s mit normalverteilter Dichte g ist im R-Paket ltm implementiert (Rizopoulos, 2006). Es wird eine Alternative zur obigen EM-Methode verwendet.

Mit den Definitionen von Unterkapitel 4.4.1 wird für jedes Item j

$$\mathbf{Y}_j = (x_{ij} - p_{kj})_{ik}^{sq}$$

berechnet (R-Befehl: `outer(x[i, j], P[k, j], "-"`). Es handelt sich um die Differenzen der Werte der Antwortmatrix der Spalte j zu den Wahrscheinlichkeiten, dass im Knotenpunkt Z_k eine korrekte Antwort gegeben wird (s. Likelihoodgleichungen (7) und (8) von Seite 14). Anschliessend wird die Spaltensumme $\mathbf{s}_c^j \in \mathbb{R}^2$ von

$$\left(p_{ik}^{zx} \cdot o_i \cdot y_{ik}^j \right)_{ik}^{sq} \left(z_{kl}^{q2} \cdot w_k \right)_{kl}^{q2} \in \mathbb{R}^{s \times 2} \quad (15)$$

gebildet - die Berechnung des GH-Integrals der Likelihoodgleichungen (7) und (8) von Seite 14 in der GH-Integrationsvariante von (10) und (11) von Seite 11 (Erwartungsschritt). Es handelt sich um die Erwartungswerte der Itemparameter des Items j unter der Bedingung der Itemparameter δ_j des vorangegangenen Schrittes und der vorliegenden Antwortvektoren.

Schliesslich wird mit

$$\mathbf{V} := (p_{ij} (1 - p_{ij}))_{ij}^{sn}$$

die Hessematrix

$$\mathbf{H}_j = \mathbf{Z}^T (v_{ij} \cdot f_i \cdot z_{il})_{il}^{s2} \in \mathbb{R}^{2 \times 2}$$

berechnet, die Matrix der zweiten Ableitungen der Likelihoodfunktionen (s. Seite 45). Anschliessend wird das Gleichungssystem

$$\mathbf{H}_j \mathbf{u}_j = \mathbf{s}_c^j$$

aufgelöst (Maximierungsschritt) und die Lösung zu δ_j^{alt} hinzugezählt:

$$\delta_j^{neu} = \delta_j^{alt} + \mathbf{u}_j \in \mathbb{R}^2.$$

Im ltm-Pakte werden anschliessend an den EM-Algorithmus noch Iterationen mit Hilfe des BFGS-Algorithmus des R-Befehls `optim()` ausgeführt, der eine Näherung an die Hessematrix mitliefert. Deren Negation stimmt im vorliegenden Fall mit der Informationsmatrix überein. Die Inverse der Informationsmatrix liefert die näherungsweise Varianzen der Item-Parameter (s. Agresti (1990, S. 114)).

Diskussion der Ergebnisse Die Ergebnisse des von Rizopoulos und des von Bock-Aitkin verwendeten EM-Algorithmus sind vergleichbar. So erhält man für den gleichen, künstlich geschaffenen Datensatz ($N=1500$, $n=5$, Trennschärfeparameter (1,1,1,1,1), Schwierigkeitsparameter (-2,-1,0,1,2), standardnormalverteilte Fähigkeitsparameter) für das Item mit dem Schwierigkeitsparameter -2 und Trennschärfeparameter 1 Ergebnisse, die sich nach 50 Iterationen erst in der 6. Stelle nach dem Komma unterscheiden, z. B. (s. Tabelle 6, Experiment reproduzierbar mit:

```
X=SchaffDatDicho();X1=X[[1]][,2:6];
b=c(0,0,0,0,0);a=c(1,1,1,1,1);betas=cbind(b,a);
BockAitkin(X1,betas); RizopoulosEM(X1,betas)).
```

	Schwierigkeit	Trennschärfe
Bock Aitkin	-2.09795255	0.9863748
Rizopoulos	-2.09795369	0.986374

Tabelle 6: Schätzer der Item-Parameter des Items mit Parametern -2 (Schwierigkeit) und 1 (Trennschärfe) - nach Bock-Aitkin- respektive Rizopoulos-EM-Algorithmus für einen künstlich erzeugten Datensatz ($N=1500$, $n=5$, Trennschärfeparameter (1,1,1,1,1), Schwierigkeitsparameter (-2,-1,0,1,2), standardnormalverteilte Fähigkeitsparameter, 15 GH-Punkte; 50 EM-Durchgänge)

Vergleicht man für 200 künstlich erzeugte Datensätze die Schätzungen, ergibt sich im Beispiel eine maximale Differenz bei den Schwierigkeitsparametern von 0.0002810508 im 5. Item, dicht

gefolgt von der Differenz des 1. Items. Die maximalen Differenzen sind bei peripheren Items grösser als bei zentralen. Im dritten Item beträgt sie z. B. $4.954355e-06$. Ähnliche Resultate ergeben sich bezüglich der Trennschärfeparameter: bei peripheren Items ist die maximale Differenz grösser - im Beispiel liegt die maximale Differenz im 5. Item (0.0003508468), dicht gefolgt von der Differenz des 1. Items, beim dritten Item beträgt sie demgegenüber $6.670504e-06$. Vergleicht man die Mittel und die Standardabweichungen der 200 künstlich erzeugten Datensätzen, so ergibt sich folgendes Bild (s. Tabellen 7 und 8, Experiment reproduzierbar mit $a=MCeval(BA=T,RIZ=T)$; $evalMCeval(a)$):

Item	Schwierigkeit					
	Mittel der Schätzer			Standardabweichungen		
	Bock Aitkin	Rizopoulos	näher	Bock Aitkin	Rizopoulos	kleiner
1	-2.033203047	-2.033209058	RIZ	0.27936461	0.27936114	RIZ
2	-1.017021263	-1.017021089	RIZ	0.11975355	0.1197489	RIZ
3	-0.000482963	-0.000483061	RIZ	0.06662164	0.06662116	RIZ
4	1.006106045	1.006109562	BA	0.12914403	0.12914515	BA
5	1.999418816	1.999424101	RIZ	0.25514361	0.25513522	RIZ

Tabelle 7: Mittel der Schätzer der Schwierigkeitsparameter von 200 Datensätzen ($N=1500$, $n=5$, Trennschärfeparameter $(1,1,1,1,1)$, Schwierigkeitsparameter $(-2,-1,0,1,2)$, standardnormalverteilte Fähigkeitsparameter) berechnet nach Bock-Aitkin- respektive Rizopoulos-EM-Algorithmus - steht unter „näher“ „RIZ“, liegt das Mittel der Rizopoulos-Schätzer näher am wirklichen Parameter, sonst das Mittel der Bock-Aitkin-Schätzer, steht bei den Standardabweichungen der Schätzer unter „kleiner“ „RIZ“, weisen die Rizopoulos-Schätzer eine kleinere Standardabweichung auf als die Bock-Aitkin-Schätzer (15 GH-Punkte; 50 EM-Durchgänge).

Item	Trennschärfe					
	Mittel der Schätzer			Standardabweichungen		
	Bock Aitkin	Rizopoulos	näher	Bock Aitkin	Rizopoulos	kleiner
1	1.019413	1.0194074	RIZ	0.1721491	0.1721437	RIZ
2	0.99814	0.9981381	RIZ	0.1434385	0.1434305	RIZ
3	0.9944903	0.994504	RIZ	0.1521237	0.1521322	BA
4	1.0162828	1.0162772	RIZ	0.1459605	0.1459568	RIZ
5	1.0347179	1.0347116	RIZ	0.1659145	0.1659061	RIZ

Tabelle 8: Mittel der Schätzer der Trennschärfeparameter von 200 Datensätzen ($N=1500$, $n=5$, Trennschärfeparameter $(1,1,1,1,1)$, Schwierigkeitsparameter $(-2,-1,0,1,2)$, standardnormalverteilte Fähigkeitsparameter) berechnet nach Bock-Aitkin- respektive Rizopoulos-EM-Algorithmus - steht unter „näher“ „RIZ“, liegt das Mittel der Rizopoulos-Schätzer näher am wirklichen Parameter, sonst das Mittel der Bock-Aitkin-Schätzer, steht bei den Standardabweichungen der Schätzer unter „kleiner“ „RIZ“, weisen die Rizopoulos-Schätzer eine kleinere Standardabweichung auf als die Bock-Aitkin-Schätzer (15 GH-Punkte; 50 EM-Durchgänge).

Der Algorithmus von Rizopoulos schneidet also leicht besser ab, wobei Unterschiede in der 5. oder 6. Stelle nach dem Komma kaum als relevant betrachtet werden können. Bei Tests mit uniform oder verschoben chi-quadratverteilten Fähigkeitsparametern zeigt sich allerdings, dass der Algorithmus von Rizopoulos viel stabiler ist als der Bock-Aitkin-Algorithmus. Bei 200 Durchgängen führt der Bock-Aitkin-Algorithmus bei uniformer Verteilung auf $[-5,5]$ oder bei verschobener Chi-Quadrat-Verteilung mit 5 Freiheitsgraden regelmässig zum Abbruch, während der Algorithmus von Rizopoulos problemlos rechnet. Führen beide Algorithmen die Berechnungen zu Ende, bewegen sich die Unterschiede der Resultate bei nicht-normalverteilten Fähigkeitsparametern im Rahmen

der Schätzer bei normalverteilten Parametern und sind vernachlässigbar. Zuletzt spricht neben der grösseren Stabilität auch die grössere Schnelligkeit für den EM-Algorithmus von Rizopoulos. Auf Grund der Sensibilität beider Verfahren auf Verletzungen der Normalverteilungsbedingung bezüglich Fähigkeitsparameter wird es in der Folge darum gehen, das Schätzverfahren von dieser Voraussetzung unabhängig zu machen.

4.4.3 Schätzung mit empirischer Verteilung

Bock und Aitkin (1981) erwähnen die Möglichkeit, für die Dichte g statt der Normalverteilung eine aus den Daten geschätzte empirische Verteilung der Fähigkeiten zu verwenden. Sie schlagen für g folgende geschätzte Wahrscheinlichkeitsfunktion vor, wobei die Abstände zwischen den Z_k - wie dies übrigens auch bei der GH-Integration der Fall ist - konstant gewählt werden:

$$g(Z_k) := \frac{f_k}{N}$$

N ist die Anzahl der Antwortvektoren in der Antwortematrix \mathbf{x} . f_k ist die erwartete Häufigkeit von Personen, die Fähigkeit Z_k aufzuweisen - bedingt auf die vorkommenden Antwortvektoren \mathbf{x}_i und die vorläufigen Itemparameter $\boldsymbol{\delta}$ - siehe (13) auf Seite 15. Bei jedem Durchlauf l wird

$$w_k^{(l)} := \frac{f_k^{(l)}}{N}$$

neu berechnet und im nächsten Durchlauf als Gewicht verwendet. Zum Starten kann eine uniforme Wahrscheinlichkeitsfunktion auf die Stützpunkte verwendet werden. Man kann aber auch mit einer angenommenen Standardnormalverteilung, also mit GH-Stützpunkten und GH-Gewichten starten. Für die Implementierung genügt es, im obigen Rizopolous-Algorithmus allenfalls andere Punkte für Z_k - statt der GH-Punkte - zu wählen und nach dem ersten Durchgang w_k durch die hier definierten Gewichte zu ersetzen. Die Startverteilung bei den EM-Iterationen ist jeweils eine Wahrscheinlichkeitsfunktion oder relative Häufigkeitsverteilung (s. Definition 6 auf Seite 47), in der Folge werden wir aber auch von Start-Verteilungen sprechen. Gemeint sind die entsprechenden Wahrscheinlichkeitsfunktionen oder relativen Häufigkeitsverteilungen.

Diskussion der Ergebnisse Startet man für standardnormalverteilte Fähigkeiten mit GH-Punkten und GH-Gewichten, um diese in den folgenden Durchgängen durch die geschätzten Häufigkeiten an den Punkten Z_k zu ersetzen, ergibt sich folgendes Bild (s. Tabellen 9 und 10, Experiment reproduzierbar durch `a=MCeval(RIZ=T,ED=T);evalMCeval(a)`):

Item	Schwierigkeit					
	Mittel der Schätzer		näher	Standardabweichungen		
	Rizopoulos	Emp. Verteilung		Rizopoulos	Emp. Verteilung	kleiner
1	-2.028797637	-1.954940367	RIZ	0.27748209	0.26401819	EV
2	-1.024657949	-0.985994404	EV	0.1299304	0.11952838	EV
3	-0.000129804	0.0006259	RIZ	0.06960651	0.05939966	EV
4	1.018618591	0.984168575	EV	0.12909697	0.12098508	EV
5	2.04152342	1.978341093	EV	0.28404873	0.27624101	EV

Tabelle 9: Mittel der Schätzer der Schwierigkeitsparameter von 200 Datensätzen ($N=1500$, $n=5$, Trennschärfeparameter $(1,1,1,1,1)$, Schwierigkeitsparameter $(-2,-1,0,1,2)$, standardnormalverteilte Fähigkeitsparameter) berechnet mit dem Rizopoulos-EM-Algorithmus mit GH-Punkten und Gewichten für alle Durchgänge oder mit GH-Punkten und Gewichten im ersten Durchgang und nachher mit geschätzter empirischer Verteilung - steht unter „näher“ „RIZ“, liegt das Mittel der Rizopoulos-Schätzer näher am wirklichen Parameter, sonst das Mittel der Schätzer mit empirischer Verteilung; steht bei den Standardabweichungen der Schätzer unter „kleiner“ „RIZ“, weisen die Rizopoulos-Schätzer eine kleinere Standardabweichung auf als die Schätzer mit empirischer Verteilung (15 GH-Punkte, 50 EM-Durchgänge).

Die Ergebnisse sind vergleichbar, was wohl wenig erstaunlich ist, startet man den Schätzprozess doch mit GH-Punkten und GH-Gewichten, also mit einer Wahrscheinlichkeitsfunktion (W-Funktion), die der Verteilung der Fähigkeiten entspricht.

Item	Trennschärfe					
	Mittel der Schätzer		näher	Standardabweichungen		
	Rizopoulos	Emp. Verteilung		Rizopoulos	Emp. Verteilung	kleiner
1	1.0178996	1.057833	RIZ	0.1655063	0.1766747	RIZ
2	0.9977732	1.036576	RIZ	0.1527087	0.1573737	RIZ
3	1.0142174	1.053029	RIZ	0.1533096	0.1559645	RIZ
4	0.9976096	1.033777	RIZ	0.1454978	0.1507464	RIZ
5	1.0070135	1.042109	RIZ	0.1690239	0.1821049	RIZ

Tabelle 10: Mittel der Schätzer der Trennschärfeparameter von 200 Datensätzen ($N=1500$, $n=5$, Trennschärfeparameter $(1,1,1,1,1)$, Schwierigkeitsparameter $(-2,-1,0,1,2)$, standardnormalverteilte Fähigkeitsparameter) berechnet nach dem Rizopoulos-EM-Algorithmus mit GH-Punkten und Gewichten für alle Durchgänge oder mit GH-Punkten und Gewichten im ersten Durchgang und nachher mit geschätzter empirischer Verteilung - steht unter „näher“ „RIZ“, liegt das Mittel der Rizopoulos-Schätzer näher am wirklichen Parameter, sonst das Mittel der Schätzer mit empirischer Verteilung; steht bei den Standardabweichungen der Schätzer unter „kleiner“ „RIZ“, weisen die Rizopoulos-Schätzer eine kleinere Standardabweichung auf als die Schätzer mit empirischer Verteilung (15 GH-Punkte, 50 EM-Durchgänge).

Bei den Trennschärfen ergibt sich - bei normalverteilten Fähigkeitsparametern und Start-W-Funktion, die dieser Verteilung entspricht - mit der Methode der empirischen Verteilung eine schwache, aber systematische Überschätzung.

Startet man mit einer uniformen Verteilung bei normalverteilten Fähigkeiten, sind die Ergebnisse bedeutend schlechter - und zwar in Abhängigkeit von der Wahl des Intervalls, auf das die uniforme Verteilung definiert wird. Je grösser das Intervall, desto mehr werden zuerst die peripheren Schwierigkeitsparameter über- und unterschätzt und desto mehr werden die Trennschärfeparameter zuerst gleichmässig unterschätzt (s. Tabelle 11, S. 26). Zu recht guten Schätzungen gelangt man, so könnte man meinen, wenn man die Grenzen des Intervalls etwas kleiner setzt als die wirklichen, peripheren Schwierigkeitsparameter. Ein Blick auf die in der Folge analysierten geschätzten Wahrscheinlichkeitsfunktionen zeigt allerdings, dass die scheinbar gute Schätzung nicht auf einer korrekt geschätzten empirischen W-Funktion ruht (s. Abbildungen 7 bis 10). Die Schätzung ist

offenbar eher ein Artefakt der zu kleinen Grenzen. Interessant ist, dass bei kleinen Intervallen die Schätzwerte der peripheren Schwierigkeitsparameter ausserhalb der Intervalle liegen, bei grossen Intervallen innerhalb. Bemerkenswert ist auch, dass sich die Vorzeichen der Schwierigkeitsparameter bei $[-20, 20]$ plötzlich umkehren (s. Tabelle 11, R-Befehle s. Anhang 6.10.2, S. 58). Dies könnte allerdings an der Anzahl der Stützstellen liegen. In der Tat ergeben sich bei der Wahl von 50 statt 15 Stützstellen für kleine Intervalle analoge Ergebnisse, die Anomalie bei $[-20, 20]$ verschwindet aber. Interessanterweise kehrt sich die beobachtete Tendenz sogar plötzlich um: gegenüber dem Intervall $[-10, 10]$ werden die peripheren Schwierigkeitsparameter wieder etwas zusammengerückt - im Beispiel fürs erste Item von -4.8 auf -4.5 und im 5. Item von 4.7 auf 4.5 - und die Unterschätzung der Trennschärfeparameter wird bei allen Trennschärfen kleiner - im Mittel in einem Beispiel mit 200 Durchgängen um 0.028 . Dies könnte daran liegen, dass bei zu kleinen Intervallen an Randpunkten Wahrscheinlichkeit „aufgestaut wird“, die eigentlich auf entferntere Punkte fallen sollte, so dass eine zusätzliche Überschätzung der Schwierigkeitsparameter erfolgt. Die Anzahl der EM-Iterations-Runden spielt keine wesentliche Rolle. Die Parameter-Schätzungen können sich sogar verschlechtern. Entsprechende Experimente wurden mit 200 Datensätzen mit $N=1500$ und $n=5, 50$ Knotenpunkten, im Intervall $[-20, 20]$ sowie $[-5, 5]$ für 50, 100, 500 und 1000 EM-Iterations-Runden durchgeführt.

Vermutlich liegen die unbefriedigenden Schätz-Resultate (s. Tabelle 11) am Umstand, dass die Verteilung der Fähigkeitsparameter durch das Verfahren nicht optimal geschätzt wird. Die folgenden Graphiken (s. Abbildungen 7 bis 10, Seite 27, für die Reproduktion s. R-Befehle Anhang 6.10.3, S. 58) zeigen, dass die entsprechenden W-Funktionen schlecht geschätzt werden - unter diskret „normalverteilt“ wird dabei folgendes verstanden: um die Stützpunkte Z_k herum wird

$$\int_{Z_k-d}^{Z_k+d} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}\theta^2\right) d\theta$$

berechnet, für $d = (Z_k - Z_{k-1})/2$, und Z_k als Wahrscheinlichkeit zugeordnet. Beim ersten Punkt wird

$$\int_{-\infty}^{Z_1+d} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}\theta^2\right) d\theta$$

und beim letzten

$$\int_{Z_q-d}^{\infty} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}\theta^2\right) d\theta$$

berechnet (s. Befehl `normdiskret()` im `irt`-Paket).

Item	Schwierigkeit		Trennschärfe	
	Mittel	Standardabweichung	Mittel	Standardabweichung
Intervall [-1.8, 1.8]				
1	-2.067034929	0.26221599	1.0024777	0.1735156
2	-1.065117641	0.14171456	0.9576376	0.1405574
3	0.006259951	0.08557809	0.921493	0.1288627
4	1.084730377	0.13561976	0.9451916	0.1266166
5	2.111403029	0.25719997	0.9752859	0.1629611
Intervall [-2, 2]				
1	-2.29537661	0.28200253	0.9020967	0.1518609
2	-1.17594833	0.14240607	0.8673745	0.1183575
3	-0.00316091	0.09042783	0.8585045	0.1122556
4	1.16480184	0.14076751	0.8741462	0.1181298
5	2.28414164	0.28181332	0.9046458	0.148761
Intervall [-3, 3]				
1	-2.978491712	0.33957953	0.6923696	0.10075088
2	-1.526957606	0.16692341	0.6701015	0.09185679
3	-0.004977273	0.09512654	0.6568497	0.08886939
4	1.533051169	0.19323092	0.6570409	0.09297334
5	3.034835597	0.36735154	0.6709533	0.10119241
Intervall [-5, 5]				
1	-4.030323462	0.5370543	0.509596	0.08905157
2	-1.97853649	0.27037305	0.5222766	0.07958869
3	0.003446859	0.09968357	0.5284565	0.07645651
4	1.960957649	0.2607689	0.5228813	0.07912203
5	4.084466258	0.57869739	0.502646	0.08528746
Intervall [-10, 10]				
1	-4.7597512	0.8500493	0.4347262	0.09213288
2	-2.26583838	0.4181546	0.457188	0.08977121
3	0.003898782	0.1500531	0.4720057	0.08148323
4	2.200792309	0.3802225	0.4678541	0.08825477
5	4.767312031	0.8168498	0.4294014	0.08265747
Intervall [-20, 20]				
1	3.951562649	0.551353	-0.5167418	0.0873033
2	1.815170004	0.3319617	-0.5695995	0.10037352
3	0.004536404	0.1862047	-0.5719203	0.10238377
4	-1.846171454	0.3112669	-0.5488973	0.0882208
5	-3.920144168	0.5264242	-0.5238726	0.09512325

Tabelle 11: Mittel der Schätzer der Schwierigkeits- und der Trennschärfeparameter von je 200 Datensätzen ($N=1500$, $n=5$, Trennschärfeparameter (1,1,1,1,1), Schwierigkeitsparameter (-2,-1,0,1,2), standardnormalverteilte Fähigkeitsparameter) berechnet mit der Methode der empirischen Verteilung mit uniformer Start-Verteilung und 15 Stützpunkten auf variable Intervalle (50 EM-Durchgänge).

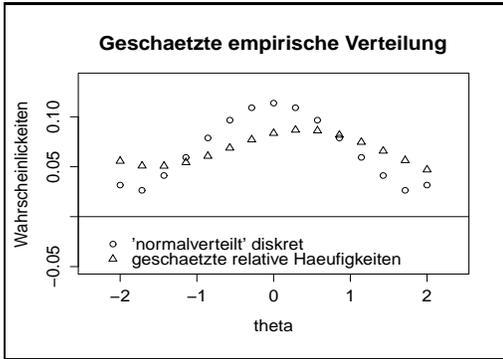


Abbildung 7: EM-Schätzung mit empirischer Verteilung: uniforme Startverteilung auf das Intervall $[-2,2]$, 15 Knoten, 50 EM-Iterationen, $N=1500$, $n = 5$, Trennschärfeparameter $(1,1,1,1,1)$, Schwierigkeitsparameter $(-2,-1,0,1,2)$, standardnormalverteilte Fähigkeiten

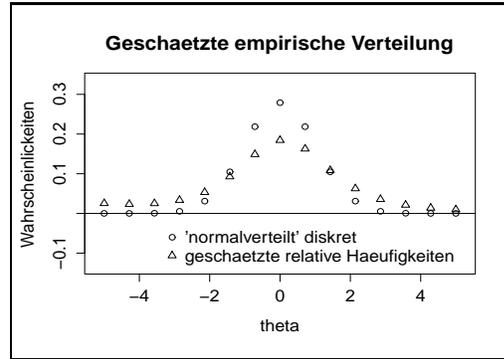


Abbildung 8: EM-Schätzung mit empirischer Verteilung: uniforme Startverteilung auf das Intervall $[-5,5]$, 15 Knoten, 50 EM-Iterationen, $N=1500$, $n = 5$, Trennschärfeparameter $(1,1,1,1,1)$, Schwierigkeitsparameter $(-2,-1,0,1,2)$, standardnormalverteilte Fähigkeiten

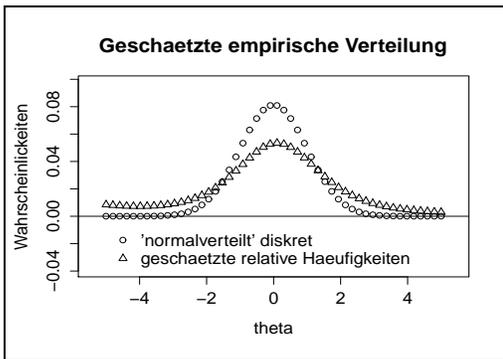


Abbildung 9: EM-Schätzung mit empirischer Verteilung: uniforme Startverteilung auf das Intervall $[-5,5]$, 50 Knoten, 50 EM-Iterationen, $N=1500$, $n = 5$, Trennschärfeparameter $(1,1,1,1,1)$, Schwierigkeitsparameter $(-2,-1,0,1,2)$, standardnormalverteilte Fähigkeiten

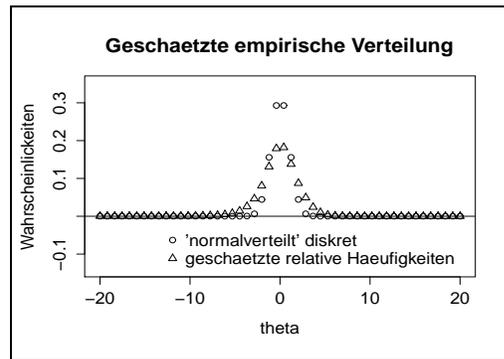


Abbildung 10: EM-Schätzung mit empirischer Verteilung: uniforme Startverteilung auf das Intervall $[-20,20]$, 50 Knoten, 100 EM-Iterationen, $N=1500$, $n = 5$, Trennschärfeparameter $(1,1,1,1,1)$, Schwierigkeitsparameter $(-2,-1,0,1,2)$, standardnormalverteilte Fähigkeiten

Die W-Funktions-Schätzungen sind also zu flach, aber immerhin fast symmetrisch, allerdings mit einer leichten Verschiebung der zentralen Lage nach rechts. Die Verteilungen werden auch mit mehr Knotenpunkten nicht besser angepasst. Die Schätzung der Verteilung bei einer uniformen Ausgangsverteilung auf $[-2, 2]$ ist schlechter als auf $[-5, 5]$, obwohl - wie bereits bemerkt - die Item-Parameter-Schätzer im ersten Falle besser sind. Dies liegt wohl daran, dass bei $[-2, 2]$ die Schätzungen der Schwierigkeits-Item-Parameter künstlich zentriert werden, was sich dann auch positiv auf die Trennschärfeparameter-Schätzungen auswirkt. Weiterhin ist bemerkenswert, dass bei $[-2, 2]$ eine unsymmetrische Verteilung geschätzt wird, obwohl die ganze Ausgangslage und das Schätzverfahren völlig symmetrisch sind.

Bei chi-quadrat-verteilten Fähigkeitsparametern und uniformer Startverteilung ergibt sich folgendes Bild (s. Tabelle 12, das Experiment kann reproduziert werden mit `al=MCEval(distr=2,df=5,ED=T,Intervall=c(-10,10)); evalMCEval2(al)`)

Item	Schwierigkeit		Trennschärfe	
	Mittel	Standardabweichung	Mittel	Standardabweichung
	Schwierigkeit	Standw	Trennschaerfe	Stabw
1	-2.3073678	0.1893656	0.6078272	0.07637274
2	-0.6713049	0.2560295	0.6277328	0.0683033
3	0.9595225	0.2583855	0.62209	0.0580778
4	2.6014659	0.3245385	0.6181223	0.06478461
5	4.2373638	0.2498054	0.6217869	0.05647394

Tabelle 12: Arithmetische Mittel und Standardabweichungen von 200 Sätzen von 5×2 Parametern - geschätzt für künstlich erzeugte Datensätze mit je 1500 Antwortvektoren und 5 Items gemäss Trennschärfeparametern (1,1,1,1,1) und Schwierigkeitsparametern (-2,-1,0,1,2) sowie chi-quadrat verteilten Fähigkeitsparametern mit $df=5$, geschätzt mit empirischer Verteilung, Startverteilung diskrete uniforme Verteilung auf [-10,10], 15 GH-Punkte, 50 EM-Durchgänge

Ausser einem werden alle Schwierigkeitsparameter überschätzt und zwar asymmetrisch: inklusive des zentralen Parameters werden die Parameter rechts - also bei den Stützpunkten, denen die W-Funktion kleinere Wahrscheinlichkeiten zuordnet - stark überschätzt. Die Trennschärfeparameter werden gleichmässig unterschätzt. Das Resultat kann wie bei der uniformen Verteilung kaum als befriedigend betrachtet werden. Gründe für die Ergebnisse können in der ungenügenden Schätzung der empirischen W-Funktion durch den Prozess gefunden werden (s. Abbildung 11, S. 30).

Bei der Standardnormalverteilung hat sich gezeigt, dass bei Übereinstimmung von Startverteilung und wirklicher Verteilung, sich gute Schätzungen ergeben. Es stellt sich die Frage, ob dies auch bei anderen Verteilungen gilt. Zuerst wird die uniforme Verteilung untersucht (s. Tabelle 13, Experiment für die Reproduktion analoger Daten:

`a=MCeval(ED=T,distr=1,unif=c(-3,3),Intervall=c(-3,3)); evalMCeval(a)`

Item	Schwierigkeit		Trennschärfe	
	Mittel	Standardabweichung	Mittel	Standardabweichung
	Schwierigkeit	Standw	Trennschaerfe	Stabw
1	-2.086149543	0.11069331	0.9679762	0.09302488
2	-1.035517125	0.09040135	0.9815945	0.08618318
3	0.001070793	0.08781782	0.9744003	0.08239422
4	1.045613635	0.10679995	0.9732819	0.08236811
5	2.078820508	0.13016257	0.9795805	0.10867498

Tabelle 13: Arithmetische Mittel und Standardabweichungen von 200 Sätzen von 5×2 Parametern - geschätzt für künstlich erzeugte Datensätze mit je 1500 Antwortvektoren und 5 Items gemäss Trennschärfeparametern (1,1,1,1,1) und Schwierigkeitsparametern (-2,-1,0,1,2) sowie uniform verteilten Fähigkeitsparametern auf [-3,3], geschätzt mit empirischer Verteilung, Startverteilung uniforme Verteilung auf [-3,3], 15 GH-Punkte, 50 EM-Durchgänge

Die Schätzungen können als gut betrachtet werden, wobei bei den peripheren Schwierigkeitsparametern eine ganz leichte Überschätzung und bei den Trennschärfeparametern eine leichte Unterschätzung festzustellen ist. Bei 50 Knotenpunkten kehrt sich der Effekt um: es ergibt sich eine leichte Unterschätzung der peripheren Schwierigkeitsparameter und eine ganz leichte Überschätzung der Trennschärfeparameter.

Bei der Chi-Quadrat-Verteilung sieht es diesbezüglich weniger gut aus. Um die geschätzte diskrete Wahrscheinlichkeitsfunktion mit der theoretischen, stetigen Dichtefunktion zu vergleichen,

wird letztere wieder diskretisiert - in der Folge „diskret 'chi-quadrat'-verteilt“ genannt - mittels

$$\int_{Z_k-d}^{Z_k+d} \frac{1}{2^{\frac{n}{2}} \cdot \Gamma\left(\frac{n}{2}\right)} \cdot x^{\frac{n}{2}-1} \cdot e^{-\frac{x}{2}} dx,$$

für $d = (Z_k - Z_{k-1}) / 2$, der Gamma-Funktion Γ und n Freiheitsgraden. Diese Wahrscheinlichkeiten werden den Stützpunkten Z_k zugeordnet. Die Stützpunkte werden regelmässig auf das Intervall der Länge

$$df + [0, Z_q]$$

verteilt, so dass $P([0, Z_q]) = 0.99$. Der erste Punkte wird auf die Hälfte des Abstandes der anderen Punkte von 0 weggesezt. Beim ersten Punkt wird entsprechend

$$\int_0^d \frac{1}{2^{\frac{n}{2}} \cdot \Gamma\left(\frac{n}{2}\right)} \cdot x^{\frac{n}{2}-1} \cdot e^{-\frac{x}{2}} dx$$

und beim letzten

$$\int_{Z_q-d}^{Z_q} \frac{1}{2^{\frac{n}{2}} \cdot \Gamma\left(\frac{n}{2}\right)} \cdot x^{\frac{n}{2}-1} \cdot e^{-\frac{x}{2}} dx$$

berechnet (s. Befehl `gewChiVert()`). Es wird nicht die gesamte Wahrscheinlichkeit rechts des letzten Punktes dem letzten Punkt zugewiesen - wie bei der diskreten 'Normalverteilung' (s. Befehl `normdiskret()`) -, da die Chi-quadrat-Verteilung für $x \rightarrow \infty$ langsamer gegen Null konvergiert als die Standardnormalverteilung. Die Schätzungen werden dadurch wohl auch wegen der Asymmetrie verzerrt. Es wurde auch der Versuch unternommen, die restliche Wahrscheinlichkeit auf alle Stützpunkte gleichmässig zu verteilen. Auch dies wirkte sich ungünstig aus. Dies lässt den Schluss zu, dass die Schätzverfahren recht sensibel auf die Startverteilung reagiert. Für chi-quadrat-verteilte Fähigkeiten mit $df=5$ und einer diskret 'chi-quadrat'-verteilten Start-W-Funktion ergab sich folgendes (s. Tabelle 14, Experiment für die Reproduktion analoger Daten: `al=MCeval(distr=2,df=5,ED=T,dfchi=5)`; `evalMCeval2(al)`):

Item	Schwierigkeit		Trennschärfe	
	Mittel	Standardabweichung	Mittel	Standardabweichung
	Schwierigkeit	Standw	Trennschaerfe	Stabw
1	-1.714845	0.07886837	1.06975	0.10257746
2	-0.762562	0.07503133	1.065154	0.08842889
3	0.204965	0.07227192	1.064604	0.09578349
4	1.150109	0.09708201	1.059565	0.08980732
5	2.105801	0.14435348	1.061243	0.11667014

Tabelle 14: Arithmetische Mittel und Standardabweichungen von 200 Sätzen von 5×2 Parametern - geschätzt für künstlich erzeugte Datensätze mit je 1500 Antwortvektoren und 5 Items gemäss Trennschärfeparametern (1,1,1,1,1) und Schwierigkeitsparametern (-2,-1,0,1,2) sowie chi-quadrat verteilten Fähigkeitsparametern mit $df=5$, geschätzt mit empirischer Verteilung, Startverteilung diskrete 'chi-quadrat' Verteilung mit $df=5$, 15 GH-Punkte, 50 EM-Durchgänge

Die Ergebnisse sind weniger befriedigend als im Falle symmetrischer Verteilungen. Es ergibt sich eine Unterschätzung des Schwierigkeitsparameters der links-peripheren Items und eine - wenn auch nur schwache - Überschätzung der rechtsperipheren, inklusive des zentralen Items. Ein Blick auf die geschätzten W-Funktionen zeigt Gründe für die Tendenz auf. (s. Abbildung 11, S. 30, für die Reproduktion mit R s. 6.10.4, S. 59):

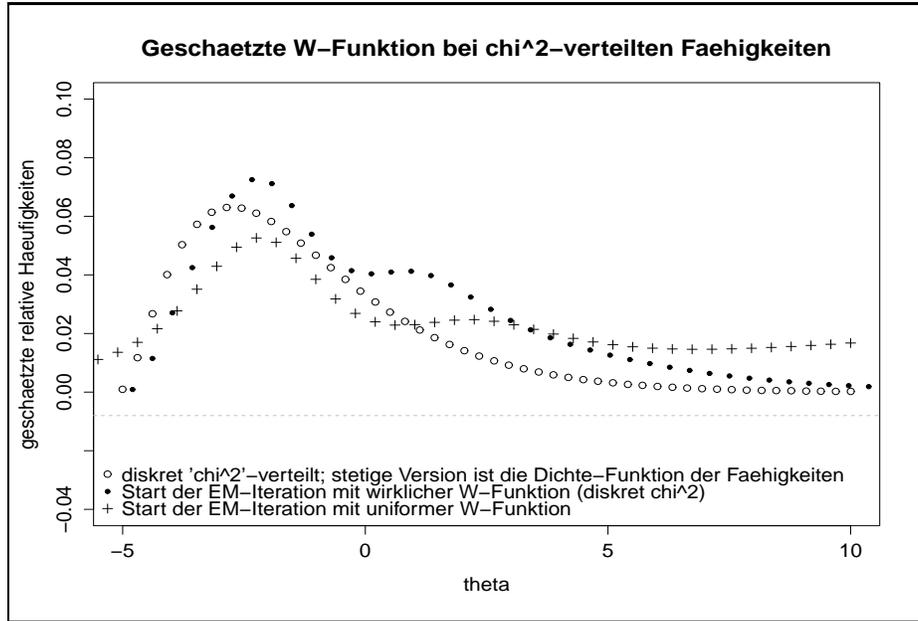


Abbildung 11: Diskretisierte Dichtefunktion der Verteilung der Fähigkeiten (Chi-Quadrat-Verteilung, $df=5$), Geschätzte W-Funktion (50 Knotenpunkte, 50 EM-Runden, Start-W-Funktion erwähnte diskretisierte Dichtefunktion), Geschätzte W-Funktion (Startverteilung uniforme W-Funktion)

Aus den durchgeführten Experimenten kann man schliessen, dass die Schätzverfahren mit empirischer Verteilung gut sind, wenn die Verteilung der Fähigkeitsparameter symmetrisch ist und man mit der wirklichen Verteilung der Fähigkeitsparameter startet. Sonst sind sie nicht überwältigend, aber bei symmetrischen besser als bei schiefen Verteilungen. Da man in der Anwendung die wirkliche Verteilung der Fähigkeiten nicht kennt, ist diese Feststellung an sich nicht hilfreich - ausser man könnte sich den Umstand zunutze machen, dass die EM-Iteration in der Nähe der wirklichen Verteilung kleinere Schritte macht ist als weiter weg. Eventuell könnte man auf diesem Hintergrund ein Näherungsverfahren für eine Start-W-Funktion entwickeln, die genügend nahe bei der wirklichen W-Funktion liegt. Mit asymmetrischen Verteilungen könnte damit aber immer noch nicht befriedigend umgegangen werden.

4.4.4 EM-Schätzverfahren mit Werten der suffizienten Statistiken

Versuchshalber wird noch ein letztes EM-Schätzverfahren behandelt. Als Stützpunkte Z_k werden die Werte der suffizienten Statistiken

$$Z_k^{(l)} := \sum_{j=1}^n \alpha_j^{(l)} x_j$$

gewählt - l für l -te EM-Iteration. Deren Anzahl wie deren Lage ist im Iterationsverfahren variabel. Die maximale Anzahl von Stützpunkten ist 2^n , n = Anzahl Items. Man geht von frei gewählten Itemparameter-Vektoren aus, z. B. $\alpha_n := \mathbf{1}_n \in \{1\}^n$ und $\beta_n := \mathbf{0}_n$. Die Wahl von α_n wirkt sich auf die Anzahl der beim Start gebildeten Stützpunkte aus. Bei $\alpha_n = \mathbf{1}_n$ ergeben sich $n+1$ Stützpunkte. Sei $q^{(l)}$ die Anzahl der Knoten, die dadurch im l -ten Durchgang entstehen. Man standardisiert die Werte $(Z_1^{(l)}, \dots, Z_q^{(l)})$. Die standardisierten Werte seien ebenfalls mit $(Z_1^{(l)}, \dots, Z_q^{(l)})$ bezeichnet. Man berechnet die relative Häufigkeitsverteilung $h(Z_k^{(l)})$ der Vorkommen von $Z_k^{(l)}$, $k \in \mathbb{N}_{q^{(l)}}^*$, und verwendet diese als Gewichte. Es wird der EM-Algorithmus von Rizopoulos verwendet. Im EM-Algorithmus von Bock-Aitkin ist das Verfahren bereits bei Antwortvektoren, die mit einem anderen Trennschärfeparametern als $(1, 1, 1, 1, 1)$ produziert wurden, instabil. Es ergeben sich folgende Resultate (s. Tabelle 15, für R-Befehle s. Anhang 6.10.5, S. 59).

Item	Schwierigkeit		Trennschärfe	
	Mittel	Standardabweichung	Mittel	Standardabweichung
Standardnormalverteilte Fähigkeitsparameter				
1	-2.044787392	0.2307635	1.0002112	0.1459735
2	-1.023394017	0.1246467	0.9881309	0.1432757
3	0.007392799	0.0646132	0.9854416	0.1324037
4	1.047567516	0.1205416	0.9724452	0.1333208
5	2.046315921	0.2623046	0.9995704	0.1476181
Standardnormalverteilte Fähigkeitsparameter (Trennschärfe Items: c(1,2,3,2,0.5))				
1	-2.061193631	0.20934228	0.9812678	0.12289754
2	-1.086196323	0.07225471	1.8829564	0.19943363
3	0.009664174	0.04011985	2.4483154	0.22581255
4	1.086532747	0.06069177	1.8954879	0.20539249
5	2.086113772	0.35109835	0.4925579	0.07813412
Uniform verteilte Fähigkeitsparameter auf [-10,10]				
1	-0.508147051	0.05120137	4.322017	0.3549863
2	-0.266178691	0.04878018	4.018384	0.2840158
3	-0.000454336	0.05423984	3.852637	0.2443863
4	0.269698955	0.05368339	4.041843	0.2924337
5	0.511493195	0.05172864	4.331874	0.4172545
Uniform verteilte Fähigkeitsparameter auf [-4,4]				
1	-0.887225687	0.04508048	2.313904	0.1743618
2	-0.448692704	0.04447247	2.270566	0.1491032
3	-0.001900981	0.04308594	2.255144	0.1432397
4	0.439872543	0.04311827	2.257401	0.1404939
5	0.883156633	0.04697371	2.300434	0.1990476
Chi-quadrat verteilte Fähigkeitsparameter mit $df = 5$				
1	-0.729244	0.0382486	2.444672	0.1765853
2	-0.3162562	0.03741425	2.386076	0.1628906
3	0.1022612	0.04538931	2.403589	0.1424536
4	0.5105192	0.0454749	2.509914	0.1462682
5	0.8773542	0.05047979	2.688886	0.2012712
Chi-quadrat verteilte Fähigkeitsparameter mit $df = 2$				
1	-1.3154723	0.0820177	1.353886	0.1301304
2	-0.5508059	0.04654772	1.458406	0.1247415
3	0.1094175	0.04899057	1.596975	0.1322036
4	0.6840449	0.06087919	1.767913	0.1371611
5	1.186326	0.0695315	1.932736	0.161838

Tabelle 15: Arithmetische Mittel und Standardabweichungen von je 200 Sätzen von 5×2 Parametern - geschätzt für künstlich erzeugte Datensätze mit je 1500 Antwortvektoren und 5 Items gemäss Trennschärfeparametern (1,1,1,1,1) und Schwierigkeitsparametern (-2,-1,0,1,2) (bei der zweiten Untertabelle mit den angegebenen Trennschärfen) für normalverteilte, uniform und chi-quadrat-verteilte Fähigkeiten, Knotenpunkte: standardisierte Summen der mit den Trennschärfen gewichteten Werte der Antwortvektoren, Gewichte pro Knotenpunkt: relative Häufigkeit deren Auftretens, 50 EM-Durchgänge.

Bei normalverteilten Fähigkeiten ist die Methode überraschend gut. Sie nimmt es mit den BA-Verfahren und RIZ-Verfahren in diesem Falle auf, obwohl keine spezifische Verteilung der Fähigkeiten vorausgesetzt wird (s. Tabelle 16).

Item	Schwierigkeiten			Trennschärfen		
	Suffiziente Statistik	Rizopoulos	näher	Suffiziente Statistik	Rizopoulos	näher
1	-2.02708374	-2.01260989	RIZ	1.0095264	1.0202628	SUF
2	-1.01663792	-0.997731071	RIZ	1.0002718	1.0221431	SUF
3	0.00957623	0.008153559	RIZ	0.9920593	1.0174904	SUF
4	1.0419095	1.026241768	RIZ	0.9727545	0.9885159	RIZ
5	2.05794058	2.059427892	SUF	0.9905845	0.9900195	SUF

Tabelle 16: Mittel der Schätzer der Schwierigkeitsparameter und der Trennschärfeparameter von 200 Datensätzen ($N=1500$, $n=5$, Trennschärfeparameter $(1,1,1,1,1)$, Schwierigkeitsparameter $(-2,-1,0,1,2)$, standardnormalverteilte Fähigkeitsparameter) berechnet nach der Methode der suffizienten Statistiken und der Rizopoulos-Methode - steht unter „näher“, „RIZ“, liegt das Mittel der Rizopoulos-Schätzer näher am wirklichen Parameter, sonst das Mittel der Schätzer mit suffizienter Statistik (15 GH-Punkte bei GH-Verfahren; 50 EM-Durchgänge).

Bei uniformer Verteilung ergeben sich systematische Verzerrungen. Es fällt auf, dass man die Schätzer durch Multiplikation mit einer Konstante und dem Kehrwert derselben Konstante zu guten Schätzern transformieren kann. Im Falle eines kleinen Intervalls, das durch die Original-Itemparameter gut abgedeckt wird, kann man als Konstante die Standardabweichung der uniformen Verteilung der verwenden (s. Tabelle 17).

Item	Schwierigkeit	Trennschaerfe	Zweite Spalte $\cdot s$	Dritte Spalte $\cdot \frac{1}{s}$
1	-0.887225687	2.313904	-2.048604111	1.002123863
2	-0.448692704	2.270566	-1.036031454	0.983354699
3	-0.001900981	2.255144	-0.004389365	0.976675617
4	0.439872543	2.257401	1.015665702	0.977653097
5	0.883156633	2.300434	2.039208666	0.996290169

Tabelle 17: Daten der vierten Untertabelle der Tabelle 15 - uniform verteilte Fähigkeiten auf $[-4, 4]$, Schwierigkeitsparameter $(-2, -1, 0, 1, 2)$ sowie Trennschärfeparameter $(1, 1, 1, 1, 1)$, Schätzungen mit Methode Suffiziente Statistik, transformiert mit der Standardabweichung $s = \sqrt{\frac{8^2}{12}} = 2.3094$.

Dieses interessante, aber empirische Ergebnis müsste noch theoretisch untermauert werden, wobei man in der Praxis die Standardabweichung nicht kennt und die Reichweite der Erkenntnis deshalb für die praktische Anwendung begrenzt ist.

Bei der Anwendung des Verfahrens mit suffizienten Statistiken bei chi-quadrat-verteilten Fähigkeiten zeigt sich erneut, dass bei schiefen Verteilungen die Güte der Schätzungen schlechter ist (s. letzte und vorletzte Untertabelle von Tabelle 15, S. 31). Ausser dem zentralen Schwierigkeitsparameter werden diese unterschätzt, auf der rechten Seite etwas weniger als auf der linken Seite. Die Trennschärfen weisen eine aufsteigende Tendenz auf: je weiter rechts der Schwierigkeitsparameter des Items steht, desto grösser wird die Trennschärfe geschätzt. Als befriedigend ist die Schätzung nicht zu betrachten. Im Vergleich zur Schätzung mit der empirischen Verteilung und uniformer Start-Verteilung (s. Tabelle 12, S. 28) zeigt sich, dass mit den suffizienten Statistiken im Gegensatz zur empirischen Methode die Schwierigkeiten stark unterschätzt werden aber weniger asymmetrisch erscheinen. Die Trennschärfeparameter werden statt überschätzt unterschätzt.

Ein Blick auf die folgenden Abbildungen (s. Abbildungen 12 bis 15, R-Befehle s. Anhang 6.10.6, S. 59) zeigt, dass die Dichtefunktionen bei symmetrischer Verteilung der Fähigkeiten recht gut geschätzt wird, während dies bei schiefen Verteilungen weniger der Fall ist. Zwar ergibt sich ei-

ne rechtsschiefe geschätzte Dichte, diese ist aber flacher und weniger linkssteil als die entsprechende Dichte der Chi-Quadrat-Verteilung. Um die geschätzte Dichte mit der theoretischen vergleichbar zu machen, mussten bei der uniformen Verteilung diese standardisiert werden: Bei uniformer Verteilung ist die Standardabweichung

$$\sqrt{\frac{(b-a)^2}{12}},$$

im Falle einer uniformen Verteilung auf $[-5, 5]$ ergibt sich damit $\sqrt{\frac{10^2}{12}} = 2.8868$. Entsprechend wurden die Intervallgrenzen standardisiert mit

$$\frac{\pm 5}{\sqrt{\frac{10^2}{12}}} \approx \pm 1.7321.$$

Dies gab dann die folgende theoretische Dichtefunktion:

$$f : \mathbb{R} \rightarrow \mathbb{R}$$

$$f(\theta) := \frac{1}{\frac{5}{\sqrt{\frac{10^2}{12}}} - \frac{-5}{\sqrt{\frac{10^2}{12}}}} \cdot 1_{\left[-\frac{5}{\sqrt{\frac{10^2}{12}}}, \frac{5}{\sqrt{\frac{10^2}{12}}}\right]}(\theta).$$

Bei der Chi-quadrat-Verteilung wurde nicht die theoretische Dichte standardisiert, sondern es wurden die geschätzten Daten (Fähigkeiten) zurücktransformiert mit

$$\left(\hat{\theta} + \frac{5}{\sqrt{2 \cdot 5}}\right) \cdot \frac{5}{\sqrt{2 \cdot 5}}$$

- die Standardabweichung der Chi-Quadrat-Verteilung beträgt $\sqrt{2n}$ mit n Freiheitsgraden. (s. Abbildungen 12 bis 15, R-Befehle Anhang 6.10.6, S. 59)

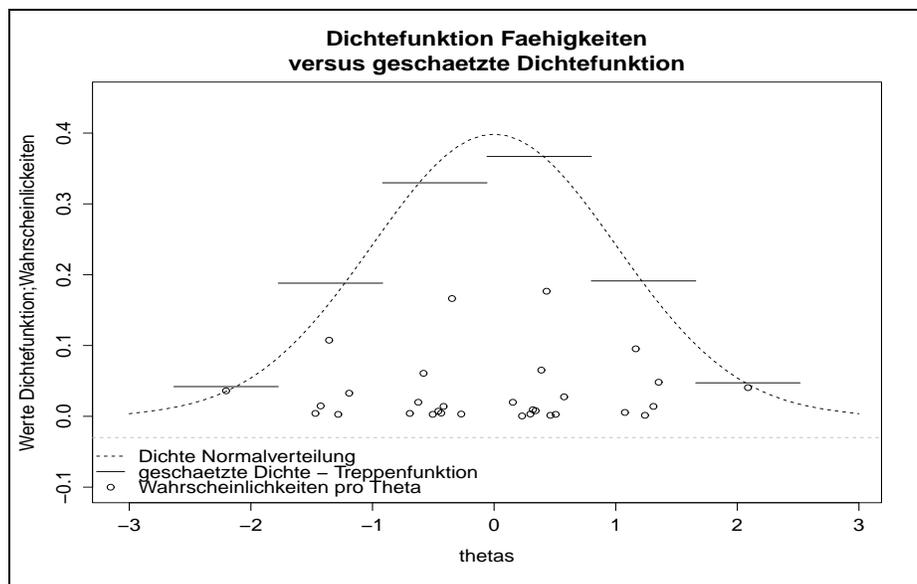


Abbildung 12: Geschätzte Dichtefunktion ($N=1500$, $n=5$, Trennschärfeparameter $(1,1,1,1,1)$, Schwierigkeitsparameter $(-2,-1,0,1,2)$, standardnormalverteilte Fähigkeitsparameter) berechnet nach der Methode der suffizienten Statistiken (50 EM-Durchgänge)

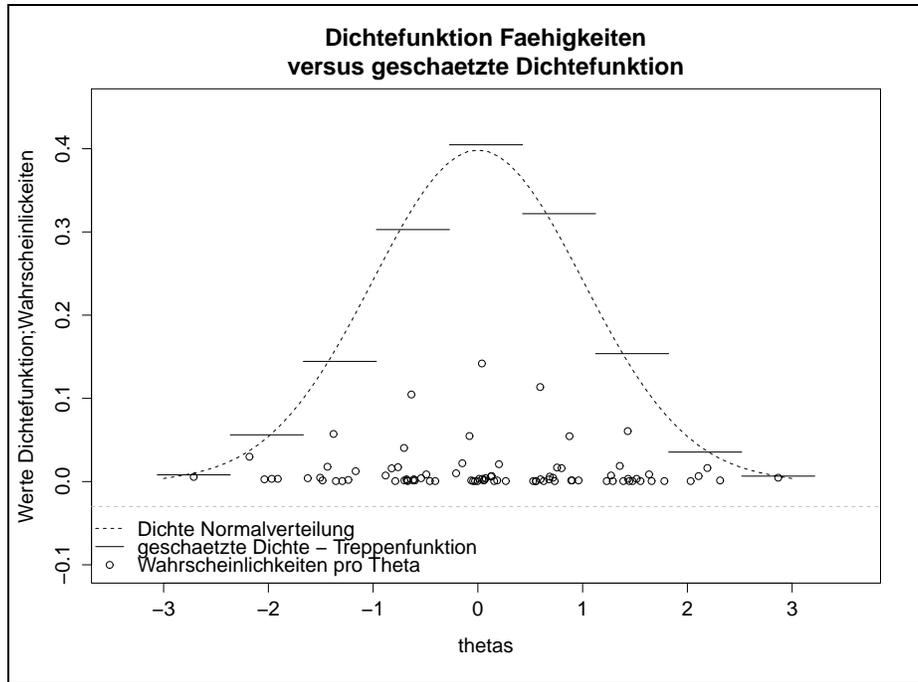


Abbildung 13: Geschätzte Dichtefunktion ($N = 1500$, $n = 8$, Trennschärfeparameter $(1,1,1,1,1,1,1,1)$, Schwierigkeitsparameter $(-3.5,-2.5,-1.5,-0.5,0.5,1.5,2.5,3.5)$, standardnormalverteilte Fähigkeitsparameter) berechnet nach der Methode der suffizienten Statistiken (50 EM-Durchgänge)

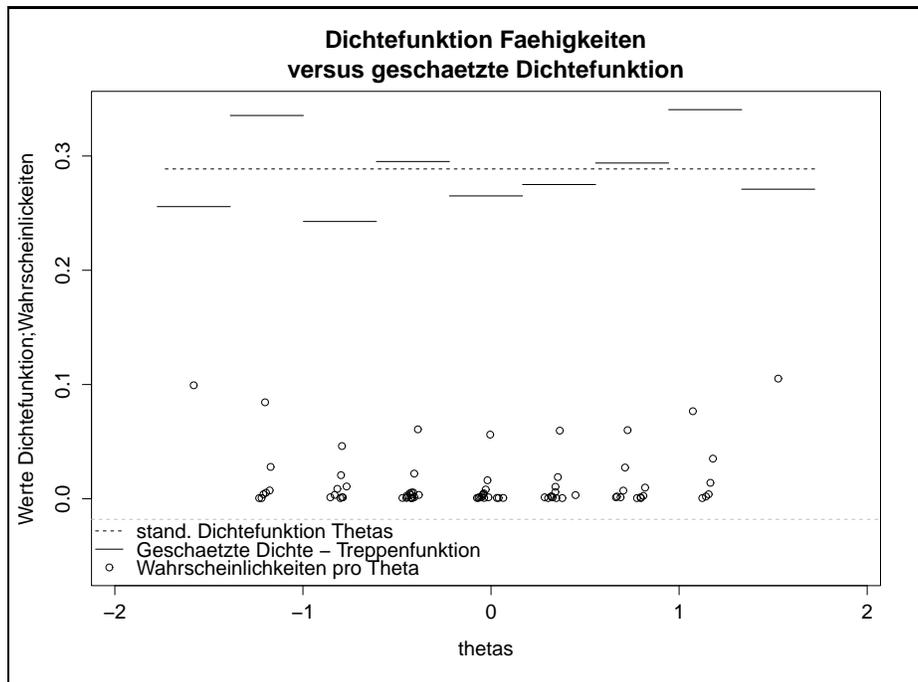


Abbildung 14: Geschätzte Dichtefunktion ($N = 1500$, $n = 8$, Trennschärfeparameter $(1,1,1,1,1,1,1,1)$, Schwierigkeitsparameter $(-3.5,-2.5,-1.5,-0.5,0.5,1.5,2.5,3.5)$, uniform verteilte Fähigkeitsparameter auf $[-5, 5]$) berechnet nach der Methode der suffizienten Statistiken (50 EM-Durchgänge)

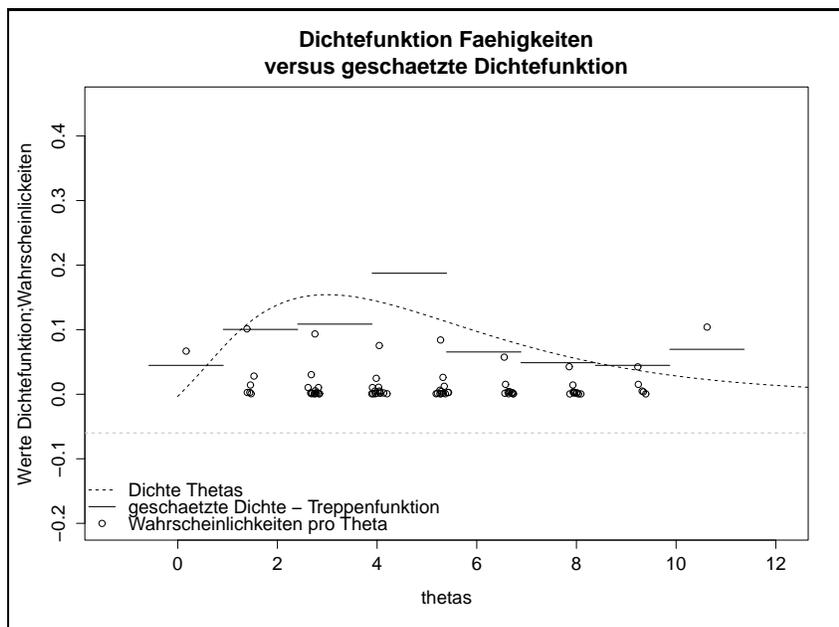


Abbildung 15: Geschätzte Dichtefunktion ($N = 1500$, $n = 8$, Trennschärfeparameter $(1,1,1,1,1,1,1,1)$, Schwierigkeitsparameter $(-3.5,-2.5,-1.5, -0.5,0.5,1.5,2.5,3.5)$, chi-quadrat-verteilte Fähigkeitsparameter, $df = 5$) berechnet nach der Methode der suffizienten Statistiken (50 EM-Durchgänge)

5 Schlusswort

Ziel der Arbeit war es, nach Wegen und Möglichkeiten zu suchen, die Item-Parameter-Schätzung für dichotome, logistische Item Response Theorie-Modelle (IRT-Modelle) von der Annahme der Normalverteiltheit der Fähigkeiten zu befreien. Beim Versuch, das „Graded Item Response-Modell“ (GIRT-Modell) von Fumiko Samejima (1969) für ordinale, polytome Items auf Daten im Rahmen einer Mobbing-Studie im Wallis anzuwenden, ergab sich, dass das Modell nicht passte. Eine erste Vermutung bestand darin, die fehlende Anpassung durch die Asymmetrie der Betroffenheit im Falle von Mobbing zu erklären. Entsprechend sollten ursprünglich Schätzprobleme im Rahmen des GIRT-Modells behandelt werden. Die Schätzprobleme für dieses Modell entsprechen aber den einfacheren für dichotome Items (s. Samejima (1997b)).

Das Schätzverfahren von Birnbaum (JMLE) ist von Verteilungsannahmen unabhängig. Entsprechend wurde zuerst versucht, auf dieses zurückzugreifen. Die Gründe, wieso das Verfahren nicht mehr angewendet wird, wurden aber in der Praxis schnell sichtbar. Es ist völlig instabil. In ersten Software-Implementierungen in den 70er Jahren wurden denn auch für die Parameterschätzungen ad hoc Grenzen für die Parameter eingeführt, um das Verfahren zu stabilisieren.

In der Folge ging es deshalb darum, die MMLE genauer zu verstehen und zu prüfen, wie das Verfahren von der Normalverteiltheitsannahme bezüglich der Fähigkeitsparameter unabhängig gemacht werden könnte. Zuerst wurde die Schätzmethode implementiert, die hier unter dem Untertitel „EM-Schätzverfahren mit Werten der suffizienten Statistiken“ vorgestellt wurde. Sie ist „verteilungsfrei“ und hat zudem für den Einsteiger den Vorteil, mit wirklichen relativen Häufigkeiten statt mit geschätzten zu arbeiten. Beim Programmieren erwies sich, dass das „Wie“ der Implementierung in Bezug auf die Geschwindigkeit eine grosse Rolle spielt. Ein erster Versuch erwies sich als viel zu langsam - es dauerte damit viel länger, für *einen* Datensatz die Parameter zu schätzen als dies mit den später implementierten Verfahren für 200 Datensätze durchzuführen. Ein zweiter Versuch mit möglichst viel Matrizen-Rechnung unter weitgehender Umgehung von Schleifen erwies sich als wesentlich schneller. Es zeigte sich, dass die Implementierung mit dem Maximierungs-

schritt von Bock-Aitkin (Berechnung von logistischen Regressionen) zwar für Einer-Vektoren als Trennschärfeparameter bei normalverteilten Fähigkeitsparametern funktioniert, aber bereits bei anderen Trennschärfeparametern versagt. Bei anderen Verteilungen der Fähigkeitsparameter war das Verfahren völlig instabil. Statt also möglichst schnell zum GIRT-Modell überzugehen, musste das Zwei-Parameter-Modell für dichotome Antwortvektoren näher analysiert werden.

Da das ltm-Paket von Rizopoulos für standardnormalverteilte Fähigkeiten problemlos auch für uniform oder chi-quadrat-verteilte Fähigkeiten funktionierte - abgesehen von der daraus resultierenden fehlenden Modellanpassung - , galt es, die Implementierung von Rizopoulos im ltm.fit-Befehl seines R-Paketes ltm zu untersuchen. Der EM-Algorithmus im ltm.fit-Befehl erwies sich als Glücksfall für die hier vorgenommenen Analysen. Er konnte als Baustein in verschiedenen Schätz-Verfahren verwendet werden und erwies sich als völlig stabil. Es fehlt allerdings noch eine theoretische Begründung der Methode. Auf Nachfrage gab Herr Rizopoulos keine näheren Erklärungen an: er hätte den Algorithmus direkt implementiert und er könne keine weitere Literatur dazu angeben. Eine indirekte Qualifikation ergibt sich für den Algorithmus aber durch die Güte der Schätzungen und die Übereinstimmung mit dem Bock-Aitkin-Verfahren im Falle standardnormalverteilter Fähigkeiten. Der EM-Algorithmus von Rizopoulos erlaubte es in der Folge, die Auswirkungen nicht-normalverteilter Fähigkeiten auf die analysierten Schätzverfahren zu analysieren.

Es zeigte sich, dass die untersuchten Schätzmethode - abgesehen vom Verfahren „Empirische Verteilung“ bei uniformer Start-W-Funktion - gut für standardnormalverteilte Fähigkeiten funktionieren. Besonders interessant: auch das „verteilungsfreie“ Verfahren mit suffizienten Statistiken führt in diesem Fall zu sehr guten Schätzern. Standardnormalverteilte Fähigkeiten sind also für die untersuchten Verfahren besonders geeignet. Bei anderen Verteilungen funktionieren die Methoden eher schlecht. Bei der Methode mit den suffizienten Statistiken zeigt sich bei uniform verteilten Fähigkeiten, die durch die Schwierigkeitsparameter gut abgedeckt sind, dass mittels Transformation mit Hilfe der Standardabweichung der uniformen Verteilung der Fähigkeiten gute Schätzer berechnet werden könnten. Dies ist in der Praxis allerdings nicht hilfreich, da man weder Verteilung der Fähigkeiten noch die Güte der Abdeckung kennt. Bei schiefsymmetrischen Verteilungen sind die Verfahren nicht vertrauenserweckend.

In der Literatur werden weitere Vorschläge gemacht, wie man mit widerspenstigen Daten umgehen könnte. Erwähnt seien etwa der Vorschlag von Samejima (1997a), nicht punktsymmetrische ICCs zu verwenden, oder die vielfältigen Vorschläge mit Bayesschen Schätzmethode (Fox, 2010) - aus Zeitgründen konnten sie hier nicht berücksichtigt werden.

Zum Schluss noch eine Bemerkung zur Skala von Daten (Fähigkeiten), die mittels IRT geschätzt werden. Für den Fall der dichotomen, logistischen Modelle wurde nachgewiesen, dass die IRT bei überabzählbar unendlich vielen Items zu intervallskalierten Daten führt (s. Anhang 6.1). Dies ist ein bemerkenswertes Ergebnis. Lange glaubte man, dass Intervallskalen eine empirische Operation des Zusammenfügens voraussetzen wie z. B. das Aneinanderfügen von Stäben auf einer Geraden oder das Aufeinanderschichten von Gewichten auf einer Waage (s. für eine ausführliche Darlegung der Voraussetzungen des metrischen Messens Stegmüller (1970, S. 19-109)). Diese Operation muss durch die Skala f mit Hilfe der Addition in den rationalen Zahlen gespiegelt werden können, d.h. es muss gelten $f(x) + f(y) = f(x * y)$ - wobei $x * y$ das durch die Operation des Zusammenfügens aus den Objekten x und y entstandene neue Objekt aus der Menge der zu messenden Objekte ist.

Allerdings werden in der Praxis nur wenige Items zur Messung verwendet werden können. Betrachtet man die Abbildungen 12 bis 15, fällt auf, dass die Daten auf der θ -Achse in Klumpen auftreten. Die ursprünglichen Daten sind demgegenüber gleichmässiger verteilt. Zudem werden Personen mit unterschiedlichen Fähigkeiten aber zufälligerweise gleichen Antwortvektoren gleiche Fähigkeiten zugordnet. Die ursprünglichen Distanzen zwischen den Daten werden so nicht korrekt geschätzt - aber um Distanzen geht es bei Intervallskalen. Allerdings werden durch die IRT-Methoden die Fähigkeiten differenzierter geschätzt als etwa beim Zusammenzählen von Punkten. Bezüglich der Anwendung von Regressionsanalysen auf durch IRT-Methoden produzierte Daten kann man bemerken, dass man damit Zusammenhänge zwischen Variablen aufzeigen kann, die berechneten Koeffizienten aber keinesfalls als Schätzung der Kennzahlen des tatsächlichen Zusammenhangs zwischen den „wirklichen“ Variablen gedeutet werden können (s. Anhang 6.9, S. 55).

6 Anhang

6.1 Skalen

Definition 1. Sei eine Menge B von zu messenden Objekten gegeben und eine Zerlegung P von B mit den Elementen A_i (= Äquivalenzklassen). Dann wird eine Abbildung $f : B \rightarrow \mathbb{R}$ (Messskala (bezüglich P)) genannt genau dann, wenn: $f(x) = f(y)$ genau dann, wenn $x, y \in A_i$.

Erfüllt eine Skala nur die Bedingungen der Definition 1, wird sie „Nominalskala“ genannt. Nominalskalen sind bestimmt bis auf Transformationen h , die injektiv sind: Wenn f eine Nominalskala bezüglich P ist, so ist jede Funktion $g : B \rightarrow \mathbb{R}$ mit

- $g = h \circ f$ und
- $h \circ f(x) = h \circ f(y)$ genau dann, wenn $f(x) = f(y)$

eine Nominalskala bezüglich P (\circ ist die Verkettung von Funktionen).

Definition 2. Sei eine Menge B von zu messenden Objekten gegeben und eine Zerlegung P von B mit den Elementen A_i . Sei f eine Skala. f wird „Ordinalskala“ (bezüglich P und T) genannt genau dann, wenn

- es eine asymmetrische, irreflexive und transitive Relation T auf B gibt, so dass $(x, y) \in T$ oder $(y, x) \in T$ genau dann, wenn x und y zu verschiedenen Äquivalenzklassen A_i gehören, und
- wenn $(x, y) \in T$ mit $x \in A_i$ und $y \in A_j$, dann ist $(x', y') \in T$ für alle $x' \in A_i$ und $y' \in A_j$
- es gilt $f(x) > f(y)$ genau dann, wenn $(x, y) \in T$.

Ordinalskalen sind bestimmt bis auf strikt monoton steigende Transformationen h . Wenn f eine Ordinalskala bezüglich P und T ist, so ist jede Funktion $g : B \rightarrow \mathbb{R}$ mit

- $g = h \circ f$ und
- h ist strikt monoton steigend

eine Ordinalskala bezüglich P und T . Zu beachten ist, dass durch solche Transformationen Abstandsverhältnisse zwischen Skalenwerten nicht erhalten bleiben. Gibt es z.B. drei Äquivalenzklassen A_i und sei f mit $f(x_1) = 1$, $f(x_2) = 2$ und $f(x_3) = 3$, $x_i \in A_i$, eine Ordinalskala. Dann ist auch g mit $g(x_1) = 1$, $g(x_2) = 10$ und $g(x_3) = 1000$ eine Ordinalskala, wobei man g durch die Verkettung $h \circ f$ erhält und $h : \{1, 2, 3\} \rightarrow \mathbb{R}$ eine strikt monoton steigende Funktion ist mit $h(1) = 1$, $h(2) = 10$, $h(3) = 1000$. Es gilt

$$\frac{f(x_2) - f(x_1)}{f(x_3) - f(x_2)} = 1 \neq \frac{g(x_2) - g(x_1)}{g(x_3) - g(x_2)} = \frac{9}{990}.$$

Definition 3. Sei eine Menge B von zu messenden Objekten gegeben und eine Zerlegung P von B mit den Elementen A_i . Sei f eine Ordinalskala bezüglich P und T . Dann ist f eine Intervallskala (bezüglich P und T) genau dann, wenn f durch den Messvorgang bis auf affine Transformationen $h(f(x)) = af(x) + b$ mit $b \in \mathbb{R}$, $a \in \mathbb{R}^+ \setminus \{0\}$ bestimmt ist.

Intervallskalen sind etwa dann gegeben, wenn es eine empirische Verknüpfungen $*$ gibt, so dass $f(x * y) = f(x) + f(y)$ für $x, y, x * y \in B$. Beispiele für solche Verknüpfungen sind das Aneinanderfügen von Stäben auf einer Geraden oder das Aufeinanderlegen von Gewichten in einer Waagschale. Präziser gilt:

Satz 4. Sei eine Menge B von zu messenden Objekten gegeben und eine Zerlegung P von B mit den Elementen A_i . Sei f eine Ordinalskala bezüglich P und T . Wenn es eine empirische Verknüpfungsrelation $*$ gibt, so dass $f(x * y) = f(x) + f(y)$ und wenn es für alle $r \in \mathbb{Q}$ ein $x \in B$ gibt, so dass $f(x) = r$, dann ist f durch den Messvorgang bis auf affine Transformationen $h(f(x)) = af(x) + b$ mit $b \in \mathbb{R}$, $a \in \mathbb{R}^+ \setminus \{0\}$ bestimmt.

Beweis. Seien zwei Ordinalskalen f und g bezüglich P und T gegeben, so dass jede die Voraussetzungen des Satzes erfüllt. Zu zeigen ist:

Es gibt ein $a > 0$ und $a, b \in \mathbb{R}$, so dass $g(x) = af(x) + b$.

Gemäss Voraussetzung gibt es ein $x_0 \in B$, so dass $f(x_0) = 0$ und ein $y_0 \in B$, so dass $g(y_0) = 0$.

OBdA sei $(y_0, x_0) \in T$. Es gibt damit ein $z \in B$, so dass

$$x_0 * z = y_0$$

und folglich

$$g(x_0 * z) = g(x_0) + g(z) = g(y_0) = 0.$$

Damit ist

$$b := g(x_0) = -g(z).$$

Sei $g(x) = h(f(x))$. Wir versuchen zu zeigen, dass damit gilt $h(f(x)) = af(x) + b$. Mit der Voraussetzung gilt:

$$\begin{aligned} g(x) + b &= g(x) + g(x_0) \\ &= g(x * x_0) \\ &= h(f(x * x_0)) \\ &= h(f(x) + f(x_0)) \\ &= h(f(x)) \end{aligned}$$

und entsprechend

$$\begin{aligned} g(x) &= h(f(x)) - b \\ g(y) &= h(f(y)) - b. \end{aligned}$$

Setzt man

$$h'(f(x)) := h(f(x)) - b$$

gilt mit

$$\begin{aligned} g(x * y) &= h(f(x * y)) - b \\ &= h(f(x) + f(y)) - b \\ &= h'(f(x) + f(y)) \\ g(x) &= h(f(x)) - b = h'(f(x)) \\ g(y) &= h(f(y)) - b = h'(f(y)). \end{aligned}$$

Also ist

$$h'(f(x) + f(y)) = h'(f(x)) + h'(f(y)).$$

Es handelt sich um eine der Funktionalgleichung von Cauchy (1882-1974, S. 98 f.), welche in \mathbb{Q} die Lösung

$$h'(f(x)) = af(x)$$

aufweist. Mit $h'(f(x)) = h(f(x)) - b$ gilt

$$h(f(x)) = af(x) + b.$$

Damit $g(x) > g(y)$ für $(x, y) \in T$, muss $a > 0$ gelten. □

Lange glaubte man, für Intervallskalen sei die Existenz von empirischen Operationen der beschriebenen Art nötig. Der Beweis des folgenden Satzes zeigt, dass dem nicht so ist und dass durch die IRT-Verfahren unter bestimmten Bedingungen intervallskalierte Daten erzeugt werden. Der Beweis folgt Fischer und Molenaar (1995, S. 20 f.):

Satz 5. *Wenn es für alle Schwierigkeitsparameter β mit $\beta \in \mathbb{R}$ ein Item gibt mit ICC der Form $f(\theta - \beta)$, so dass $f: \mathbb{R} \rightarrow]0, 1[$ eine stetige, bijektive und steigende Funktion ist, dann sind β und θ bis auf affine Transformationen $a\theta + b_1$ und $a\beta + b_2$ mit $b_i, a \in \mathbb{R}$, $a > 0$ bestimmt.*

Beweis. Sei ein Modell mit ICCs der Form $f(\theta - \beta)$ (stetig, strikt monoton steigend mit Werten in $]0, 1[$) gegeben. Für jedes Individuum mit dem Fähigkeitsparameter $\theta \in \mathbb{R}$ und dem Schwierigkeitsparameter β ist die Wahrscheinlichkeit einer korrekten Antwort dadurch festgelegt. Wir nehmen an, es gebe eine weitere Darstellung, die mit diesen Wahrscheinlichkeiten übereinstimmt, mit Parametern $k(\theta)$ und $l(\beta)$ und mit ICC $m(k(\theta) - l(\beta))$, so dass

$$f(\theta - \beta) = m(k(\theta) - l(\beta)), \quad (16)$$

mit strikt monoton steigenden Funktionen k und l , wobei m durch (16) definiert wird. Damit gilt

$$m^{-1}(f(\theta - \beta)) = k(\theta) - l(\beta). \quad (17)$$

Für $\beta = 0$ erhält man:

$$\begin{aligned} m^{-1}(f(\theta - 0)) &= k(\theta) - l(0) \implies \\ m^{-1}(f(\theta)) + l(0) &= k(\theta) \end{aligned}$$

Mit $h := m^{-1}(f)$ und $l_0 := l(0)$ gilt

$$k(\theta) = h(\theta) + l_0. \quad (18)$$

Ähnlich erhält man für $\theta = 0$ und $k_0 := k(0)$

$$m^{-1}(f(0 - \beta)) = k(0) - l(\beta) \implies$$

und damit

$$\begin{aligned} l(\beta) &= -m^{-1}(f(-\beta)) + k(0) \\ &= -h(-\beta) + k_0. \end{aligned} \quad (19)$$

In (17) eingesetzt und mit der Definition von h ergibt sich

$$\begin{aligned} h(\theta - \beta) &= m^{-1}(f(\theta - \beta)) \\ &= h(\theta) + l_0 - (-h(-\beta) + k_0) \\ &= h(\theta) + l_0 + h(-\beta) - k_0. \end{aligned} \quad (20)$$

Mit $\theta = \beta = 0$ ergibt sich daraus

$$\begin{aligned} h(0) &= h(0) + l_0 + h(0) - k_0 \iff \\ 0 &= l_0 + h(0) - k_0 \iff \\ h(0) &= k_0 - l_0, \end{aligned}$$

womit in die Gleichung (20) eingesetzt

$$\begin{aligned} h(\theta - \beta) &= h(\theta) + l_0 + h(-\beta) - k_0 \\ &= h(\theta) - h(0) + h(-\beta) \end{aligned}$$

folgt. Zieht man von beiden Seiten $h(0)$ ab, erhält man

$$h(\theta - \beta) - h(0) = h(\theta) - h(0) + h(-\beta) - h(0).$$

Mit

$$h^*(x) := h(x) - h(0)$$

und $\beta' := -\beta$ ergibt sich daraus Funktionalgleichung von Cauchy (s. Seite 38)

$$h^*(\theta + \beta') = h^*(\theta) + h^*(\beta').$$

Diese hat in \mathbb{R} unter den getroffenen Voraussetzungen (Stetigkeit, strikte Monotonie), die Lösung $h^*(x) = ax$, $a > 0$. Da $h^*(x) = h(x) - h(0)$ ergibt sich daraus: $h(x) = ax + h(0)$
Aus (18) und (19) folgt

$$\begin{aligned} k(\theta) &= h(\theta) + l_0 \\ &= a\theta + h(0) + l_0 \end{aligned}$$

und

$$\begin{aligned} l(\beta) &= -h(-\beta) + k_0 \\ &= -(a(-\beta) + h(0)) + k_0 \\ &= a\beta - h(0) + k_0. \end{aligned}$$

Damit ergeben sich für β und θ die zulässigen Transformationen $a\theta + h(0) + l_0 =: a\theta + b_1$ und $a\beta - h(0) + k_0 =: a\beta + b_2$, so dass $b_1, b_2, a \in \mathbb{R}$ und $a > 0$. \square

Da ICCs der Form $\text{ICC}(\theta) = \frac{\exp(a(\theta - \beta))}{1 + \exp(a(\theta - \beta))}$ strikt monoton und stetig sind, sind dadurch θ und β bis auf affine Transformationen bestimmt.

Die Voraussetzung, dass es ein überabzählbar unendlich viele Items gibt, könnte man als anspruchsvoll ansehen. Fischer und Molenaar (1995, S. 21 ff.) diskutiert Abschwächungen dieser Voraussetzung und beweist den Satz: Wenn es eine endliche Menge von Items I_i mit $i \in \mathbb{N}_k^*$, $k \geq 3$, mit ICCs der Form $f(\theta - \beta_i)$ für $\beta_i, \theta \in \mathbb{R}$ und f ist stetig, bijektiv und steigend ($\mathbb{R} \rightarrow (0, 1)$) und wenn alle Quotienten $\frac{\beta_j - \beta_i}{\beta_q - \beta_v}$ rationale Zahlen sind, dann sind die θ eindeutig bestimmt bis auf positive lineare Transformationen, an allen Punkten $\theta_{in} = \beta_i + n\Delta$ für $n \in \mathbb{N}$ und $i \in \mathbb{N}_k^*$, wobei Δ die grösste reelle Zahl ist, so dass $\beta_j - \beta_i = w_{ij}\Delta$ für alle Paare (I_i, I_j) und w_{ij} ist eine ganze Zahl. An allen Punkte zwischen den θ_{in} sind die θ eindeutig bestimmt bis auf stetige, strikt monotone Transformationen. Wenn es mindestens 3 Items gibt, so dass $\frac{\beta_j - \beta_i}{\beta_q - \beta_v}$ irrational ist, ist die θ -Skala eine Intervallskala.

6.2 Eigenschaften logistischer ICCs

•

$$\begin{aligned}
 \frac{\exp(\alpha(\theta - \beta))}{1 + \exp(\alpha(\theta - \beta))} &= \frac{\exp(\alpha\theta - \alpha\beta)}{1 + \exp(\alpha\theta - \alpha\beta)} \\
 &= \frac{\frac{\exp(\alpha\theta)}{\exp(\alpha\beta)}}{1 + \frac{\exp(\alpha\theta)}{\exp(\alpha\beta)}} \\
 &= \frac{\frac{\exp(\alpha\theta)}{\exp(\alpha\beta)}}{\frac{\exp(\alpha\beta) + \exp(\alpha\theta)}{\exp(\alpha\beta)}} \\
 &= \frac{\exp(\alpha\theta)}{\exp(\alpha\beta) + \exp(\alpha\theta)} \\
 &= \frac{1}{\frac{\exp(\alpha\beta) + \exp(\alpha\theta)}{\exp(\alpha\theta)}} \\
 &= \frac{1}{1 + \exp(\alpha\beta - \alpha\theta)} \\
 &= \frac{1}{1 + \exp(\alpha(\beta - \theta))}
 \end{aligned}$$

- p ist strikt monoton steigend für $\alpha > 0$.

$$\begin{aligned}
 \frac{d}{d\theta} \frac{\exp(\alpha(\theta - \beta))}{1 + \exp(\alpha(\theta - \beta))} &= \frac{\alpha \exp(\alpha(\theta - \beta)) (1 + \exp(\alpha(\theta - \beta))) - \exp(\alpha(\theta - \beta)) \alpha \exp(\alpha(\theta - \beta))}{(1 + \exp(\alpha(\theta - \beta)))^2} \\
 &= \frac{\alpha \exp(\alpha(\theta - \beta)) [1 + \exp(\alpha(\theta - \beta)) - \exp(\alpha(\theta - \beta))]}{(1 + \exp(\alpha(\theta - \beta)))^2} \\
 &= \frac{\alpha \exp(\alpha(\theta - \beta))}{(1 + \exp(\alpha(\theta - \beta)))^2}
 \end{aligned}$$

Da

$$\frac{\exp(\alpha(\theta - \beta))}{(1 + \exp(\alpha(\theta - \beta)))^2} > 0,$$

hängt die Steigung nur vom Vorzeichen von α ab.

- Die Steigung α ist maximal in θ mit $p(\theta) = 0.5$ und zwar in $\theta = \beta$.
Mit

$$c := c(\theta) := \exp(\alpha(\theta - \beta))$$

gilt

$$\frac{d}{d\theta} c = \alpha c$$

und damit

$$\begin{aligned}
 \frac{d}{d\theta} \left(\alpha \frac{\exp(\alpha(\theta - \beta))}{(1 + \exp(\alpha(\theta - \beta)))^2} \right) &= \frac{d}{d\theta} \left(\alpha \frac{c}{(1 + c)^2} \right) \\
 &= \alpha \frac{\alpha c (1 + c)^2 - c^2 (1 + c) \alpha c}{(1 + c)^4} \\
 &= \alpha \frac{\alpha c (1 + c) [1 + c - 2c]}{(1 + c)^4} \\
 &= \alpha^2 \frac{c (1 + c) [1 - c]}{(1 + c)^4} \\
 &= \alpha^2 \frac{c (1 - c)}{(1 + c)^3}
 \end{aligned}$$

Mit Nullsetzen:

$$\begin{aligned}
 \alpha^2 \frac{c(1-c)}{(1+c)^3} = 0 &\iff \\
 1 - c = 0 &
 \end{aligned}$$

und damit

$$\begin{aligned}
 1 - \exp(\alpha(\theta - \beta)) = 0 &\iff \\
 \exp(\alpha(\theta - \beta)) = 1 &\iff \\
 \alpha(\theta - \beta) = 0 &\iff \\
 \theta - \beta = 0 &\iff \\
 \theta = \beta &
 \end{aligned}$$

Eingesetzt ergibt sich

$$p(\beta) = \frac{\exp(\alpha(\beta - \beta))}{1 + \exp(\alpha(\beta - \beta))} = \frac{1}{2}$$

Damit liegt dort ein Extremum. Die zweite Ableitung ist

$$\begin{aligned}
 \frac{d}{d\theta} \alpha^2 \frac{c(1-c)}{(1+c)^3} &= \alpha^2 \frac{[\alpha c(1-c) + c(-\alpha c)](1+c)^3 - c(1-c)3(1+c)^2 \alpha c}{(1+c)^6} \\
 &= \alpha^2 \frac{[\alpha c(1-c) + c(-\alpha c)](1+c) - c(1-c)3\alpha c}{(1+c)^4} \\
 &= \alpha^2 \frac{[\alpha c - \alpha c^2 - \alpha c^2](1+c) - c(1-c)3\alpha c}{(1+c)^4} \\
 &= \alpha^3 c \frac{[1-2c](1+c) - c(1-c)3}{(1+c)^4} \\
 &= \alpha^3 c \frac{1-2c^2-c-3c+3c^2}{(1+c)^4} \\
 &= \alpha^3 c \frac{1-4c+c^2}{(1+c)^4}
 \end{aligned}$$

In $\theta = \beta$ gilt: $c(\beta) = \exp(\alpha(\beta - \beta)) = \exp(\alpha(0.5 - 0.5)) = 1$ und damit

$$\alpha^3 \frac{1-4+1}{(1+1)^4} = \alpha^3 \frac{-2}{2^4} < 0$$

für $\alpha > 0$. Entsprechend ist die Steigung in $\theta = \beta$ maximal.

- Im Punkt β hat die Kurve die Steigung $\frac{\alpha}{4}$.

$$p'(\beta) = \alpha \frac{\exp(\alpha(\beta - \beta))}{(1 + \exp(\alpha(\beta - \beta)))^2} = \alpha \frac{1}{(1 + 1)^2} = \frac{\alpha}{4}$$

- Die Log-Odds der logistischen ICC sind eine affine Funktion der Parameter α und β .

$$\begin{aligned} \ln\left(\frac{p(\theta)}{1 - p(\theta)}\right) &= \ln\frac{\frac{\exp(\alpha(\theta - \beta))}{1 + \exp(\alpha(\theta - \beta))}}{1 - \frac{\exp(\alpha(\theta - \beta))}{1 + \exp(\alpha(\theta - \beta))}} \\ &= \ln\frac{\frac{\exp(\alpha(\theta - \beta))}{1 + \exp(\alpha(\theta - \beta))}}{\frac{1 + \exp(\alpha(\theta - \beta)) - \exp(\alpha(\theta - \beta))}{1 + \exp(\alpha(\theta - \beta))}} \\ &= \ln \exp(\alpha(\theta - \beta)) \\ &= \alpha(\theta - \beta) \end{aligned}$$

- Logistische ICCs mit unterschiedlicher maximaler Trennschärfe schneiden sich in

$$\frac{\alpha_1\beta_1 - \alpha_2\beta_2}{\alpha_1 - \alpha_2},$$

denn

$$\begin{aligned} \frac{\exp(\alpha_1(\theta - \beta_1))}{1 + \exp(\alpha_1(\theta - \beta_1))} &= \frac{\exp(\alpha_2(\theta - \beta_2))}{1 + \exp(\alpha_2(\theta - \beta_2))} \iff \\ \exp(\alpha_1(\theta - \beta_1))(1 + \exp(\alpha_2(\theta - \beta_2))) &= \exp(\alpha_2(\theta - \beta_2))(1 + \exp(\alpha_1(\theta - \beta_1))) \iff \\ \exp(\alpha_1(\theta - \beta_1)) &= \exp(\alpha_2(\theta - \beta_2)) \iff \\ \alpha_1\theta - \alpha_1\beta_1 &= \alpha_2\theta - \alpha_2\beta_2 \iff \\ \theta &= \frac{\alpha_1\beta_1 - \alpha_2\beta_2}{\alpha_1 - \alpha_2}. \end{aligned}$$

Also weisen zwei logistische ICCs einen Schnittpunkt auf genau dann, wenn $\alpha_1 - \alpha_2 \neq 0$, d.h. $\alpha_1 \neq \alpha_2$

6.3 Newton-Raphson-Verfahren

Das Maximum der LL-Funktion der logistischen Regression wird nicht wie sonst bei ableitbaren Funktionen üblich mittels Ableitung und Nullsetzung berechnet, sondern direkt mit Hilfe der sogenannten Newton-Raphson-Methode. Es handelt sich um ein iteratives Verfahren. Die Newton-Raphson-Methode setzt voraus, dass (i) die Funktion in einer Umgebung U der Lage (a, b) des Maximums konkav ist und (ii) der Startpunkt (a_1, b_1) des Iterationsverfahrens sowie die Zwischenergebnisse (a_i, b_i) in U liegen. Konkret wird ein Startpunkt (a_1, b_1) gewählt, dessen LL-Funktionswert in der Nähe des Maximums liegt, z.B. mittels Berechnung der klassischen linearen Regression durch die empirischen Logits

$$\ln\left(\frac{x_i + 0.5}{n_i - x_i + 0.5}\right).$$

In diesem Punkt wird dann das Taylor-Polynom 2. Grades g berechnet:

$$g: \mathbb{R}^2 \rightarrow \mathbb{R}$$

$$g(\mathbf{a}) := \ln f(\mathbf{a}_1) + \mathbf{q}_{\mathbf{a}_1}^\top (\mathbf{a} - \mathbf{a}_1) + \frac{1}{2} (\mathbf{a} - \mathbf{a}_1)^\top \mathbf{H}_{\mathbf{a}_1} (\mathbf{a} - \mathbf{a}_1)$$

mit

$$\mathbf{a} = \begin{pmatrix} a \\ b \end{pmatrix},$$

$$\mathbf{a}_1 = \begin{pmatrix} a_1 \\ b_1 \end{pmatrix} \quad \text{und}$$

$$\mathbf{q} = \begin{pmatrix} \frac{\partial \ln f(a,b)}{\partial a} \\ \frac{\partial \ln f(a,b)}{\partial b} \end{pmatrix},$$

dem Vektor der ersten partiellen Ableitungen der logistischen LL-Funktion nach a und nach b sowie der Hesse-Matrix

$$\mathbf{H} = \begin{pmatrix} \frac{\partial^2 \ln f(a,b)}{\partial a \partial a} & \frac{\partial^2 \ln f(a,b)}{\partial b \partial a} \\ \frac{\partial^2 \ln f(a,b)}{\partial a \partial b} & \frac{\partial^2 \ln f(a,b)}{\partial b \partial b} \end{pmatrix},$$

d.h. der Matrix der zweiten partiellen Ableitungen der logistischen LL-Funktion nach a und b .

Der Vektor

$$\mathbf{q}_{\mathbf{a}_1}$$

entsteht aus \mathbf{q} durch Ersetzen von a und b durch a_1 und b_1 . Dies gilt auch für die Matrix

$$\mathbf{H}_{\mathbf{a}_1},$$

die aus \mathbf{H} durch Ersetzen von a und b durch a_1 und b_1 entsteht.

Das Taylor-Polynom g hat in \mathbf{a}_1 dieselbe maximale Steigung wie die LL-Funktion $\ln f$ und zwar in dieselbe Richtung sowie dasselbe (konkave) Krümmungsverhalten wie $\ln f$. Es hat entsprechend das Maximum von \mathbf{a} ausgesehen ungefähr oder genau in dieselbe Richtung wie $\ln f$. \mathbf{a}_2 sei die Lage dieses Maximums des Taylorpolynoms. Im nächsten Schritt wird in \mathbf{a}_2 das Taylor-Polynom 2. Grades berechnet, wiederum die Lage des Maximums bestimmt etc. Im Allgemeinen konvergiert das Verfahren schnell, so dass gewöhnlich nur wenige Durchgänge nötig sind, um ein gegebenes Präzisionsniveau zu erreichen - definiert durch die Differenz zweier aufeinanderfolgender Näherungswerte, die eine vorgegebene Grenze unterschreitet.

Durch Ableiten von g und Nullsetzen erhält man

$$g'(\mathbf{a}) = \mathbf{q}_{\mathbf{a}_1} + \mathbf{H}_{\mathbf{a}_1} (\mathbf{a} - \mathbf{a}_1) = 0$$

und durch Auflösen nach \mathbf{a} unter der Voraussetzung der Regularität von $\mathbf{H}_{\mathbf{a}_1}$ die Lage des Maximums von g

$$\mathbf{a}_2 := \mathbf{a} = \mathbf{a}_1 - \mathbf{H}_{\mathbf{a}_1}^{-1} \mathbf{q}_{\mathbf{a}_1}.$$

Für die $t + 1$ -te Iteration gilt dabei

$$\mathbf{a}_{t+1} = \mathbf{a}_t - \mathbf{H}_{\mathbf{a}_t}^{-1} \mathbf{q}_{\mathbf{a}_t}.$$

Diese Methoden werden nun auf die logistische Regression angewendet. Für

$$\pi : \mathbb{R}^2 \rightarrow]0, 1[$$

$$\pi(a, b) := \frac{\exp(a\theta_i + b)}{1 + \exp(a\theta_i + b)}$$

gilt (s. Seite 41)

$$\frac{\partial}{\partial a} (\pi(a, b)) = \theta_i \pi(a, b) (1 - \pi(a, b)) \quad \text{und}$$

$$\frac{\partial}{\partial b} (\pi(a, b)) = \pi(a, b) (1 - \pi(a, b)).$$

Damit ergeben sich mit

$$\frac{\partial}{\partial a} \ln(\pi(a, b)) = \frac{\theta_i \pi(a, b) (1 - \pi(a, b))}{\pi(a, b)} = \theta_i (1 - \pi(a, b)),$$

$$\begin{aligned}\frac{\partial}{\partial b} \ln(\pi(a, b)) &= \frac{\pi(a, b)(1 - \pi(a, b))}{\pi(a, b)} = 1 - \pi(a, b), \\ \frac{\partial}{\partial a} \ln(1 - \pi(a, b)) &= \frac{-\theta_i \pi(a, b)(1 - \pi(a, b))}{1 - \pi(a, b)} = -\theta_i \pi(a, b)\end{aligned}$$

und

$$\frac{\partial}{\partial b} \ln(1 - \pi(a, b)) = \frac{-\pi(a, b)(1 - \pi(a, b))}{1 - \pi(a, b)} = -\pi(a, b)$$

die Ableitungen

$$\begin{aligned}\frac{\partial}{\partial a} \ln f(a, b) &= \frac{\partial}{\partial a} \sum_{i=1}^m x_i \ln \pi(a, b) + (n_i - x_i) \ln(1 - \pi(a, b)) \\ &= \sum_{i=1}^m x_i \theta_i (1 - \pi(a, b)) - (n_i - x_i) \theta_i \pi(a, b) \\ &= \sum_{i=1}^m \theta_i (x_i - n_i \pi(a, b))\end{aligned}\tag{21}$$

und

$$\begin{aligned}\frac{\partial}{\partial b} \ln f(a, b) &= \frac{\partial}{\partial a} \sum_{i=1}^m x_i \ln \pi(a, b) + (n_i - x_i) \ln(1 - \pi(a, b)) \\ &= \sum_{i=1}^m x_i (1 - \pi(a, b)) - (n_i - x_i) \pi(a, b) \\ &= \sum_{i=1}^m (x_i - n_i \pi(a, b)).\end{aligned}\tag{22}$$

Zudem erhält man

$$\begin{aligned}\frac{\partial^2}{\partial a \partial a} \ln f(a, b) &= \frac{\partial}{\partial a} \sum_{i=1}^m (\theta_i x_i - \theta_i n_i \pi(a, b)) \\ &= -\sum_{i=1}^m \theta_i^2 n_i \pi(a, b) (1 - \pi(a, b)),\end{aligned}$$

$$\begin{aligned}\frac{\partial^2}{\partial a \partial b} \ln f(a, b) &= \frac{\partial}{\partial b} \sum_{i=1}^m (\theta_i x_i - \theta_i n_i \pi(a, b)) \\ &= -\sum_{i=1}^m \theta_i n_i \pi(a, b) (1 - \pi(a, b)),\end{aligned}$$

$$\begin{aligned}\frac{\partial^2}{\partial b \partial a} \ln f(a, b) &= \frac{\partial}{\partial a} \sum_{i=1}^m (x_i - n_i \pi(a, b)) \\ &= -\sum_{i=1}^m \theta_i n_i \pi(a, b) (1 - \pi(a, b))\end{aligned}$$

und

$$\begin{aligned}\frac{\partial^2}{\partial b \partial b} \ln f(a, b) &= \frac{\partial}{\partial b} \sum_{i=1}^m (x_i - n_i \pi(a, b)) \\ &= -\sum_{i=1}^m n_i \pi(a, b) (1 - \pi(a, b)),\end{aligned}$$

womit alle Grössen für den Iterationsprozess hergeleitet sind.

Zu beachten ist, dass bei der zweiten Ableitung die Werte x_i der Statistiken X_i verschwinden. Entsprechend fallen die Erwartungswerte $E(l(X_i))$ der Komponenten $l(X_i)$ der Hesse-Matrix mit diesen Komponenten zusammen. Die Matrix der Negation dieser Erwartungswerte der Hesse-Matrix nennt man „Informationsmatrix“. Die Inverse der Informationsmatrix ist die Kovarianzmatrix der Schätzstatistiken der Parameter (Agresti, 2002, S. 10). Die geschätzten Schwierigkeits- und Trennschärfeparameter der logistischen *ICC* erhält man mit den Festlegungen (4) von Seite 9.

6.4 Gauss-Hermite-Integration

Die Gauss-Hermite-Integration (s. z.B. Gautschi (2004)) ist eine Näherungsformel für Integrale der Form

$$\int_{-\infty}^{+\infty} f(x) e^{-x^2} dx.$$

Diese werden mit der Summe

$$\sum_{i=1}^q f(x_i) w_i$$

angenähert - mit Gauss-Hermite-Stütz- oder Knotenpunkten x_i (GH-Knoten) und Gauss-Hermite-Gewichten w_i (GH-Gewichte). Die GH-Knoten sind die Nullstellen der Hermite-Polynome n -ten Grades H_n , die man rekursiv wie folgt definieren kann

$$\begin{aligned} H_0 &:= 1 \\ H_1 &:= 2x \\ H_{n+1}(x) &:= 2xH_n(x) - 2nH_{n-1}(x). \end{aligned}$$

Die Gewichte werden wie folgt bestimmt:

$$w_i = \frac{2^{n-1} n! \sqrt{\pi}}{n^2 (H_{i-1}(x_i))^2}.$$

Da $f(x) e^{-x^2}$ nicht genau die Form von $f(x) \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$ hat, gilt es das Integral

$$\int_{-\infty}^{+\infty} f(y) \frac{1}{\sqrt{2\pi}} e^{-\frac{y^2}{2}} dy$$

durch Substitution geeignet zu transformieren. Mit $y = \sqrt{2}x$ und $\frac{d}{dx}(\sqrt{2}x) = \sqrt{2}$ erhält man

$$\int_{-\infty}^{+\infty} f(\sqrt{2}x) \frac{1}{\sqrt{2\pi}} e^{-\frac{(\sqrt{2}x)^2}{2}} \sqrt{2} dx = \int_{-\infty}^{+\infty} f(\sqrt{2}x) \frac{1}{\sqrt{\pi}} e^{-x^2} dx.$$

Gemäss Gauss-Hermite-Integrationsformel ergibt sich also

$$\int_{-\infty}^{+\infty} f(x) \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} dx \approx \sum_{i=1}^q f(\sqrt{2}x_i) \frac{1}{\sqrt{\pi}} w_i.$$

Die Nullstellen der Hermite-Polynome sind entsprechend mit $\sqrt{2}$ zu multiplizieren und die GH-Gewichte mit $\frac{1}{\sqrt{\pi}}$. Die GH-Stützpunkte und GH-Gewichte sind in manchen Tabellen-Sammlungen aufgeführt, s. etwa (Beyer, 1987, S. 464). Auf dem Internet findet man zudem Möglichkeiten, bis zu einem bestimmten n die Stützpunkte und die Gewichte zu berechnen, z.B. (eFunda, 2014) bis $n = 32$. Im R-Paket *ltm* liegt im Verzeichnis `data` eine Liste der GH-Punkte und Gewichte bis $n = 101$ vor (`gh.rda`). Diese wurde ins R-Paket *irt* übernommen.

6.5 Wahrscheinlichkeitstheoretischer Hintergrund

Definition 6. f ist eine Wahrscheinlichkeitsfunktion (W -Funktion) genau dann, wenn f eine Abbildung $f : \Omega \rightarrow [0, 1]$ ist, so dass Ω abzählbar ist und $\sum_{\omega \in \Omega} f(\omega) = 1$

Relative Häufigkeitsverteilungen sind gemäss dieser Definition W -Funktionen.

Definition 7. f ist eine Dichtefunktion genau dann, wenn f eine Abbildung $f : \mathbb{R} \rightarrow \mathbb{R}^+$ ist, so dass $\int f dx = 1$.

Definition 8. F ist eine Verteilungsfunktion genau dann, wenn F eine Abbildung $\mathbb{R} \rightarrow [0, 1]$ ist, so dass $F(x) = \int_{-\infty}^x f dx$ ist, für eine Dichtefunktion f .

Definition 9. Sei (Ω, \mathcal{A}) ein Messraum (mit einer Sigma-Algebra \mathcal{A} auf Ω) und P ein Wahrscheinlichkeitsmass auf \mathcal{A} , $(\mathbb{R}, \mathcal{B})$ ein Borelscher Messraum, \mathcal{G} eine Unter-Sigma-Algebra von \mathcal{A} , X eine $(\Omega, \mathcal{A}) - (\mathbb{R}, \mathcal{B})$ -messbare, lebesgue-integrierbare Abbildung von Ω nach \mathbb{R} (Zufallsvariable). Eine $(\Omega, \mathcal{G}) - (\mathbb{R}, \mathcal{B})$ -messbare Abbildung $f : \Omega \rightarrow \mathbb{R}$ heisst Bedingter Erwartungswert von X unter \mathcal{G} genau dann, wenn

$$\int_G X dP = \int_G f dP \text{ für alle } G \in \mathcal{G}.$$

f wird mit $E(X|\mathcal{G})$ abgekürzt, wobei f nur P -fast überall bestimmt ist. Die Menge der Bedingten Erwartungswerte wird gewöhnlich ebenfalls mit $E(X|\mathcal{G})$ abgekürzt. Für eine Zufallsvariable $Y : (\Omega, \mathcal{A}, P) \rightarrow (\Omega_1, \mathcal{A}_1)$ schreiben wir in Hinblick auf die durch Y auf Ω erzeugte Unter-Sigma-Algebra $\sigma(\{Y^{-1}(A_1) | A_1 \in \mathcal{A}_1\})$ von \mathcal{A} : $E(X|Y)$, wobei $Y^{-1}(A_1) := \{\omega | Y(\omega) \in A_1\}$.

Ohne Beweis wird festgehalten:

Satz 10. Seien $X : (\Omega, \mathcal{A}, P) \rightarrow (\mathbb{R}, \mathcal{B})$ und $Y : (\Omega, \mathcal{A}, P) \rightarrow (\Omega_2, \mathcal{A}_2)$ Zufallsvariablen, X lebesgue-integrierbar. Dann gibt es eine $(\Omega_2, \mathcal{A}_2) - (\mathbb{R}, \mathcal{B})$ -messbare Abbildung g , so dass $g \circ Y \in E(X|Y)$. g ist P_Y -fast überall bestimmt (P_Y ist das durch Y auf \mathcal{A}_2 erzeugte Bildmass).

Definition 11. Seien $(\Omega_2, \mathcal{A}_2)$ und $(\Omega_1, \mathcal{A}_1)$ zwei Messräume. Eine Abbildung $K : \Omega_2 \times \mathcal{A}_1 \rightarrow \mathbb{R}$ heisst stochastischer Kern von $(\Omega_2, \mathcal{A}_2)$ nach $(\Omega_1, \mathcal{A}_1)$ genau dann, wenn

- $K(\cdot, A_1) : \Omega_2 \rightarrow \mathbb{R}$ ist eine $(\Omega_2, \mathcal{A}_2) - (\mathbb{R}, \mathcal{B})$ -messbare Abbildung für alle $A_1 \in \mathcal{A}_1$ und
- $K(\omega_2, \cdot) : \mathcal{A}_1 \rightarrow \mathbb{R}$ ist ein Wahrscheinlichkeitsmass auf \mathcal{A}_1 für alle $\omega_2 \in \Omega_2$.

Definition 12. Seien $X : (\Omega, \mathcal{A}, P) \rightarrow (\Omega_1, \mathcal{A}_1)$ und $Y : (\Omega, \mathcal{A}, P) \rightarrow (\Omega_2, \mathcal{A}_2)$ Zufallsvariablen, X lebesgue-integrierbar. Ein stochastischer Kern K von $(\Omega_2, \mathcal{A}_2)$ nach $(\Omega_1, \mathcal{A}_1)$ heisst bedingte Verteilung von X bezüglich P unter Y genau dann, wenn $K(\cdot, A_1) \in E(1_{A_1} \circ X|Y)$ für alle $A_1 \in \mathcal{A}_1$.

Satz 13. $K(\cdot, A_1) \in E(1_{A_1} \circ X|Y)$ für alle $A_1 \in \mathcal{A}_1$ genau dann, wenn $\int_{A_2} K(\cdot, A_1) dP_Y = P(X^{-1}(A_1) \cap Y^{-1}(A_2))$ für alle $A_2 \in \mathcal{A}_2, A_1 \in \mathcal{A}_1$.

Beweis.

$$\begin{aligned} \int_{A_2} K(y, A_1) dP_Y &= \int_{Y^{-1}(A_2)} 1_{A_1} \circ X(\omega) dP \\ &= \int_{Y^{-1}(A_2)} X^{-1}(A_1) dP = P(X^{-1}(A_1) \cap Y^{-1}(A_2)) \end{aligned}$$

□

Analog zu $P(X^{-1}(A_1)) := P(\{\omega | X(\omega) \in A_1\}) =: P_X(A_1) =: P(X \in A_1)$ definiert man:
 $P(X^{-1}(A_1) \cap Y^{-1}(A_2)) =: P(X \in A_1, Y \in A_2)$.

Satz 14. Für einen Kern $K(\cdot, A_1) \in E(1_{A_1} \circ X|Y)$ für alle $A_1 \in \mathcal{A}_1$ gilt:
 $K(y, A_1) = P(X^{-1}(A_1) | Y = y)$, P_Y -fast überall.

Beweis. $\int_{\{y\}} K(y, A_1) dP_Y = K(y, A_1) P_Y(y) = K(y, A_1) P(\{\omega | Y(\omega) = y\})$. Mit $P(Y = y) := P(\{\omega | Y(\omega) = y\})$ und $K(y, A_1) = P(X^{-1}(A_1) \cap \{Y = y\})$ folgt: $K(y, A_1) = \frac{P(X^{-1}(A_1) \cap \{Y=y\})}{P(Y=y)}$ und damit der Satz für $P(Y = y) \neq 0$, was P_Y -fast überall gilt. \square

Ein Kern $K(y, A_1) \in E(1_{A_1} \circ X | Y)$ für alle $A_1 \in \mathcal{A}_1$ ordnet also für alle y jedem $A_1 \in \mathcal{A}_1$ die bedingte Wahrscheinlichkeit

$$P(X^{-1}(A_1) | Y = y)$$

zu, was den Namen „Bedingte Verteilung“ für K erklärt.

Definition 15. Seien $X : (\Omega, \mathcal{A}) \rightarrow (\Omega_1, \mathcal{A}_1)$ und $Y : (\Omega, \mathcal{A}) \rightarrow (\Omega_2, \mathcal{A}_2)$ P -verteilte Zufallsvariablen und μ ein σ -endliches Mass auf \mathcal{A}_1 . Eine $(\mathcal{A}_1 \otimes \mathcal{A}_2)$ -messbare Funktion $f : \Omega_1 \times \Omega_2 \rightarrow \mathbb{R}^+$ heisst bedingte μ -Dichte von X bezüglich des W -Masses P unter Y genau dann, wenn der durch

$$K(\omega_2, A_1) := \int_{A_1} f_{\omega_2}(\omega_1) d\mu,$$

(für alle $\omega_2 \in \Omega_2$ und $A_1 \in \mathcal{A}_1$) definierte Kern K von Ω_2 nach Ω_1 der Bedingung

$$\begin{aligned} \int_{A_2} K(\omega_2, A_1) dP_Y(\omega_2) &= \int_{A_2} \left(\int_{A_1} f_{\omega_2}(\omega_1) d\mu \right) dP_Y(\omega_2) \\ &= P(X^{-1}(A_1) \cap Y^{-1}(A_2)) \end{aligned}$$

genügt (für alle $A_1 \in \mathcal{A}_1$ und $A_2 \in \mathcal{A}_2$) mit $(\mathcal{A}_1 \otimes \mathcal{A}_2) = \sigma(\{A_1 \times A_2 | A_1 \in \mathcal{A}_1 \text{ und } A_2 \in \mathcal{A}_2\})$ und $f_{\omega_2}(\omega_1) := f(\omega_2, \omega_1)$ für fixes ω_2 .

Ohne Beweis wird festgehalten:

Satz 16. Seien $X : (\Omega, \mathcal{A}) \rightarrow (\Omega_1, \mathcal{A}_1)$ und $Y : (\Omega, \mathcal{A}) \rightarrow (\Omega_2, \mathcal{A}_2)$ P -verteilte Zufallsvariablen, μ ein σ -endliches Mass auf \mathcal{A}_1 , g_Y eine Dichte von P_Y bezüglich eines σ -endliches Masses ν auf \mathcal{A}_2 . Eine $(\mathcal{A}_1 \otimes \mathcal{A}_2)$ -messbare Funktion $f : \Omega_1 \times \Omega_2 \rightarrow \mathbb{R}^+$ ist genau dann eine bedingte μ -Dichte von X bezüglich des W -Masses P unter Y wenn die durch

$$g(\omega_1, \omega_2) := f(\omega_1, \omega_2) g_Y(\omega_2)$$

definierte Funktion $g : \Omega_1 \times \Omega_2 \rightarrow \mathbb{R}^+$ eine $\mu \otimes \nu$ -Dichte der $(\mathcal{A}, \mathcal{A}_1 \otimes \mathcal{A}_2)$ -messbaren Abbildung $(X, Y) : (\Omega, \mathcal{A}) \rightarrow (\Omega_1 \times \Omega_2)$ ist.

Satz 17. Seien $X : (\Omega, \mathcal{A}) \rightarrow (\Omega_1, \mathcal{A}_1)$ und $Y : (\Omega, \mathcal{A}) \rightarrow (\Omega_2, \mathcal{A}_2)$ P -verteilte Zufallsvariablen, μ ein σ -endliches Mass auf \mathcal{A}_1 , ν ein σ -endliches Mass auf \mathcal{A}_2 und g eine $\mu \otimes \nu$ -Dichte von (X, Y) . Dann ist die durch

$$f(\omega_1, \omega_2) := \frac{g(\omega_1, \omega_2)}{\int g_{\omega_2}(\omega_1) d\mu}$$

mit $(\omega_1, \omega_2) \in \Omega_1 \times \Omega_2$ gegebenen Abbildung $f : \Omega_1 \times \Omega_2 \rightarrow \mathbb{R}^+$ eine bedingte μ -Dichte von X unter Y .

6.6 Eine suffiziente Statistik für Fähigkeiten im dichotomen, logistischen Zwei-Parameter-Modell

Definition 18. Sei $(\mathbb{R}^n, \mathfrak{B}^n)$ ein Stichprobenraum und $\mathcal{W} := \{P_\gamma | \gamma \in \Gamma\}, \Gamma \subset \mathbb{R}^n$ (bijektive Parametrisierung von \mathcal{W}) eine Menge von Wahrscheinlichkeits-Massen auf \mathfrak{B}^n und $X = (X_1, \dots, X_n)$ eine $(\Omega, \mathcal{A}) - (\mathbb{R}^n, \mathfrak{B}^n)$ -messbare Stichprobe (n -Vektor von Zufallsvariablen) mit $P_X \in \mathcal{W}$. Eine Statistik $T : (\mathbb{R}^n, \mathfrak{B}^n) \rightarrow (\mathbb{R}, \mathcal{B})$ heisst suffizient bezüglich \mathcal{W} genau dann, wenn es für alle $B \in \mathfrak{B}^n$ eine Abbildung g_B gibt, so dass $g_B \circ T(X) \in E_\gamma(1_B | T(X))$ für alle $\gamma \in \Gamma$ ($E_\gamma(1_B | T(X))$ ist der bedingte Erwartungswert von 1_B unter $T^{-1}(B)$ bezüglich der Wahrscheinlichkeitsverteilung P_γ , s. vorangehender Anhang).

Damit ist T suffizient bezüglich \mathcal{W} , wenn pro B eine Funktion g_B mit der angegebenen Eigenschaft für alle γ wählbar ist.

Satz 19. $g_B \circ T \in E(1_B|T)$ genau dann, wenn $\int_{A_1} g_B dP_{\gamma,T} = P_\gamma(T^{-1}(A) \cap B)$ für alle $A \in \mathcal{B}$.

Beweis. Für $g_B \circ T \in E(1_B|T)$ gilt: $\int_{T^{-1}(A)} g_B \circ T dP_\gamma = \int_{T^{-1}(A)} 1_B dP_\gamma = P_\gamma(T^{-1}(A) \cap B)$. \square

Satz 20. Für g_B gemäss Definition 18 gilt

$$g_B(t) = \frac{P_\gamma(\{X|T(X) = t\} \cap B)}{P_\gamma(\{X|T(X) = t\})}$$

$P_{\gamma,T}$ -fast überall.

Beweis. Mit

$$\begin{aligned} \int_{T^{-1}(\{t\})} g_B \circ T dP_\gamma &= \int_{\{t\}} g_B(t) dP_{\gamma,T} \\ &= g_B(t) P_{\gamma,T}(\{t\}) \\ &= g_B(t) P_\gamma(\{X|T(X) = t\}) \end{aligned}$$

erhält man mit

$$\begin{aligned} \int_{T^{-1}(\{t\})} g_B \circ T dP_\gamma &= P(T^{-1}(\{t\}) \cap B) \\ &= P(\{X|T(\mathbf{x}) = t\} \cap B) \end{aligned}$$

die zu beweisende Aussage

$$g_B(t) = \frac{P_\gamma(\{X|T(X) = t\} \cap B)}{P_\gamma(\{X|T(X) = t\})}.$$

\square

Der Satz zeigt den Zusammenhang mit Definitionen, die Suffizienz mittels Bedingten Wahrscheinlichkeitsverteilungen definieren (T ist suffizient bezüglich \mathcal{W} genau dann, wenn die bedingte Verteilung, die durch $P_\gamma(X \in B|T(x) = t)$ gegeben ist, unabhängig von γ wählbar ist - s. z.B. Müller (1991, S. 433 ff.)).

Satz 21. (*Faktorisierungskriterium*) Sei $X := (X_1, \dots, X_n)$ eine Stichprobe mit den i.i.d. Zufallsvariablen X_i und mit $X : (\Omega, \mathcal{A}) \rightarrow (\mathbb{R}^n, \mathcal{B}^n)$, und sei $T : (\mathbb{R}^n, \mathcal{B}^n) \rightarrow (\mathbb{R}, \mathcal{B})$ eine Statistik. T ist suffizient bezüglich der Menge der Wahrscheinlichkeitsmasse $\mathcal{W} := \{P_\gamma | \gamma \in \Gamma\}$ mit der Parametermenge Γ , wenn sich die Likelihood-Funktion L von X darstellen lässt als

$$L(X, \gamma) = g(T(X), \gamma) \cdot h(X),$$

so dass $g(T(X), \gamma)$ von γ und $T(X)$ abhängt, nicht aber explizit von X , und $h(X)$ eine von γ unabhängige Funktion der Stichprobe ist.

Beweis. Für fixes t gilt:

$$\begin{aligned} P_\gamma(X = \mathbf{x}|T(X) = t) &= \frac{P_\gamma(X = \mathbf{x}, T(X) = t)}{P_\gamma(T = t)} \\ &= \frac{g(t, \gamma) h(\mathbf{x})}{\sum_{T(\mathbf{x})=t} g(t, \gamma) h(\mathbf{x})} \\ &= \frac{h(\mathbf{x})}{\sum_{T(\mathbf{x})=t} h(\mathbf{x})} \end{aligned}$$

d.h. die bedingte Verteilung $P_\gamma(X = \mathbf{x}|T(X))$ ist von γ unabhängig wählbar. \square

Satz 22. Im logistischen Modell ist $\sum_{j=1}^n x_{ij} \alpha_j$ eine suffiziente Statistik für θ_i .

Der Beweis folgt Birnbaum (1968, S. 428 ff.).

Beweis. Es gilt

$$\begin{aligned} \mathbb{P}(\mathbf{X}_i = \mathbf{x}_i | \Theta_i = \theta_i, \mathbf{\Delta} = \mathbf{\delta}) &= \prod_{j=1}^n (p(\theta_i))^{x_{ij}} (1 - p(\theta_i))^{1-x_{ij}} \\ &= \prod_{j=1}^n \left(\frac{\exp(\alpha_j (\theta_i - \beta_j))}{1 + \exp(\alpha_j (\theta_i - \beta_j))} \right)^{x_{ij}} \left(1 - \frac{\exp(\alpha_j (\theta_i - \beta_j))}{1 + \exp(\alpha_j (\theta_i - \beta_j))} \right)^{1-x_{ij}} \\ &= \prod_{j=1}^n \left(\frac{\exp(\alpha_j (\theta_i - \beta_j))}{1 + \exp(\alpha_j (\theta_i - \beta_j))} \right)^{x_{ij}} \left(1 - \frac{\exp(\alpha_j (\theta_i - \beta_j))}{1 + \exp(\alpha_j (\theta_i - \beta_j))} \right) \\ &\quad \left(1 - \frac{\exp(\alpha_j (\theta_i - \beta_j))}{1 + \exp(\alpha_j (\theta_i - \beta_j))} \right)^{-x_{ij}}. \end{aligned}$$

Mit

$$\left(\frac{\exp(\alpha_j (\theta_i - \beta_j))}{1 + \exp(\alpha_j (\theta_i - \beta_j))} \right)^{x_{ij}} \left(1 - \frac{\exp(\alpha_j (\theta_i - \beta_j))}{1 + \exp(\alpha_j (\theta_i - \beta_j))} \right)^{-x_{ij}} = \frac{\left(\frac{\exp(\alpha_j (\theta_i - \beta_j))}{1 + \exp(\alpha_j (\theta_i - \beta_j))} \right)^{x_{ij}}}{\left(1 - \frac{\exp(\alpha_j (\theta_i - \beta_j))}{1 + \exp(\alpha_j (\theta_i - \beta_j))} \right)^{x_{ij}}}$$

und

$$\begin{aligned} \frac{\left(\frac{\exp(\alpha_j (\theta_i - \beta_j))}{1 + \exp(\alpha_j (\theta_i - \beta_j))} \right)^{x_{ij}}}{\left(1 - \frac{\exp(\alpha_j (\theta_i - \beta_j))}{1 + \exp(\alpha_j (\theta_i - \beta_j))} \right)^{x_{ij}}} &= \left(\frac{\frac{\exp(\alpha_j (\theta_i - \beta_j))}{1 + \exp(\alpha_j (\theta_i - \beta_j))}}{1 - \frac{\exp(\alpha_j (\theta_i - \beta_j))}{1 + \exp(\alpha_j (\theta_i - \beta_j))}} \right)^{x_{ij}} \\ &= (\exp(\alpha_j (\theta_i - \beta_j)))^{x_{ij}} \\ &= \exp(x_{ij} \alpha_j (\theta_i - \beta_j)) \end{aligned}$$

gilt

$$\prod_{j=1}^n \left(\frac{\exp(\alpha_j (\theta_i - \beta_j))}{1 + \exp(\alpha_j (\theta_i - \beta_j))} \right)^{x_{ij}} \left(1 - \frac{\exp(\alpha_j (\theta_i - \beta_j))}{1 + \exp(\alpha_j (\theta_i - \beta_j))} \right).$$

Damit ist

$$\begin{aligned} &\left(1 - \frac{\exp(\alpha_j (\theta_i - \beta_j))}{1 + \exp(\alpha_j (\theta_i - \beta_j))} \right)^{-x_{ij}} \\ &= \prod_{j=1}^n \left(1 - \frac{\exp(\alpha_j (\theta_i - \beta_j))}{1 + \exp(\alpha_j (\theta_i - \beta_j))} \right) \prod_{j=1}^n (\exp(x_{ij} \alpha_j (\theta_i - \beta_j))) \\ &= \prod_{j=1}^n \left(1 - \frac{\exp(\alpha_j (\theta_i - \beta_j))}{1 + \exp(\alpha_j (\theta_i - \beta_j))} \right) \exp\left(\sum_{j=1}^n x_{ij} \alpha_j (\theta_i - \beta_j)\right) \\ &= \prod_{j=1}^n \left(1 - \frac{\exp(\alpha_j (\theta_i - \beta_j))}{1 + \exp(\alpha_j (\theta_i - \beta_j))} \right) \exp\left(\theta_i \sum_{j=1}^n x_{ij} \alpha_j\right) \exp\left(-\sum_{j=1}^n x_{ij} \alpha_j \beta_j\right). \end{aligned}$$

Entsprechend ist $\sum_{j=1}^n x_{ij} \alpha_j$ gemäss Faktorisierungskriterium eine suffiziente Statistik für θ_i , die allerdings von den zu schätzenden Parametern α_j abhängig ist. Im Raschmodell wird von mit 1 identischen Trennschärfeparametern ausgegangen. Entsprechend ist in diesem Modell die Summe der Punkte des Antwortvektors eine suffiziente Statistik der Fähigkeiten, die von keinen Parametern abhängig ist. Dies führt zu spezifischen Schätzmethoden, der Bedingten Maximum Likelihood Schätzmethode (CMLE), s. Fischer und Molenaar (1995, S. 44) und Baker und Kim (2004, Kapitel 5: The Rasch Model). \square

Birnbaum (1968, S. 431 ff.) zeigt, dass die Existenz einer minimal suffizienten Statistik für θ_i , die als Werte die gewichteten Summen der binären Antworten eines Tests annimmt, ein logistisches Modell impliziert. Andersen (1972) zeigt, dass eine von den Itemparametern unabhängige

suffiziente Statistik ein logistisches Modell mit für alle Items identischen Trennschärfeparametern impliziert und dass deren Werte aus den ungewichteten Summen der Komponenten der Antwortvektoren bestehen.

6.7 EM-Algorithmus

Der EM-Algorithmus dient der iterativen Berechnung von Maximum-Likelihood-Schätzern, wenn man die Beobachtungen als unvollständige Daten anschaut (Dempster et al., 1977). Man geht von

1. zwei Stichprobenräumen \mathcal{Y} und \mathcal{H} aus, mit $\mathbf{y} \in \mathcal{Y}$ und $\mathbf{x} \in \mathcal{H}$, wobei \mathbf{y} die beobachteten Daten sind und \mathbf{x} die vollständigen, nicht beobachteten Daten sind.
2. einer Abbildung $h : \mathcal{H} \rightarrow \mathcal{Y}$
3. einer Familie von Stichprobendichten $f(\mathbf{x}|\phi)$ auf \mathcal{H} , die von den Parametern ϕ abhängen, samt den entsprechenden Dichten $g(\mathbf{y}|\phi)$ auf $\mathcal{H}(\mathbf{y})$ gegeben durch

$$g(\mathbf{y}|\phi) = \int_{\mathcal{H}(\mathbf{y})} f(\mathbf{x}|\phi) dx$$

mit $\mathcal{H}(\mathbf{y}) := \{\mathbf{x} \in \mathcal{H} \text{ mit } h(\mathbf{x}) = \mathbf{y}\}$

aus. Ziel ist es, das $g(\mathbf{y}|\phi)$ maximierende ϕ^* zu finden.

Dempster et al. (1977) formulieren den EM-Algorithmus u.a. für sogenannte natürliche Exponentialfamilien, die durch Dichtefunktionen der folgenden Form definiert sind

$$f(\mathbf{x}|\phi) = b(\mathbf{x}) a(\phi) \exp\left(\phi \mathbf{t}(\mathbf{x})^\top\right)$$

(s. für Exponentialfamilien samt Beispielen auch Agresti (1990, S. 80 f.)). Für solche Familien definieren sie („E“ für Erwartung und „M“ für Maximierung):

- E-Schritt: Man schätzt den Wert der suffizienten Statistik $\mathbf{t}(\mathbf{x})$ mittels Berechnung von $\mathbf{t}^p = \mathbb{E}(\mathbf{t}(\mathbf{x}) | \mathbf{y}, \phi^p)$
- M-Schritt: Man berechnet ϕ^{p+1} durch Lösung der Gleichung $\mathbb{E}(\mathbf{t}(\mathbf{x}) | \phi) = \mathbf{t}^p$

$\mathbb{E}(\mathbf{t}(\mathbf{x}) | \phi) = \mathbf{t}^p$ entspricht den Likelihood-Gleichungen bei Maximum-Likelihood-Schätzungen von Parametern bezüglich Daten, die gemäss einer regulären Exponentialfamilie verteilt sind (s. Agresti (1990, S. 113)). Bezüglich der Konvergenzeigenschaften des EM-Algorithmus bei natürlichen Exponentialfamilien sei auf Dempster et al. (1977) verwiesen, bezüglich anderer Verteilungsfamilien auf Wu (1983). Im Zusammenhang mit IRT-Modellen geht es um die Berechnung von ML-Schätzern für die Item-Parameter auf dem Hintergrund von unbeobachteten Werten θ gestützt auf beobachtete Antwortvektoren. Die Informationen fehlen entsprechend bezüglich den Fähigkeiten θ . Wären diese bekannt, könnte man auf Grund der Antwortvektoren gewöhnliche logistische Regressionen berechnen, um die Item-Parameter zu schätzen. Im Falle des Rasch-Modells (Trennschärfeparameter = 1) gibt es eine von der Trennschärfe unabhängige suffiziente Statistik für die Fähigkeiten (= Anzahl korrekter Antworten). Die ICCs des Raschmodells gehören denn auch zu den natürlichen Exponentialfamilien. In diesem Falle werden die Erwartungswerte dieser Statistiken - gegeben die beobachteten Daten und die vorläufigen Item-Parameter - im Maximierungsschritt des EM-Algorithmus verwendet: den entsprechenden Werten werden die Häufigkeiten deren Vorkommens sowie die Häufigkeiten korrekter Antworten in den Antwortvektoren an diesen Stellen pro Item zugeordnet, um dann logistische Regressionen zu berechnen (Maximierungsschritt). Im Falle des Zwei-Parameter-Modells fehlen einfache suffiziente Statistiken. Entsprechend werden von Bock und Aitkin statt der Werte der suffizienten Statistiken die Erwartungswerte, eine

bestimmte Fähigkeit zu haben, unter der Bedingung, einen bestimmten Antwortvektor aufzuweisen, verwendet. Auf Grund des Satzes von Bayes ist die bedingte Verteilung von θ unter der Bedingung $\mathbf{X}_i = \mathbf{x}_i$ gegeben durch

$$g(\theta|\mathbf{X}_i = \mathbf{x}_i) = \frac{P(\mathbf{X}_i = \mathbf{x}_i|\theta) g(\theta)}{P(\mathbf{X}_i = \mathbf{x}_i)}.$$

Entsprechend ist der bedingte Erwartungswerte unter der Bedingung $\mathbf{X}_i = \mathbf{x}_i$

$$\mathbb{E}(\theta|\mathbf{X}_i = \mathbf{x}_i) = \frac{\int \theta g(\theta) \prod_{j=1}^n p_j(\theta)^{x_{ij}} q_j(\theta)^{1-x_{ij}} d\theta}{\int g(\theta) \prod_{j=1}^n p_j(\theta)^{x_{ij}} q_j(\theta)^{1-x_{ij}} d\theta},$$

angenähert durch die Summe (Gauss-Hermite-Näherung, s. Anhang 6.4)

$$\mathbb{E}(\theta|\mathbf{X}_i = \mathbf{x}_i) \approx \frac{\sum_{k=1}^q Z_k g(Z_k) \prod_{j=1}^n p_j(Z_k)^{x_{ij}} q_j(Z_k)^{1-x_{ij}} w_k}{\sum_{k=1}^q g(Z_k) \prod_{j=1}^n p_j(Z_k)^{x_{ij}} q_j(Z_k)^{1-x_{ij}} w_k}.$$

Die Anzahl erwarteter Personen mit Fähigkeit Z_k ist entsprechend näherungsweise

$$f_{jk} \approx \frac{\sum_{i=1}^N Z_k g(Z_k) \prod_{j=1}^n p_j(Z_k)^{x_{ij}} q_j(Z_k)^{1-x_{ij}} w_k}{\sum_{k=1}^q g(Z_k) \prod_{j=1}^n p_j(Z_k)^{x_{ij}} q_j(Z_k)^{1-x_{ij}} w_k}.$$

Die erwartete Anzahl korrekter Antworten auf das Item j an der Stelle Z_k ist dann

$$r_{jk} \approx \frac{\sum_{i=1}^N x_{ij} Z_k g(Z_k) \prod_{j=1}^n p_j(Z_k)^{x_{ij}} q_j(Z_k)^{1-x_{ij}} w_k}{\sum_{k=1}^q g(Z_k) \prod_{j=1}^n p_j(Z_k)^{x_{ij}} q_j(Z_k)^{1-x_{ij}} w_k}$$

(Erwartungsschritt). Im Maximierungsschritt werden durch die Berechnung einer logistischen Regression optimale Parameter bezüglich $(r_{jk}, f_k - r_{jk})^\top$ an den Stellen Z_k für jedes Item j berechnet (s. auch Baker und Kim (2004, S. 169, Some Background on the EM Algorithm))

6.8 Herleitung der Likelihood-Gleichungen für die MMLE-Methode bei dichotomen, logistischen Zweiparameter-Modellen

Die Herleitung folgt Baker und Kim (2004, S. 160, ff.), wobei dort die entsprechenden Herleitungen für das Drei-Parameter-Modell von Birnbaum durchgeführt werden.

$\boldsymbol{\delta}_j = (\alpha_j, \beta_j)^\top$ sind die Item-Parameter der n Items; θ_i die Fähigkeitsparameter der N Personen, $x_{ij} \in \{0, 1\}$ die Antwort der i -ten Person auf das j -te Item, $\mathbf{x}_i = (x_{i1}, \dots, x_{in})^\top$ der Antwortvektor der i -ten Person auf die n Items. $P(\mathbf{x}_i|\theta_i, \boldsymbol{\delta}) := \mathbb{P}(\mathbf{X}_i = \mathbf{x}_i|\Theta_i = \theta_i, \boldsymbol{\Delta} = \boldsymbol{\delta})$. g sei die Dichtefunktion der *i.i.d.*-verteilten Zufallsvariablen, welche Fähigkeiten als Werte annehmen. $a_j := \alpha_j$ und $b_j := -\alpha_j \beta_j$.

Sei

$$P(\mathbf{x}_i) := \int P(\mathbf{x}_i|\theta_i, \boldsymbol{\delta}) g(\theta_i) d\theta_i$$

die Wahrscheinlichkeit, dass die i -te Person bei den Itemparametern $\boldsymbol{\delta}$ auf die n Items den Antwortvektor \mathbf{x}_i liefert.

Dann ist bei fixen \mathbf{x}_i

$$L := \prod_{i=1}^N P(\mathbf{x}_i)$$

die Likelihood-Funktion von $\boldsymbol{\delta}$ und logarithmiert ergibt sich

$$\ln L = \sum_{i=1}^N \ln P(\mathbf{x}_i).$$

Um die Likelihood-Gleichung für a_j zu finden, setzt man

$$\frac{\partial \ln L}{\partial a_j} = 0.$$

Damit gilt dann

$$\begin{aligned} \frac{\partial \ln L}{\partial a_j} &= \sum_{i=1}^N \frac{\partial}{\partial a_j} \ln P(\mathbf{x}_i) \\ &= \sum_{i=1}^N \frac{1}{P(\mathbf{x}_i)} \frac{\partial}{\partial a_j} \left(\int P(\mathbf{x}_i|\theta_i, \boldsymbol{\delta}) g(\theta_i) d\theta_i \right). \end{aligned}$$

Aus Gründen der Übersichtlichkeit lässt man i in θ_i fallen - die θ_i können als Ergebnis einer Zufallsstichprobe gesehen werden kann. Durch Vertauschen von Integral und Ableitung - die erlaubt für Funktionen f , wenn es eine messbare Funktion F gibt, die über Mengen aus der Borelschen σ -Algebra \mathcal{B} integrierbar ist, so dass $\left| \frac{d}{da_j} f(a_j) \right| < F$ und f stetig ableitbar ist - ergibt sich

$$\frac{\partial \ln L}{\partial a_j} = \sum_{i=1}^N \frac{1}{P(\mathbf{x}_i)} \int \frac{\partial}{\partial a_j} P(\mathbf{x}_i|\theta, \boldsymbol{\delta}) g(\theta) d\theta.$$

Da wegen $\frac{d}{dx} \ln f(x) = \frac{1}{f(x)} \frac{d}{dx} f(x)$, gilt $f(x) \cdot \frac{d}{dx} \ln f(x) = \frac{d}{dx} f(x)$ und im vorliegenden Fall

$$\frac{\partial}{\partial a_j} P(\mathbf{x}_i|\theta, \boldsymbol{\delta}) = P(\mathbf{x}_i|\theta, \boldsymbol{\delta}) \frac{\partial}{\partial a_j} \ln P(\mathbf{x}_i|\theta, \boldsymbol{\delta}).$$

Damit erhält man

$$\begin{aligned} \frac{\partial \ln L}{\partial a_j} &= \sum_{i=1}^N \frac{1}{P(\mathbf{x}_i)} \int P(\mathbf{x}_i|\theta, \boldsymbol{\delta}) g(\theta) \frac{\partial}{\partial a_j} \ln P(\mathbf{x}_i|\theta, \boldsymbol{\delta}) d\theta \\ &= \sum_{i=1}^N \int \frac{P(\mathbf{x}_i|\theta, \boldsymbol{\delta}) g(\theta)}{P(\mathbf{x}_i)} \frac{\partial}{\partial a_j} \ln P(\mathbf{x}_i|\theta, \boldsymbol{\delta}) d\theta. \end{aligned}$$

Mit der Bayesschen Formel

$$P(\theta|\mathbf{x}_i, \boldsymbol{\delta}) = \frac{P(\mathbf{x}_i|\theta, \boldsymbol{\delta}) g(\theta)}{\int P(\mathbf{x}_i|\theta, \boldsymbol{\delta}) g(\theta) d\theta} = \frac{P(\mathbf{x}_i|\theta, \boldsymbol{\delta}) g(\theta)}{P(\mathbf{x}_i)}$$

gilt entsprechend

$$\frac{\partial \ln L}{\partial a_j} = \sum_{i=1}^N \int P(\theta|\mathbf{x}_i, \boldsymbol{\delta}) \frac{\partial}{\partial a_j} \ln P(\mathbf{x}_i|\theta, \boldsymbol{\delta}) d\theta.$$

Da mit $q_j(\theta) = 1 - p_j(\theta)$ und für $p_j(\theta) = \frac{\exp(-a(\theta-b))}{1+\exp(-a(\theta-b))}$

$$P(\mathbf{x}_i|\theta, \boldsymbol{\delta}) = \prod_{j=1}^n p_j(\theta)^{x_{ij}} q_j(\theta)^{1-x_{ij}}$$

gilt, erhält man (mit Kettenregel)

$$\frac{\partial \ln L}{\partial a_j} = \sum_{i=1}^N \int P(\theta|\mathbf{x}_i, \boldsymbol{\delta}) \frac{\partial}{\partial a_j} \ln \left(\prod_{j=1}^n p_j(\theta)^{x_{ij}} q_j(\theta)^{1-x_{ij}} \right) d\theta \quad (23)$$

$$= \sum_{i=1}^N \int \frac{P(\theta|\mathbf{x}_i, \boldsymbol{\delta}) \frac{\partial}{\partial a_j} \prod_{j=1}^n p_j(\theta)^{x_{ij}} q_j(\theta)^{1-x_{ij}}}{\prod_{j=1}^n p_j(\theta)^{x_{ij}} q_j(\theta)^{1-x_{ij}}} d\theta. \quad (24)$$

Nun gilt

$$\frac{\partial}{\partial a_j} \prod_{j=1}^n p_j(\theta)^{x_{ij}} q_j(\theta)^{1-x_{ij}} = \prod_{h \neq j}^n p_h(\theta)^{x_{ih}} q_h(\theta)^{1-x_{ih}} \cdot \frac{\partial}{\partial a_j} p_j(\theta)^{x_{ij}} q_j(\theta)^{1-x_{ij}} \quad (25)$$

und (mit Produktregel und anschliessend Kettenregel)

$$\begin{aligned} \frac{\partial}{\partial a_j} p_j(\theta)^{x_{ij}} q_j(\theta)^{1-x_{ij}} &= q_j(\theta)^{1-x_{ij}} \frac{\partial}{\partial a_j} p_j(\theta)^{x_{ij}} + p_j(\theta)^{x_{ij}} \frac{\partial}{\partial a_j} q_j(\theta)^{1-x_{ij}} \\ &= q_j(\theta)^{1-x_{ij}} x_{ij} p_j(\theta)^{x_{ij}-1} \frac{\partial}{\partial a_j} p_j(\theta) + \\ &\quad p_j(\theta)^{x_{ij}} (1-x_{ij}) q_j(\theta)^{1-x_{ij}-1} \frac{\partial}{\partial a_j} q_j(\theta). \end{aligned} \quad (26)$$

Es ist (Quotienten- und Kettenregel)

$$\begin{aligned} \frac{\partial}{\partial a_j} p_j(\theta) &= \frac{\partial}{\partial a_j} \frac{e^{a_j \theta + b_j}}{1 + e^{a_j \theta + b_j}} \\ &= \frac{e^{a_j \theta + b_j} \theta (1 + e^{a_j \theta + b_j}) - e^{a_j \theta + b_j} (e^{a_j \theta + b_j} \theta)}{(1 + e^{a_j \theta + b_j})^2} \\ &= \frac{\theta (e^{a_j \theta + b_j} (1 + e^{a_j \theta + b_j}) - e^{a_j \theta + b_j} e^{a_j \theta + b_j})}{(1 + e^{a_j \theta + b_j})^2} \\ &= \frac{\theta (e^{a_j \theta + b_j} + e^{a_j \theta + b_j} e^{a_j \theta + b_j} - e^{a_j \theta + b_j} e^{a_j \theta + b_j})}{(1 + e^{a_j \theta + b_j})^2} \\ &= \frac{\theta e^{a_j \theta + b_j}}{(1 + e^{a_j \theta + b_j})^2} = \theta \frac{e^{a_j \theta + b_j}}{1 + e^{a_j \theta + b_j}} \frac{1}{1 + e^{a_j \theta + b_j}} \\ &= p_j(\theta) q_j(\theta) \theta =: K, \end{aligned}$$

denn $q_j(\theta) := 1 - p_j(\theta) = 1 - \frac{e^{a_j \theta + b_j}}{1 + e^{a_j \theta + b_j}} = \frac{1 + e^{a_j \theta + b_j} - e^{a_j \theta + b_j}}{1 + e^{a_j \theta + b_j}} = \frac{1}{1 + e^{a_j \theta + b_j}}$.

Analog gilt: $\frac{\partial}{\partial a_j} q_j(\theta) = \frac{\partial}{\partial a_j} (1 - p_j(\theta)) = -K$.

Damit ist mit (25), (26) und

$$H := \prod_{h \neq j}^n p_h(\theta)^{x_{ih}} q_h(\theta)^{1-x_{ih}}$$

$$\begin{aligned} \frac{\partial}{\partial a_j} \prod_{j=1}^n p_j(\theta)^{x_{ij}} q_j(\theta)^{1-x_{ij}} &= H \left[q_j(\theta)^{1-x_{ij}} \frac{\partial}{\partial a_j} p_j(\theta)^{x_{ij}} + p_j(\theta)^{x_{ij}} \frac{\partial}{\partial a_j} q_j(\theta)^{1-x_{ij}} \right] \\ &= H \left[q_j(\theta)^{1-x_{ij}} x_{ij} p_j(\theta)^{x_{ij}-1} K + p_j(\theta)^{x_{ij}} (1-x_{ij}) q_j(\theta)^{1-x_{ij}-1} (-K) \right] \\ &= KH \cdot \left[q_j(\theta)^{1-x_{ij}} x_{ij} p_j(\theta)^{x_{ij}-1} - p_j(\theta)^{x_{ij}} (1-x_{ij}) q_j(\theta)^{1-x_{ij}-1} \right]. \end{aligned}$$

Es gilt

$$\left[q_j(\theta)^{1-x_{ij}} x_{ij} p_j(\theta)^{x_{ij}-1} - p_j(\theta)^{x_{ij}} (1-x_{ij}) q_j(\theta)^{1-x_{ij}-1} \right] = \begin{cases} 1 & \text{für } x_{ij} = 1 \\ -1 & \text{für } x_{ij} = 0 \end{cases},$$

denn für $x_{ij} = 1$ ist $q_j(\theta)^{1-1} \cdot 1 \cdot p_j(\theta)^{1-1} - p_j(\theta)^1 (1-1) q_j(\theta)^{1-1-1} = 1 - 0 = 1$,
und für $x_{ij} = 0$ ist $q_j(\theta)^{1-0} \cdot 0 \cdot p_j(\theta)^{0-1} - p_j(\theta)^0 (1-0) q_j(\theta)^{1-0-1} = 0 - 1 = -1$.

Damit gilt mit den Definitionen von H und K :

$$\begin{aligned} \frac{\partial}{\partial a_j} \prod_{j=1}^n p_j(\theta)^{x_{ij}} q_j(\theta)^{1-x_{ij}} &= (-1)^{x_{ij}+1} H K \\ &= (-1)^{x_{ij}+1} \prod_{h \neq j}^n p_h(\theta)^{x_{ih}} q_h(\theta)^{1-x_{ih}} p_j(\theta) q_j(\theta) \theta \end{aligned}$$

und man erhält mit (24)

$$\begin{aligned}
\frac{\partial \ln L}{\partial a_j} &= \sum_{i=1}^N \int \frac{P(\theta|\mathbf{x}_i, \boldsymbol{\delta}) \frac{\partial}{\partial a_j} \prod_{j=1}^n p_j(\theta)^{x_{ij}} q_j(\theta)^{1-x_{ij}}}{\prod_{j=1}^n p_j(\theta)^{x_{ij}} q_j(\theta)^{1-x_{ij}}} d\theta \\
&= \sum_{i=1}^N (-1)^{x_{ij}+1} \int \frac{P(\theta|x_i, \boldsymbol{\delta}) \prod_{h \neq j} p_h(\theta)^{x_{ih}} q_h(\theta)^{1-x_{ih}} \cdot p_j(\theta) q_j(\theta) \theta}{\prod_{j=1}^n p_j(\theta)^{x_{ij}} q_j(\theta)^{1-x_{ij}}} d\theta \\
&= \sum_{i=1}^N (-1)^{x_{ij}+1} \int \frac{P(\theta|x_i, \boldsymbol{\delta}) \cdot p_j(\theta) q_j(\theta) \theta}{p_j(\theta)^{x_{ij}} q_j(\theta)^{1-x_{ij}}} d\theta.
\end{aligned}$$

Dies ist die erste Ableitung nach a_j , also die Likelihood-Gleichung für a_j .
Diese Gleichung kann umformuliert werden. Denn es gilt

$$(-1)^{x_{ij}+1} \frac{1}{p_j(\theta)^{x_{ij}} q_j(\theta)^{1-x_{ij}}} = \frac{x_{ij} - p_j(\theta)}{p_j(\theta) q_j(\theta)}$$

da

$$(-1)^{x_{ij}+1} = \begin{cases} -1 & \text{für } x_{ij} = 0 \\ 1 & \text{für } x_{ij} = 1 \end{cases}.$$

Damit gilt für $x_{ij} = 1$

$$(-1)^{x_{ij}+1} \frac{1}{p_j(\theta)^{x_{ij}} q_j(\theta)^{1-x_{ij}}} = \frac{1}{p_j(\theta)} = \frac{1 - p_j(\theta)}{p_j(\theta)(1 - p_j(\theta))} = \frac{1 - p_j(\theta)}{p_j(\theta) q_j(\theta)} = \frac{x_{ij} - p_j(\theta)}{p_j(\theta) q_j(\theta)}$$

und für $x_{ij} = 0$

$$(-1)^{x_{ij}+1} \frac{1}{p_j(\theta)^{x_{ij}} q_j(\theta)^{1-x_{ij}}} = -\frac{1}{q_j(\theta)} = \frac{0 - 1}{q_j(\theta)} = \frac{p_j(\theta)(0 - 1)}{p_j(\theta) q_j(\theta)} = \frac{0 - p_j(\theta)}{p_j(\theta) q_j(\theta)} = \frac{x_{ij} - p_j(\theta)}{p_j(\theta) q_j(\theta)}.$$

Entsprechend gilt für die Likelihood-Gleichung für a_j

$$\begin{aligned}
\frac{\partial \ln L}{\partial a_j} &= \sum_{i=1}^N \int (-1)^{x_{ij}+1} \frac{P(\theta|\mathbf{x}_i, \boldsymbol{\delta}) \cdot p_j(\theta) q_j(\theta) \theta}{p_j(\theta)^{x_{ij}} q_j(\theta)^{1-x_{ij}}} d\theta \\
&= \sum_{i=1}^N \int \frac{(x_{ij} - p_j(\theta)) P(\theta|\mathbf{x}_i, \boldsymbol{\delta}) \cdot p_j(\theta) q_j(\theta) \theta}{p_j(\theta) q_j(\theta)} d\theta \\
&= \sum_{i=1}^N \int (x_{ij} - p_j(\theta)) P(\theta|x_i, \boldsymbol{\delta}) \theta d\theta = 0.
\end{aligned}$$

Für b_j erhält man man analog

$$\frac{\partial \ln L}{\partial b_j} = \sum_{i=1}^N \int (x_{ij} - p_j(\theta)) P(\theta|\mathbf{x}_i, \boldsymbol{\delta}) d\theta = 0. \quad (27)$$

Für das Newton-Raphson-Nährungsverfahren müssten noch die zweiten Ableitungen berechnet werden.

6.9 Regressionsbeispiel

Mit dem irt-Befehl `SchaffDatDicho()` wird ein standardnormalverteilter Fähigkeits-Datensatz $\boldsymbol{\theta}$ samt der dazugehörigen Antwortematrix \mathbf{x} geschaffen. Es werden die Werte einer zweiten Variable Y berechnet, so dass der Zusammenhang zwischen den Variablen $\boldsymbol{\theta}$ und Y mittels linearer Regression geschätzt werden kann und die Residuen normalverteilt sind. Es wird die Regressionsgerade der Variable Y auf $\boldsymbol{\theta}$ geschätzt. Darauf hin werden die Fähigkeiten zurückgeschätzt - mittels $\mathbf{x}\hat{\boldsymbol{\alpha}}$, wobei $\hat{\boldsymbol{\alpha}}$ die geschätzten Trennschärfeparameter sind. Die geschätzten Fähigkeiten werden standardisiert

und es wird die Regression der Variable Y auf die geschätzten Fähigkeiten $\hat{\theta}$ berechnet. Zudem wird die Korrelation zwischen den ursprünglichen Fähigkeiten und den geschätzten Fähigkeiten berechnet.

```
X=SchaffDatDicho();X1=X[[1]][,2:6];b=c(0,0,0,0,0);a=c(1,1,1,1,1);betas=cbind(b,a)
para=RizopoulosEM(X1,betas)
echTet=X[[1]][,1]
y=echTet*1.5+rnorm(1500,0,1)
ur=lm(y~echTet);ur
Dies ergibt die Koeffizienten (s. Tabelle 18):
```

(Intercept)	fTet
-0.04807	1.51849

Tabelle 18: Ordinatenabschnitt und Steigungskoeffizient der geschätzten Regressionsgleichung von Y auf θ .

```
plot(echTet,y)
abline(ur)
Dies ergibt die Abbildung 16.
```

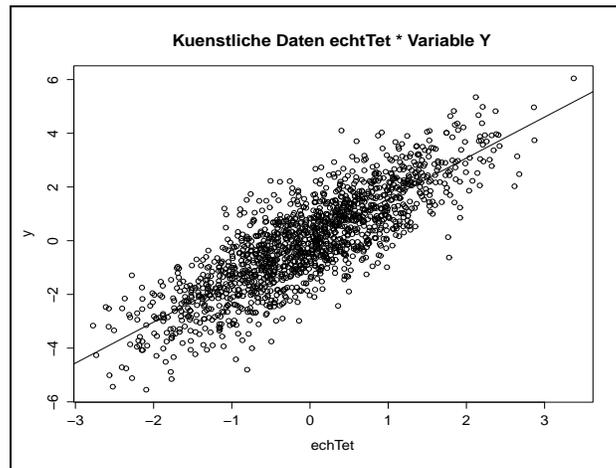


Abbildung 16: Streudiagramm normalverteilter künstlicher Daten θ (= echtTet) und der Variable Y mit geschätzter Regressionsgerade und normalverteilten Residuen; für Koeffizienten s. Tabelle 18

Berechnung der standardisierten geschätzten Fähigkeiten und der Regression von Y auf $\hat{\theta}$:

```
gTet=X[[1]][,2:6]*%para[,2]
gTet=(gTet-mean(gTet))/sd(gTet)
sch=lm(y~gTet);sch
Dies ergibt die Koeffizienten (s. Tabelle 19):
```

(Intercept)	fTet
-0.03672	1.12895

Tabelle 19: Ordinatenabschnitt und Steigungskoeffizient der Regressionsgleichung von Y auf $\hat{\theta}$.

```
plot(fTet,y)
abline(sch)
ergibt die Abbildung 17:
```

Überprüfung auf Normalverteiltheit der Residuen der Regression der Variable Y auf $\hat{\theta}$ ergibt mit den folgenden Befehlen die Abbildung 18.

```
qqnorm(sch$residuals,
main="QQ-Plot auf Normalverteilung",
xlab="Theoretische Quantile",
ylab="Empirische Quantile")
qqline(sch$residuals)
```

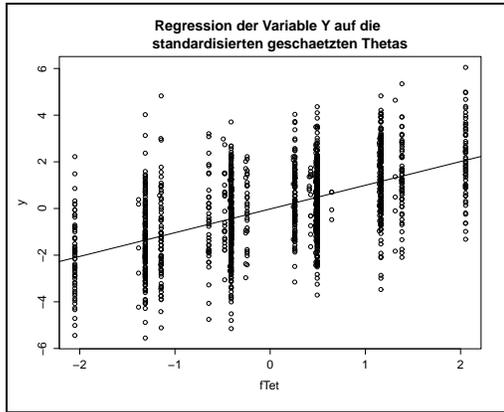


Abbildung 17: Zusammenhang zwischen standardisiertem geschätztem $\hat{\theta}$ und der Variable Y , für die Koeffizienten s. Tabelle 19

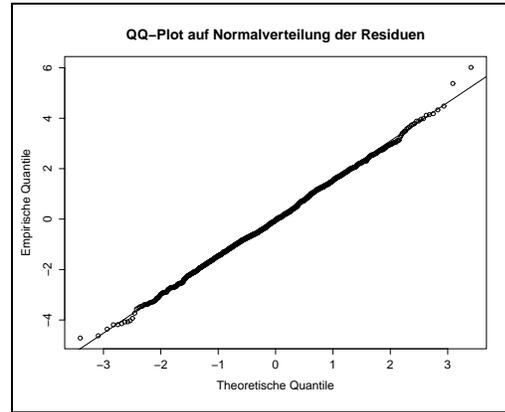


Abbildung 18: QQ-Plot auf Normalverteilung der Residuen (Regression der Variable Y auf $\hat{\theta}$)

Die Pearson-Korrelation zwischen dem ursprünglichen θ und dem geschätzten $\hat{\theta}$ beträgt 0.681 (mit `cor(echTet, fTet)`). Es wird ersichtlich, dass so wenige Items θ nicht genügend genau zurückschätzen können. Es ergeben sich Klumpungen von Daten, die in den ursprünglichen Daten nicht vorkommen. Entsprechend kann man nicht davon ausgehen, dass Distanzverhältnisse reproduziert werden und es ist fragwürdig, dass hier eine Intervallskala erzeugt wird. Andererseits ist festzustellen, dass die Daten im Vergleich zur einfachen Summenbildung über die Antwortvektoren differenzierter werden.

Analoge Resultate - allerdings mit einer höheren Korrelation von 0.8 von θ und $\hat{\theta}$ - werden erreicht, wenn man mit Daten, die statt mit den Trennschärfeparametern (1,1,1,1,1) mit (1,2,3,2,0.5) produziert wurden, arbeitet. Um ein entsprechendes Experiment zu reproduzieren genügt es mit `paraMatrix=cbind(c(-2,-1,0,1,2),c(1,2,3,2,0.5));X=SchaffDatDicho(paraMatrix)` zu starten und dann die obigen Befehle auszuführen. Die Resultate sind also nicht davon abhängig, dass man implizit ein Rasch-Modell mit den spezifischen suffizienten Statistiken verwendet (natürliche Exponentialfamilie).

Arbeitet man mit 20 statt mit 5 Items ergeben sich die folgenden Graphiken 19 und 20:

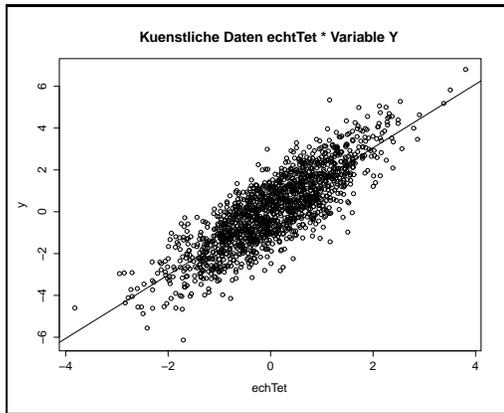


Abbildung 19: Zusammenhang von Y und faktischem Fähigkeitsvektor θ .

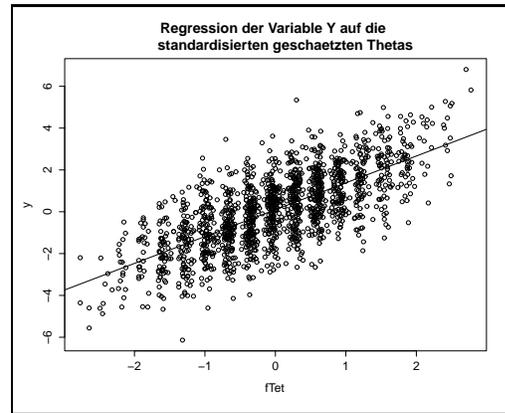


Abbildung 20: Zusammenhang von Y und geschätztem Fähigkeitsvektor $\hat{\theta}$ bei 20 Items

Die Korrelation zwischen den ursprünglichen Daten und dem geschätzten $\hat{\theta}$ beläuft sich auf 0.8520966, steigt also im Vergleich zur Schätzung mit 5 Items. Die Steigung der Regressionsgerade ist beim Zusammenhang zwischen geschätztem $\hat{\theta}$ und der Variable Y auch hier kleiner als bei der Regression von Y auf das ursprüngliche θ . Es liegt aber immer noch Klumpung vor. Diese würde mit genügend vielen Items offenbar verschwinden, wobei in der Praxis auch 20 Items unrealistisch sind.

6.10 R-Befehle

6.10.1 R-Befehle für Abbildungen 3 bis 6, Seite 18

```
a=MCeval(BA=T)
Item 1, Schwierigkeit:
qqnorm(a[1,seq(1,400,by=2)]); qqline(a[1,seq(1,400,by=2)])
Item 1, Trennschärfe:
qqnorm(a[1,seq(2,400,by=2)]); qqline(a[1,seq(2,400,by=2)])
Item 3, Schwierigkeit:
qqnorm(a[3,seq(1,400,by=2)]); qqline(a[3,seq(1,400,by=2)])
Item 3, Trennschärfe:
qqnorm(a[3,seq(2,400,by=2)]); qqline(a[3,seq(2,400,by=2)]).
```

6.10.2 R-Befehle für Tabellen 11, Seite 26

```
a=MCeval(ED=T,Intervall=c(-1.8,1.8));evalMCeval(a)
a=MCeval(ED=T,Intervall=c(-2,2));evalMCeval(a)
a=MCeval(ED=T,Intervall=c(-3,3));evalMCeval(a)
a=MCeval(ED=T,Intervall=c(-5,5));evalMCeval(a)
a=MCeval(ED=T,Intervall=c(-10,10));evalMCeval(a)
a=MCeval(ED=T,Intervall=c(-20,20));evalMCeval(a).
```

6.10.3 R-Befehle für die Abbildungen 7 bis 10, Seite 27

```
X=SchaffDatDicho();X1=X[[1]][,2:6]
bei=Empiric.dist.EM(X1,betas,Intervall=c(-2,2),empVert=T,Knoten=15,runden=50)
normdiskret(bei$EmpirischeVerteilung[,1])
points(bei$EmpirischeVerteilung[,1],bei$EmpirischeVerteilung[,2],pch=2)
```

```

bei=Empiric_dist_EM(X1,betas,Intervall=c(-5,5),empVert=T,Knoten=15,runden=50)
normdiskret(bei$EmpirischeVerteilung[,1])
points(bei$EmpirischeVerteilung[,1],bei$EmpirischeVerteilung[,2],pch=2)

bei=Empiric_dist_EM(X1,betas,Intervall=c(-5,5),empVert=T,Knoten=50,runden=50)
normdiskret(bei$EmpirischeVerteilung[,1])
points(bei$EmpirischeVerteilung[,1],bei$EmpirischeVerteilung[,2],pch=2)

bei=Empiric_dist_EM(X1,betas,Intervall=c(-20,20),empVert=T,Knoten=50,runden=50)
normdiskret(bei$EmpirischeVerteilung[,1])
points(bei$EmpirischeVerteilung[,1],bei$EmpirischeVerteilung[,2],pch=2)

legend("bottom",legend=c("\normalverteilt' diskret", "geschaetzte relative
Haeufigkeiten"),pch=c(1,2),bty="n").

```

6.10.4 R-Befehle für die Abbildung 11, Seite 30.

```

X=SchaffDatDicho(distr = 2,df=5);
X1=X[[1]][,2:6]; b=c(0,0,0,0,0); a=c(1,1,1,1,1); betas=cbind(b,a)
plot(seq(-5,10,15/(50-1)), gewChiVert(kpunkte=50,df=5)[,2], ylim=c(-0.04,0.10))
bei=Empiric_dist_EM(X1,betas, Knoten=50, empVert=T,Intervall=c(-10,10))
points(bei[[2]][,1],bei[[2]][,2],pch=3)
bei=Empiric_dist_EM(X1,betas, Knoten=50, runden=50, Protokoll=F, empVert=T,dfchi=5)
points(bei[[2]][,1],bei[[2]][,2],pch=20)

```

6.10.5 R-Befehle für die Tabelle 15, Seite 31

```

a=MCeval(SuS=T); evalMCeval(a)
a=MCeval(SuS=T,paraMatrix=cbind(c(-2,-1,0,1,2),c(1,2,3,2,0.5))); evalMCeval(a)
a=MCeval(SuS=T,distr=1,unif=c(-10,10)); evalMCeval(a)
a=MCeval(SuS=T,distr=1,unif=c(-4,4)); evalMCeval(a)
a=MCeval(SuS=T,distr=2,df=5); evalMCeval(a)
a=MCeval(SuS=T,distr=2,df=2); evalMCeval(a).

```

6.10.6 R-Befehle für Abbildungen 12 bis 15, Seite 33

```

X=SchaffDatDicho();X1=X[[1]][,2:6];trenn=c(1,1,1,1,1);schw=c(0,0,0,0,0)
betas=cbind(schw,trenn)
a=SufstatEM(X1, betas, empVert =T )
dichte(a$EmpirischeVerteilung,6,hoehe=0.45)
weich=smooth.spline(seq(-3,3,0.1), dnorm(seq(-3,3,0.1)), spar=0.5)
lines(weich,lty=2)
points(a$EmpirischeVerteilung[,2],a$EmpirischeVerteilung[,4])

X=SchaffDatDicho(n=8)
X1=X[[1]][,2:9];trenn=c(1,1,1,1,1,1,1,1);schw=c(0,0,0,0,0,0,0,0)
betas=cbind(schw,trenn)
a=SufstatEM(X1, betas, empVert =T )
dichte(a$EmpirischeVerteilung,AnzInt=9,hoehe=0.45,tiefe=0.1)
weich=smooth.spline(seq(-3,3,0.1), dnorm(seq(-3,3,0.1)), spar=0.5)
lines(weich,lty=2)
points(a$EmpirischeVerteilung[,2],a$EmpirischeVerteilung[,4])

X=SchaffDatDicho(distr = 1, unif=c(-5,5), n=8)

```

```

X1=X[[1]][,2:9];trenn=c(1,1,1,1,1,1,1,1,1);schw=c(0,0,0,0,0,0,0,0,0)
betas=cbind(schw,trenn)
a=SufstatEM(X1, betas, empVert =T )
dichte(a$Empirische_Verteilung,AnzInt=9, tiefe=0.06)
h=5/sqrt(10^2/12);h
lines(c(-h,h),c(1/(2*h),1/(2*h)),lty=2)
points(a$Empirische_Verteilung[,2],a$Empirische_Verteilung[,4])

X=SchaffDatDicho(distr = 2, df = 5,n=8)
X1=X[[1]][,2:9];trenn=c(1,1,1,1,1,1,1,1,1);schw=c(0,0,0,0,0,0,0,0,0)
betas=cbind(schw,trenn)
a=SufstatEM(X1, betas, empVert =T )
a$Empirische_Verteilung[,2]=(a$Empirische_Verteilung[,2]+5/sqrt(10))*sqrt(10)
dichte(a$Empirische_Verteilung,AnzInt=8,hoehe=0.45,tiefe=0.2)
dat=seq(0,15,0.1);chi=dchisq(dat,5);
weich=smooth.spline(dat,chi, spar=0.5)
lines(weich,lty=2)
points(a$Empirische_Verteilung[,2],a$Empirische_Verteilung[,4])

```

6.11 R-Paket IRT

IRT - Implementierung einiger Schätzmethoden fuer logistische Zweiparameter-Modelle und dichotome Antwortvariablen

Description

Es werden verschiedene Item-Parameter-Schätzmethoden (Item Response Theory) fuer den Fall dichotomer Antwortvariablen und logistischer Zweiparameter-Modelle implementiert, und zwar gemaess der sogenannte Marginal Maximum Likelihood-Schätzung (MMLE). Eine erste Variante wurde von Bock und Liebermann entwickelt und es wurde das Newton-Raphson-Naehungsverfahren verwendet (hier nicht implementiert). Spaeter lieferten Bock und Aitkin fuer die MMLE einen EM-Algorithmus - hier implementiert (s. Befehl `BockAitkin`). Die Methode setzt standardnormalverteilte Faehigkeitsparameter voraus und verwendet fuer die Integration die Gauss-Hermite-Integrations-Naeherung (GH-Integration). Weiterhin ist der EM-Algorithmus fuer ebenfalls standardnormalverteilte Faehigkeiten von Rizopoulos implementiert (Befehl `RizopoulosEM`, ebenfalls mit GH-Integration). Als dritte Methode wird ein Vorschlag von Bock und Aitkin implementiert, welche in der EM- Iteration die Wahrscheinlichkeitsfunktion der Faehigkeiten auf fixe Knotenpunkten mitschaetzt und diese Schätzungen als Gewichte in der Integration verwendet (Befehl: `Empiric_dist_EM`). Zuletzt wird ein Verfahren implementiert, das relative Haeufigkeiten auf den standardisierten Werten der suffizienten Statistiken $\sum(a_{jxij})$ (Summe der mit den Trennschaerfeparametern gewichtete Antworten pro Person; Befehl `SufstatEM`) verwendet. Fuer die letzten zwei Methoden wird der EM-Algorithmus von Rizopoulos verwendet. Fuer naehere Beschreibungen siehe die entsprechenden Funktionen

Details

Package: irt Type: Package Version: 1.0 Date: 2014-11-22 License: GNU General Public License Version 2, June 1991

Author(s)

Paul Ruppen; Maintainer: paul.ruppen@bluewin.ch

References

Birnbaum, A. (1968). Some Latent Trait Models and their Use in Inferring an Examinee's Ability. In F. Lord & M. Novick (Hrsg.), *Statistical Theories of Mental Test Scores* (S. 397-479). Reading: Addison-Wesley.

Bock, R. & Aitkin, M. (1981). Marginal Maximum Likelihood Estimation of Item Parameters: Application of an EM Algorithm. *Psychometrika*, 46, 443-459. van der Linden, W. J. & Hambleton, R. K. (Hrsg.). (1997). *Handbook of Modern Item Response Theory*. New York: Springer. Baker, F. B. & Kim, S.-H. (2004). *Item Response Theory: Parameter Estimation Techniques*. Basel: Dekker. Rizopoulos, D. (2006, November). ltm: An R Package for Latent Variable Modeling and Item Response Theory Analysis. *Journal of Statistical Software*, 17, (5).

Examples

s. Befehle `BockAitkin`, `RizopoulosEM`, `Empiric_dist_EM`, `SufstatEM` im Index [Package irt version 1.0 Index]
`BockAitkin {irt}` R Documentation

Item-Parameter-Schaetzung mit der Bock-Aitkin-Methode

Description

Es werden die Item-Parameter des logistischen Zweiparameter-Modells von Birnbaum mit dem EM-Algorithmus von Bock-Aitkin (1981) geschätzt.

Usage

BockAitkin(X, startbetas, runden = 50, Knoten = 15, Protokoll = F)

Arguments

X NXn-Matrix mit Antwortvektoren pro Person als Zeilen (N = Anzahl Personen, n = Anzahl Items)
startbetas Start-Item-Parameter (nX2-Matrix, erste Spalte Schwierigkeitsparameter, zweite Spalte Trennschärfeparameter, n = Anzahl Items)
runden Anzahl Iterations-Runden
Knoten Anzahl Knoten bei der Gauss-Hermite-Integration (zwischen 1 und 101)
Protokoll Bei TRUE wird ein Protokoll der Iterationen geliefert

Value

Bei Protokoll = T wird eine Liste mit den geschätzten Parametern und den Ergebnissen der verschiedenen Iterationen geliefert, durch rep(0,2) getrennt. Bei Protokoll = F werden nur die geschätzten Parameter als Matrix geliefert.

Author(s)

Paul Ruppen

References

Bock, R. & Aitkin, M. (1981). Marginal Maximum Likelihood Estimation of Item Parameters: Application of an EM Algorithm. *Psychometrika*, 46, 443-459.

Rizopoulos, D. (2006, November). ltm: An R Package for Latent Variable Modeling and Item Response Theory Analysis. *Journal of Statistical Software*, 17, (5).

Examples

```
X=SchaffDatDicho();X=X[[1]][,2:6];b=c(0,0,0,0,0);a=c(1,1,1,1,1);betas=cbind(b,a)
```

Funktionsbefehl

```
> BockAitkin
function (X, startbetas , runden = 50, Knoten = 15, Protokoll = F)
{
  n = ncol(X)
  Z = GHModKnot(Knoten)
  Xre = Xredu(X)
  ghw = gh[[Knoten]]$w * 1/sqrt(pi)
  betas1 = startbetas
  runde = 0
  repeat {
    runde = 1 + runde
    pr = probs(Z %*% t(betas1))
    p.xz <- exp(Xre$datenred %*% t(log(pr)) + ((1 - Xre$datenred) %*%
      t(log(1 - pr))))
    p.xzg = t(t(p.xz) * ghw)
    p.zx = p.xzg/rowSums(p.xzg)
    fk = colSums(p.zx * Xre$freqsXi)
    rkj = t(p.zx) %*% (Xre$datenred * Xre$freqsXi)
    matkoeff = matrix(rep(0, 2 * n), ncol = 2)
    for (j in 1:n) {
```

```

      a1 = glm(cbind(rkj[, j], fk - rkj[, j]) ~ Z[, 2],
              family = quasibinomial)
      matkoeff[j, 1] = a1$coefficients[1]
      matkoeff[j, 2] = a1$coefficients[2]
    }
    betas1 = matkoeff
    if (Protokoll == T) {
      betas2 = cbind(-betas1[, 1]/betas1[, 2], betas1[,
        2])
      startbetas = rbind(startbetas, rep(0, 2), betas2)
    }
    if (runde > runden)
      break
  }
  betas1[, 1] = -betas1[, 1]/betas1[, 2]
  colnames(betas1) = c("Schwierigkeit", "Trennschaerfe")
  if (Protokoll == T) {
    prot = list(Parameter = betas1, Protokoll = startbetas)
    prot
  }
  else betas1
}
<environment: namespace:irt >
dichte {irt}    R Documentation

```

Wahrscheinlichkeitsfunktion in Dichtefunktion

Description

Fuer diskrete Daten wird eine Dichtefunktion (Treppenfunktion) berechnet und gezeichnet.

Usage

dichte(W, AnzInt = NULL, Spalte = NULL, hoehe = NULL, tiefe = NULL, main = "Dichtefunktion Faehigkeiten versus geschaeztzte Dichtefunktion")

Arguments

W Datenmatrix mit Theta-Werten und Wahrscheinlichkeiten.
AnzInt Anzahl Intervalle, auf denen die Dichtefunktion jeweils konstant ist.
Spalte Es können die Spalten in der Datenmatrix festgelegt werden, die zur Berechnung zu verwenden sind. Voreinstellung zweite und vierte Spalte.
hoehe Graphikparameter fuer Anpassung der Höhe der Graphik.
tiefe Graphikparameter fuer Anpassung der Tiefe unter Theta-Achse.
main Haupttitel der Graphik

Details

Die Intervalle sind von konstanter Laenge. Das erste und das letzte Datum werden um die halbe Intervalllaenge ueberlappt. Die Anzahl der Intervalle wird am besten mit $n+1$ festgelegt (n = Anzahl Items). Die Faehigkeitsballungen liegen bei der Methode mit suffizienten Statistiken bei dieser Festlegung und bei symmetrischen Verteilungen ungefähr in der Mitte der Intervalle. Die Routine ist fuer die Ausgabe beim Befehl: `SufstatEM(X1, betas, empVert =T)` gedacht.

Value

Es wird eine Graphik ausgegeben.

Author(s)

Paul Ruppen

Examples

```
X=SchaffDatDicho();X1=X[[1]][,2:6];trenn=c(1,1,1,1,1);schw=c(0,0,0,0,0);betas=cbind(schw,trenn)
a=SufstatEM(X1, betas, empVert =T )
dichte(a$Empirische_Verteilung,6,hoehe=0.45)
weich=smooth.spline(seq(-3,3,0.1), dnorm(seq(-3,3,0.1)), spar=0.5)
lines(weich,lty=2)
points(a$Empirische_Verteilung[,2],a$Empirische_Verteilung[,4])
legend("bottomleft",legend=c("Dichte Normalverteilung",
"geschaetzte Dichte - Treppenfunktion",
"Wahrscheinlichkeiten pro Theta"),
lty=c(2,1,0),pch=c(-1,-1,1),bty="n")
```

```
X=SchaffDatDicho(n=8)
X1=X[[1]][,2:9];trenn=c(1,1,1,1,1,1,1,1);schw=c(0,0,0,0,0,0,0,0);betas=cbind(schw,trenn)
a=SufstatEM(X1, betas, empVert =T )
dichte(a$Empirische_Verteilung,AnzInt=9,hoehe=0.45,tiefe=0.1)
weich=smooth.spline(seq(-3,3,0.1), dnorm(seq(-3,3,0.1)), spar=0.5)
lines(weich,lty=2)
points(a$Empirische_Verteilung[,2],a$Empirische_Verteilung[,4])
legend("bottomleft",legend=c("Dichte Normalverteilung",
"geschaetzte Dichte - Treppenfunktion",
"Wahrscheinlichkeiten pro Theta"),
lty=c(2,1,0),pch=c(-1,-1,1),bty="n")
```

```
X=SchaffDatDicho(distr = 1, unif=c(-5,5), n=8)
X1=X[[1]][,2:9];trenn=c(1,1,1,1,1,1,1,1);schw=c(0,0,0,0,0,0,0,0);betas=cbind(schw,trenn)
a=SufstatEM(X1, betas, empVert =T )
```

```

dichte(a$Empirische.Verteilung,AnzInt=9, tiefe=0.06)
h=5/sqrt(10^2/12);h
lines(c(-h,h),c(1/(2*h),1/(2*h)),lty=2)
points(a$Empirische.Verteilung[,2],a$Empirische.Verteilung[,4])
legend("bottomleft",legend=c("stand. Dichtefunktion Thetas", "Geschaetzte Dichte - Treppenfunktion",
"Wahrscheinlichkeiten pro Theta"),
lty=c(2,1,0),pch=c(-1,-1,1),bty="n")
sum(a$Empirische.Verteilung[,4])

X=SchaffDatDicho(distr = 2, df = 5,n=8)
X1=X[[1]][,2:9];trenn=c(1,1,1,1,1,1,1,1);schw=c(0,0,0,0,0,0,0,0);betas=cbind(schw,trenn)
a=SufstatEM(X1, betas, empVert = T )
a$Empirische.Verteilung[,2]=(a$Empirische.Verteilung[,2]+5/sqrt(10))*sqrt(10)
dichte(a$Empirische.Verteilung,AnzInt=8,hoehe=0.45,tiefe=0.2)
dat=seq(0,15,0.1);chi=dchisq(dat,5);
weich=smooth.spline(dat,chi, spar=0.5)
lines(weich,lty=2)
points(a$Empirische.Verteilung[,2],a$Empirische.Verteilung[,4])
legend("bottomleft",legend=c("Dichte Thetas", "geschaetzte Dichte - Treppenfunktion",
"Wahrscheinlichkeiten pro Theta"),
lty=c(2,1,0),pch=c(-1,-1,1),bty="n")
[Package irt version 1.0 Index]

```

Funktionsbefehl

```

> dichte
function (W, AnzInt = NULL, Spalte = NULL, hoehe = NULL, tiefe = NULL,
  main = "Dichtefunktion Faehigkeiten versus geschaeetzte Dichtefunktion")
{
  if (is.null(Spalte)) {
    z = W[, 2]
    w = W[, 4]
  }
  else {
    z = W[, Spalte[1]]
    w = W[, Spalte[2]]
  }
  datf = as.data.frame(cbind(z, w))
  dat = as.matrix(datf[order(datf$z), ])
  z = as.vector(dat[, 1])
  w = as.vector(dat[, 2])
  q = length(z)
  if (is.null(AnzInt))
    AnzInt = 5
  IL = (max(z) - min(z))/(AnzInt - 1)
  IntGrenz = seq(z[1] - IL/2, by = IL, length.out = AnzInt)
  vek = (IntGrenz[1] < z) & (z <= IntGrenz[2])
  for (i in 2:(AnzInt - 1)) {
    vek1 = (IntGrenz[i] < z) & (z <= IntGrenz[i + 1])
    vek = cbind(vek, vek1)
  }
  vek = cbind(vek, z > IntGrenz[AnzInt])
}

```

```

WerteW = t(vek) %% w
HD = WerteW/IL
if (!is.null(hoehe))
  h = hoehe
else h = max(c(HD, w))
if (!is.null(tiefe))
  t = tiefe
else t = 1/(2 * h)
plot(c(min(z) - IL, max(z) + IL), c(-t, h), type = "n", main = main,
      xlab = "thetas", ylab = "Werte Dichtefunktion; Wahrscheinlichkeiten")
abline(-t + t * 0.7, 0, col = "gray", lty = 2)
for (i in 1:(AnzInt - 1)) lines(c(IntGrenz[i], IntGrenz[i +
  1]), c(HD[i], HD[i]))
lines(c(IntGrenz[AnzInt], IntGrenz[AnzInt] + IL), c(HD[AnzInt],
  HD[AnzInt]))
}
<environment: namespace:irt >

```

Item-Parameter-Schaetzung mit geschaezter empirischer Verteilung gemaess Bock-Aitkin**Description**

Es werden die Item-Parameter des logistischen Zweiparameter-Modells von Birnbaum mit einer von Bock-Aitkin (1981) skizzierten Methode geschaezt. Fuer fixe Punkte (Faehigkeiten) werden die erwarteten Anzahlen von Personen mit den entsprechenden Faehigkeiten geschaezt. Diese Anzahl geteilt durch die Anzahl der Personen ergibt eine geschaezte Wahrscheinlichkeitsfunktion auf die Knotenpunkte, die statt der Gauss-Hermite-Gewichte verwendet wird. Die entsprechenden Schaetzungen werden fuer jeden EM-Durchgang neu berechnet. Statt des Bock-Aitkin-EM-Algorithmus wird der EM-Algorithmus aus dem ltm-Paket von Rizopoulos verwendet, da dieser stabiler ist.

Usage

```
Empiric_dist_EM(X, startbetas, Knoten = 15, Intervall = NULL, runden = 50, Protokoll = F,
empVert = F, dfchi = NULL)
```

Arguments

X	NXn-Matrix mit Antwortvektoren pro Person als Zeilen (N = Anzahl Personen, n = Anzahl Items)
startbetas	Start-Item-Parameter (nX2-Matrix, erste Spalte Schwierigkeitsparameter, zweite Spalte Trennschärfeparameter, n = Anzahl Items)
Knoten	Anzahl Knoten fuer die geschaezte empirische Verteilung
Intervall	Bei Angabe von Intervallgrenzen als Vektor c(-a,a) mit $a > -a$ werden die Anzahl der Knoten regelmaessig auf das Intervall verteilt. Als Gewicht wird $1/(\text{Anzahl Knoten})$ gewaehlt. Ohne Angabe werden im ersten Durchgang Gauss-Hermite-Punkte und Gauss-Hermite-Gewichte verwendet.
runden	Anzahl EM-Iterations-Runden
Protokoll	Bei TRUE wird ein Protokoll der Iterationen geliefert
empVert	Wenn TRUE wird eine Liste ausgegeben; zweites Listenelement: Knoten mit letzten, geschaezten Wahrscheinlichkeitsfunktion auf den Knotenpunkten.
dfchi	Wenn ein Freiheitsgrad angegeben wird, werden die Knoten-Punkte in gleichen Abstaenden auf ein Intervall verteilt, so dass die Wahrscheinlichkeit ausserhalb des letzten rechten Punktes zu liegen, 0.99 betraegt.

Value

Liefert Matrix der geschaezten Item-Parameter. Bei Protokoll = T wird eine Liste mit zwei Listenelementen geliefert: zuerst Matrix der geschaezten Item-Parameter, dann Protokoll des Iterationsprozesses. Bei empVert= TRUE wird eine Liste ausgegeben: erstes Listenelement Matrix der geschaezten Item-Parameter; zweites Listenelement Matrix der Knoten und der letzten, geschaezten Wahrscheinlichkeitsfunktion auf den Knotenpunkten.

Author(s)

Paul Ruppen

References

Bock, R. & Aitkin, M. (1981). Marginal Maximum Likelihood Estimation of Item Parameters: Application of an EM Algorithm. *Psychometrika*, 46, 443-459.

Rizopoulos, D. (2006, November). ltm: An R Package for Latent Variable Modeling and Item Response Theory Analysis. *Journal of Statistical Software*, 17, (5).

Examples

```
X=ScaffDatDicho();X=X[[1]][,2:6];b=c(0,0,0,0,0);a=c(1,1,1,1,1);betas=cbind(b,a)
Empiric_dist_EM(X,betas)
[Package irt version 1.0 Index]
```

Funktionsbefehl

```
> Empiric_dist_EM
function (X, startbetas , Knoten = 15, Intervall = NULL, runden = 50,
  Protokoll = F, empVert = F, dfchi = NULL)
{
  p = ncol(X)
  N = nrow(X)
  startbetas1 = startbetas
  Xre = Xredu(X)
  X2 = Xre$datenred
  obs = Xre$freqsXi
  mX = 1 - X2
  if (is.null(Intervall)) {
    ghw = gh[[Knoten]]$w * 1/sqrt(pi)
    Z = GHModKnot(Knoten)
  }
  if (!is.null(dfchi)) {
    ghw = gewChiVert(kpunkte = Knoten, df = dfchi)[, 2]
    Zhi = gewChiVert(kpunkte = Knoten, df = dfchi)[, 1] -
      dfchi
    Z = cbind(rep(1, Knoten), Zhi)
  }
  if (!is.null(Intervall)) {
    ghw = rep(1/Knoten, Knoten)
    Z = seq(from = Intervall[1], to = Intervall[2], by = (Intervall[2] -
      Intervall[1])/(Knoten - 1))
    Z = cbind(rep(1, Knoten), Z)
  }
  for (it in 1:runden) {
    betas1 = startbetas1
    pr = probs(Z %*% t(betas1))
    qr = 1 - pr
    mX = 1 - X2
    p.zx = exp(X2 %*% t(log(pr)) + mX %*% t(log(qr)))
    p.x = c(p.zx %*% ghw)
    p.zx = (p.zx/p.x) * obs
    f = ghw * colSums(p.zx)
    nb = matrix(0, p, 2)
    dvar = pr * qr
    for (i in 1:p) {
      Y = outer(X2[, i], pr[, i], "-")
      sc = colSums((p.zx * Y) %*% (Z * ghw))
      hes = crossprod(Z, (dvar[, i] * f) * Z)
      nb[i, ] = betas1[i, ] + solve(hes, sc)
    }
    betas1 = nb
    if (Protokoll == T) {
      betas2 = cbind(-betas1[, 1]/betas1[, 2], betas1[,
        2])
      startbetas = rbind(startbetas , rep(0, 2), betas2)
    }
  }
  ghw = f/N
  startbetas1 = betas1
}
```

```

}
betas1[, 1] = -betas1[, 1]/betas1[, 2]
colnames(betas1) = c("Schwierigkeit", "Trennschaerfe")
betas1
if (Protokoll == T) {
  prot = list(Parameter = betas1, Protokoll = startbetas)
  prot
}
else betas1
if (empVert == T) {
  emp = list(Parameter = betas1, Empirische_Verteilung = cbind(Z[,
  2], f/N))
  emp
}
else betas1
}
<environment: namespace:irt >

```

Berechnung von Mittelwerten und Standardabweichung von Item-Parameter-Saetzen.**Description**

Fuer die Ergebnisse von MCEval (Liste von Matrizen mit Item-Parametern) werden die Mittelwerte und die Standardabweichungen der geschaezten Itemparameter berechnet.

Usage

```
evalMCEval(Ergebnisliste)
```

Arguments

Ergebnisliste Liste, die durch MCval produziert wird.

Value

Liste mit Mittelwerten und Standardabweichungen (pro Listenelement des Ergebnisses von MCEval je ein Listenelement).

Author(s)

Paul Ruppen

Examples

```
a=MCEval(BA=T, RIZ=T); evalMCEval(a)
#Schaetzung bei normalverteilten Faehigkeiten fuer BA und RIZ (fuer 200 Datensaeetze).
al=MCEval(distr=2,df=5,ED=T,Intervall=c(-10,10)); evalMCEval(al)
#chi-quadrat verteilte Faehigkeiten, geschaezt mit uniformer Startverteilung auf Intervall;
a=MCEval(ED=T,distr=1,unif=c(-3,3),Intervall=c(-3,3)); evalMCEval(a)
#Schaetzung bei uniform verteilte Faehigkeiten (unif) durch geschaezte empirische Verteilung,
#Uniform verteilte Faehigkeiten auf unif, Startverteilung uniforme Verteilung auf Intervall
al=MCEval(distr=2,df=5,ED=T,dfchi=5); evalMCEval(al)
# Schaetzung mit empirischer Verteilung, chi-quadrat-verteilte Faehigkeiten,
# Startverteilung: gleiche Verteilung wie Verteilung der Faehigkeiten.
a=MCEval(SuS=T); evalMCEval(a)
#Schaetzung mit suffizienter Statistik (normalverteilte Faehigkeiten)
a=MCEval(SuS=T,paraMatrix=cbind(c(-2,-1,0,1,2),c(1,2,3,2,0.5))); evalMCEval(a)
#Schaetzung mit suffizienter Statistik; variable Trennschaerfen; normalverteilte Faehigkeiten
a=MCEval(SuS=T,distr=1,unif=c(-10,10)); evalMCEval(a)
#Schaetzung mit suffizienter Statistik; uniform verteilte Faehigkeiten
a=MCEval(SuS=T,distr=1,unif=c(-20,20)); evalMCEval(a)
#Schaetzung mit suffizienter Statistik; uniform verteilte Faehigkeiten
a=MCEval(SuS=T,distr=1,unif=c(-4,4)); evalMCEval(a)
#Schaetzung mit suffizienter Statistik; uniform verteilte Faehigkeiten;
a=MCEval(SuS=T,distr=2,df=5); evalMCEval(a)
#Schaetzung mit suffizienter Statistik; chi-quadrat-verteilte Faehigkeiten, df=5;
a=MCEval(SuS=T,distr=2,df=2); evalMCEval(a)
#Schaetzung mit suffizienter Statistik; chi-quadrat-verteilte Faehigkeiten, df=2;
a=MCEval(SuS=T,RIZ=T); evalMCEval(a)
#normalverteilte Faehigkeiten, RIZ- und SUS- Verfahren.
[Package irt version 1.0 Index]
```

Funktionsbefehl

```
> MCEval
function (N = 1500, n = 5, a = NULL, b = NULL, paraMatrix = NULL,
  distr = 0, unif = NULL, df = NULL, Runden = 200, BA = F,
  RIZ = F, ED = F, SuS = F, Intervall = NULL, dfchi = NULL,
  Knoten = 15, EMrunden = 50)
{
  if (is.null(a)) a = rep(1, n)
  if (is.null(b)) b = rep(0, n)
  betas = cbind(b, a)
  X = SchaffDatDicho(n = n, N = N, paraMatrix = paraMatrix,
    distr = distr, unif = unif, df = df)
  X1 = X[[1]][, 1:n + 1]
  if (BA == T) betas1 = BockAitkin(X1, betas, Knoten = Knoten,
    runden = EMrunden) else betas1 = NULL
  if (RIZ == T) betas2 = RizopoulosEM(X1, betas, Knoten = Knoten,
    runden = EMrunden) else betas2 = NULL
  if (ED == T) betas3 = Empiric_dist_EM(X1, betas, Intervall = Intervall,
    Knoten = Knoten, runden = EMrunden, dfchi = dfchi)
    else betas3 = NULL
  if (SuS == T)
    betas4 = SuffstatEM(X1, betas, runden = EMrunden)
    else betas4 = NULL
  Runde = 1
  Runden = Runden - 1
  repeat {
    X = SchaffDatDicho(n = n, N = N, paraMatrix = paraMatrix,
      distr = distr, unif = unif, df = df)
    X1 = X[[1]][, 1:n + 1]
    if (BA == T) betas1 = cbind(betas1, BockAitkin(X1, betas,
      Knoten = Knoten, runden = EMrunden))
    if (RIZ == T) betas2 = cbind(betas2, RizopoulosEM(X1, betas,
      Knoten = Knoten, runden = EMrunden))
    if (ED == T) betas3 = cbind(betas3, Empiric_dist_EM(X1, betas,
      Intervall = Intervall, Knoten = Knoten, runden = EMrunden,
      dfchi = dfchi))
    if (SuS == T) betas4 = cbind(betas4, SuffstatEM(X1, betas,
      runden = EMrunden))
    if (Runde == Runden)
      break
    Runde = Runde + 1
  }
  res = list(BockAitkin = betas1, RizopoulosEM = betas2,
    Empirische_Verteilung = betas3, SuffStat = betas4)
  if (is.null(res$BockAitkin)) res$BockAitkin = NULL
  if (is.null(res$RizopoulosEM)) res$RizopoulosEM = NULL
  if (is.null(res$Empirische_Verteilung)) res$Empirische_Verteilung = NULL
  if (is.null(res$SuffStat)) res$SuffStat = NULL
  res
}
<environment: namespace:irt >
```

Berechnung von Faehigkeiten auf Grund einer Antwortematrix und Item-Parametern mit der R-Funktion optimize

Description

Berechnet mit der R-Funktion optimize die Maxima der log-likelihood-Funktionen zur Schätzung der Fähigkeitsparameter

Usage

```
faehig_mit_optimize(X, matPara)
```

Arguments

X Matrix von Antwortvektoren (N X n), N Anzahl Antwortvektoren, n = Anzahl Items
 matPara Matrix der Parameter (n X 2), N = Anzahl Items

Value

Vektor mit geschätzten Faehigkeiten

Author(s)

Paul Ruppen

Examples

```
X=SchaffDatDicho();X1=X[[1]][,2:6];b=c(0,0,0,0,0);a=c(1,1,1,1,1);betas=cbind(b,a)
para=RizopoulosEM(X1,betas); faehig_mit_optimize(X1,para)
[Package irt version 1.0 Index]
```

Funktionsbefehl

```
> faehig_mit_optimize = function (X, matPara)
{
  pjdi = function(teta , matPara, j) {
    exp(matPara[j, 2] * (teta - matPara[j, 1]))/(1 + exp(matPara[j,
      2] * (teta - matPara[j, 1])))
  }
  logLik = function(teta , matPara, daten, i) {
    m = length(matPara[, 1]); faehig1 = 0
    for (j in 1:m) { faehig1 = (daten[i, j] * log(pjdi(teta , matPara,
      j)) + (1 - daten[i, j]) * log(1 - pjdi(teta ,
      matPara, j))) + faehig1
    }
    faehig1
  }
  paramPersonen = function(daten1, matPara) {
    n = length(daten1[, 1])
    faehign = rep(0, n)
    for (i in 1:n) {
      logLik1 = function(teta) logLik(teta , matPara, daten1, i)
      d = optimize(logLik1, c(-15, 15), maximum = T)
      faehign[i] = d[1]$maximum
    }
    faehign = (faehign - mean(faehign))/sd(faehign)
    faehign
  }
  paramPersonen(X, matPara)
}
<environment: namespace:irt >
```

Diskrete Version (Wahrscheinlichkeitsfunktion) der Dichte einer Chi-Quadratverteilung**Description**

Es werden auf Punkte die Intervalle zwischen den Punkten halbiert und die entsprechenden Wahrscheinlichkeiten zusammengefasst. Im ersten Intervall beträgt dessen Länge die Hälfte der übrigen Intervalle. Die Summe der Wahrscheinlichkeiten beträgt 0.99. Die verbleibende Wahrscheinlichkeit wird nicht auf den letzten Punkt geschlagen, da dies den Schätzprozess beeinträchtigt. Die W-Funktion spielt beim Schätzprozess nur in der ersten Runde eine Rolle.

Usage

```
gewChiVert(kpunkte = Knoten, df)
```

Arguments

kpunkte Anzahl Knotenpunkte
df Anzahl Freiheitsgrade

Value

Matrix, erste Spalte Knotenpunkte, zweite Spalte Wahrscheinlichkeiten. Die Knotenpunkte werden in der Funktion `Empiric_dist_EM` um die Freiheitsgrade nach links verschoben.

Author(s)

Paul Ruppen

Examples

```
gewChiVert(15,3)
[Package irt version 1.0 Index]
```

Funktionsbefehl

```
> gewChiVert
function (kpunkte = Knoten, df)
{
  qchisq(0.99, df)
  laenge = df + qchisq(0.99, df)
  l = laenge/(kpunkte - 1)
  vek = c(1/2, seq(1 + 1/2, laenge + 1/2, 1))
  a = diff(pchisq(vek, df))
  a = c(pchisq(vek[1], df), a)
  cbind(vek, a)
}
<environment: namespace:irt >
```

Punkte und Gewichte fuer Gauss-Hermite-Integration

Description

Es werden die Punkte und Gewichte fuer die Gauss-Hermite-Integration geliefert, $n = 1$ bis 101.

Usage

```
data("gh")
```

Format

Eine Liste mit 101 Listen mit je zwei Listenelementen

Details

Zugriff auf GH-Punkte fuer n mit `gh[[n]][1]` und auf GH-Gewichte mit `gh[[n]][2]`

Source

Aus data-Verzeichnis des R-Paketes ltm uebernommen (s. references)

References

Rizopoulos, D. (2006, November). ltm: An R Package for Latent Variable Modeling and Item Response Theory Analysis. *Journal of Statistical Software*, 17, (5).

Examples

```
gh[[15]][1]
```

```
gh[[15]][2]
```

```
[Package irt version 1.0 Index]
```

Modell-Matrix mit Gauss-Hermite-Knoten fuer Dichte der Standardnormalverteilung**Description**

Fuer ein spezifisches n (Anzahl GH-Knoten) wird eine Modellmatrix produziert, die in der ersten Spalte konstant 1 und in der zweiten Spalte die GH-Knoten fuer die Dichte der Standardnormalverteilung aufweist.

Usage

```
GHModKnot(q)
```

Arguments

q Eine ganze Zahl q zwischen 1 und 101

Details

Die GH-Knoten werden den Daten `gh.rda` entnommen. Siehe den Beschrieb der entsprechenden Datei. Die GH-Punkte werden mit $\sqrt{2}$ multipliziert.

Value

Modellmatrix; erste Spalte konstant 1, zweite Spalte GH-Knoten mit $\sqrt{2}$ multipliziert

Author(s)

Paul Ruppen

References

Fuer Umberechnung der GH-Knoten s. z.B. Baker, F. B. und Kim, S.-H. (2004), Item Response Theory: Parameter Estimation Techniques. Basel: Dekker, S. 164

Examples

```
GHModKnot(15)
```

[Package irt version 1.0 Index]

Funktionsbefehl

```
> GHModKnot
function (q)
{
  xk = gh[[q]]$x * -sqrt(2)
  Z = model.matrix(~xk)
  attr(Z, "assign") = NULL
  Z
}
<environment: namespace:irt >
```

Charakteristische Item-Kurven (Funktionen)**Description**

Mit Hilfe der zu lieferenden Schwierigkeits-Parameter (Lageparameter) und Trennschaerfeparameter (Steigung in Lageparameter) wird eine logistische ICC erstellt. Es handelt sich um eine R-Funktion, die gezeichnet werden kann oder deren Werte berechnet werden können

Usage

```
icc(schwier, diskrim)
```

Arguments

schwier Lageparameter
 diskrim Steigung im Lageparameter. Es ist eine Zahl > 0 einzugeben

Details

$icc(x) = \exp(a(x-b)) / (1 + \exp(a(x-b)))$ mit Schwierigkeitsparameter b und Trennschaerfeparameter a .

Value

Es wird eine Funktion geliefert

Author(s)

Paul Ruppen

References

IRT-Literatur, z.B. Baker und Kim (2004), Item Response Theory: Parameter Estimation Techniques. Basel:Decker.

Examples

```
f=icc(1, 2)
f(1.5)
plot(f,xlim=c(-2,4))
[Package irt version 1.0 Index]
```

Funktionsbefehl

```
> icc
function (schwier, diskrim)
{
  function(x) exp(diskrim * (x - schwer)) / (1 + exp(diskrim *
    (x - schwer)))
}
<environment: namespace:irt >
```

Berechnung der Parameter fuer viele jeweils gemeinsame Datensaeetze**Description**

Fuer die vier behandelten Schaetzmethodeen koennen fuer viele jeweils gemeinsame Datensaeetze die Parameter berechnet werden (Standardeinstellung 200). Die Verteilung der Datensaeetze kann festgelegt werden (standardnormalverteilt, uniform oder chi-quadrat-verteilt). Ebenso kann die Anzahl der Items sowie die Anzahl der Antwortevektoren sowie die Start-W-Funktion beim Schaetzprozess fuer die 'empirische Methode' festgelegt werden. Die Routine dient der Evaluation der Schaetzverfahren. Mit evalMCeval koennen in der Folge die Mittelwerte und Standardabweichungen der Itemparameter der Datensaeetze berechnet werden.

Usage

```
MCeval(N = 1500, n = 5, a = NULL, b = NULL, paraMatrix = NULL, distr = 0, unif = NULL,
df = NULL, Runden = 200, BA = F, RIZ = F, ED = F, SuS = F, Intervall = NULL, dfchi =
NULL, Knoten = 15, EMrunden = 50)
```

Arguments

N	Stichprobengroesse (Anzahl Antwortevektoren). Voreinstellung 1500.
n	Anzahl Items. Voreinstellung 5.
a	Start-Trennschaerfevektor fuer die EM-Iterationen. Voreinstellung (1,1,1,1,1). Der Vektor muss n (= Anzahl) Items Komponenten enthalten.
b	Start-Schwierigkeitsvektor fuer die EM-Iterationen. Voreinstellung (0,0,0,0,0). Der Vektor muss n (= Anzahl Items) Komponenten enthalten.
paraMatrix	Matrix der Trennschaerfe- und der Schwierigkeitsparameter, die fuer die Schaffung der kuenstlichen Daten verwendet werden (Voreinstellung (1,1,1,1,1) und (-2,-1,0,1,2)). Die Matrix muss nx2, n = Anzahl Items, Komponenten enthalten.
distr	Wahl der Verteilung, nach der die kuenstlichen Daten verteilt sind. 0 = Voreinstellung = standardnormalverteilte Faehigkeiten; 1 = uniform verteilt (Voreinstellung [-2,2], zu aendern bei unif); 2 = chi-verteilt (Voreinstellung df = 2; zu aendern unter df).
unif	Bei distr=1 kann hier die Intervallbreite der uniformen Verteilung festgelegt werden - Voreinstellung [-2,2].
df	Bei distr=2 koennen hier die Freiheitsgrade festgelegt werden - Voreinstellung df=2.
Runden	Anzahl kuenstlicher Datensaeetze, die ausgewertet werden. Voreinstellung 200.
BA	Wenn BA=TRUE, wird die Bock-Aitkins-Methode auf die Datensaeetze angewendet.
RIZ	Wenn RIZ=TRUE, wird die Methode von Rizopoulos angewendet.
ED	Wenn ED=TRUE, wird die Methode mit Hilfe der geschaezten W-Funktion angewendet.
SuS	Wenn ED=TRUE, wird die Methode mit den suffizienten Statistiken angewendet.
Intervall	Intervallbreite fuer Startverteilung bei ED=T.
dfchi	Anzahl Freiheitsgrade fuer Startverteilung bei ED=T.
Knoten	Anzahl Stuetzpunkte oder Knoten.
EMrunden	Anzahl EM-Iterationsrunden.

Value

Es wird eine Liste mit je einer Matrix (Itemparameter) pro Datensatz (Voreinstellung 200 Datensaeetze) geliefert. Anzahl Listenelemente haengt von der Anzahl auf TRUE gesetzter Parameter BA, RIZ, ED und SuS ab.

Author(s)

Paul Ruppen

Examples

```
a=MCeval(BA=T, RIZ=T); evalMCeval(a)
```

```

#Schaetzung bei normalverteilten Faehigkeiten fuer BA und RIZ (fuer 200 Datensaeetze).
al=MCeval(distr=2,df=5,ED=T,Intervall=c(-10,10)); evalMCeval(al)
#chi-quadrat verteilte Faehigkeiten, geschaezt mit uniformer Startverteilung auf Intervall;
a=MCeval(ED=T,distr=1,unif=c(-3,3),Intervall=c(-3,3)); evalMCeval(a)
#Schaetzung bei uniform verteilte Faehigkeiten (unif) durch geschaezte empirische W-Funktion,
#Uniform verteilte Faehigkeiten auf unif, Startverteilung uniforme Verteilung auf Intervall
al=MCeval(distr=2,df=5,ED=T,dfchi=5); evalMCeval(al)
# Schaetzung mit empirischer Verteilung, chi-quadrat-verteilte Faehigkeiten,
# Startverteilung: gleiche Verteilung wie Verteilung der Faehigkeiten.
a=MCeval(SuS=T); evalMCeval(a)
#Schaetzung mit suffizienter Statistik (normalverteilte Faehigkeiten)
a=MCeval(SuS=T,paraMatrix=cbind(c(-2,-1,0,1,2),c(1,2,3,2,0.5))); evalMCeval(a)
#Schaetzung mit suffizienter Statistik; variable Trennschaerfen; normalverteilte Faehigkeiten
a=MCeval(SuS=T,distr=1,unif=c(-10,10)); evalMCeval(a)
#Schaetzung mit suffizienter Statistik; uniform verteilte Faehigkeiten
a=MCeval(SuS=T,distr=1,unif=c(-20,20)); evalMCeval(a)
#Schaetzung mit suffizienter Statistik; uniform verteilte Faehigkeiten
a=MCeval(SuS=T,distr=1,unif=c(-4,4)); evalMCeval(a)
#Schaetzung mit suffizienter Statistik; uniform verteilte Faehigkeiten;
a=MCeval(SuS=T,distr=2,df=5); evalMCeval(a)
#Schaetzung mit suffizienter Statistik; chi-quadrat-verteilte Faehigkeiten, df=5;
a=MCeval(SuS=T,distr=2,df=2); evalMCeval(a)
#Schaetzung mit suffizienter Statistik; chi-quadrat-verteilte Faehigkeiten, df=2;
a=MCeval(SuS=T,RIZ=T); evalMCeval(a)
#normalverteilte Faehigkeiten, RIZ- und SUS- Verfahren.
[Package irt version 1.0 Index]

```

Funktionsbefehl

```

> MCeval
function (N = 1500, n = 5, a = NULL, b = NULL, paraMatrix = NULL,
  distr = 0, unif = NULL, df = NULL, Runden = 200, BA = F,
  RIZ = F, ED = F, SuS = F, Intervall = NULL, dfchi = NULL,
  Knoten = 15, EMrunden = 50)
{
  if (is.null(a))      a = rep(1, n)
  if (is.null(b))      b = rep(0, n)
  betas = cbind(b, a)
  X = SchaffDatDicho(n = n, N = N, paraMatrix = paraMatrix,
    distr = distr, unif = unif, df = df)
  X1 = X[[1]][, 1:n + 1]
  if (BA == T)
    betas1 = BockAitkin(X1, betas, Knoten = Knoten, runden = EMrunden)
  else betas1 = NULL
  if (RIZ == T)
    betas2 = RizopoulosEM(X1, betas, Knoten = Knoten, runden = EMrunden)
  else betas2 = NULL
  if (ED == T)
    betas3 = Empiric_dist_EM(X1, betas, Intervall = Intervall,
      Knoten = Knoten, runden = EMrunden, dfchi = dfchi)
  else betas3 = NULL
}

```

```

if (SuS == T)
  betas4 = SufstatEM(X1, betas, runden = EMrunden)
else betas4 = NULL
Runde = 1
Runden = Runden - 1
repeat {
  X = SchaffDatDicho(n = n, N = N, paraMatrix = paraMatrix,
    distr = distr, unif = unif, df = df)
  X1 = X[[1]][, 1:n + 1]
  if (BA == T)
    betas1 = cbind(betas1, BockAitkin(X1, betas, Knoten = Knoten,
      runden = EMrunden))
  if (RIZ == T)
    betas2 = cbind(betas2, RizopoulosEM(X1, betas, Knoten = Knoten,
      runden = EMrunden))
  if (ED == T)
    betas3 = cbind(betas3, Empiric_dist_EM(X1, betas,
      Intervall = Intervall, Knoten = Knoten, runden = EMrunden,
      dfchi = dfchi))
  if (SuS == T)
    betas4 = cbind(betas4, SufstatEM(X1, betas, runden = EMrunden))
  if (Runde == Runden)
    break
  Runde = Runde + 1
}
res = list(BockAitkin = betas1, RizopoulosEM = betas2,
  Empirische_Verteilung = betas3, SuffStat = betas4)
if (is.null(res$BockAitkin)) res$BockAitkin = NULL
if (is.null(res$RizopoulosEM)) res$RizopoulosEM = NULL
if (is.null(res$Empirische_Verteilung)) res$Empirische_Verteilung = NULL
if (is.null(res$SuffStat)) res$SuffStat = NULL
res
}
<environment: namespace:irt >

```

Plot fuer 'diskrete Normalverteilung'**Description**

Vorgegebenen Punkten Z_k , die mit gleichen Abständen auseinander liegen, wird die Wahrscheinlichkeit, dass die Werte einer standardnormalverteilten Zufallsvariable ins Intervall $[Z_j-d, Z_j+d]$ fallen, zugeordnet. Bei Z_1 wird mit $]-\infty, Z_1+d[$ gerechnet, beim letzten Punkt Z_q mit $[Z_q-d, \infty[$ mit der Haelfte d der Distanzen zwischen den Punkten des einzugebenden Vektors

Usage

```
normdiskret(x)
```

Arguments

`x` ein Vektor von Punkten mit gleichem Abstand

Details

Zweck der Routine ist, die geschaeztzte empirische Verteilung mit der diskretisierten Standardnormalverteilung zu vergleichen, um im Fall normalverteilter Faehigkeiten, die geschätzte empirische Verteilung mit der Verteilung, welche die Faehigkeitsparameter produziert hat, zu vergleichen. Entsprechend ist der Titel der Graphik gewaehlt.

Value

Es wird eine Graphik ausgegeben

Author(s)

Paul Ruppen

Examples

```
X=SchaffDatDicho();X1=X[[1]][,2:6];b=c(0,0,0,0,0);a=c(1,1,1,1,1);betas=cbind(b,a)
bei1=Empiric_dist_EM(X1,betas,Intervall=c(-5,5),empVert=T,Knoten=50,runden=50)
normdiskret(bei1$Empirische_Verteilung[,1])
points(bei1$Empirische_Verteilung[,1],bei1$Empirische_Verteilung[,2],pch=2)
legend("bottom",legend=c("normalverteilt' diskret", "geschaeztzte relative Haeufigkeiten"),
      pch=c(1,2),bty="n")
[Package irt version 1.0 Index]
```

Funktionsbefehl

```
> normdiskret = function (x)
{
  d = (x[2] - x[1])/2
  n = length(x)
  y = seq(x[1] + d, x[n] - d, 2 * d)
  m = length(y)
  z = pnorm(y)
  z = diff(z, lag = 1)
  z = c(pnorm(y[1]), z, 1 - pnorm(y[m]))
  h = max(z)
  h = h + 1/5 * h
  plot(c(y[1] - 0.5, y[n - 1] + 0.5), c(-h/2.5, h), type = "n",
       xlab = "theta", ylab = "Wahrscheinlichkeiten",
       main = "Geschaetzte empirische Verteilung")
  points(x, z)
  abline(0, 0)
}
<environment: namespace:irt >
```

Graphische Darstellung von ICCs**Description**

Es werden die Item Character Curves (ICCs) fuer spezifische Parameter gezeichnet. Verwendet werden kuenstlich geschaffene Daten (aus der Liste, die mit dem Befehl SchaffDatDicho() geschaffen wird).

Usage

```
PlotDich(listeDich, xgrenz = NULL)
```

Arguments

listeDich Liste mit zwei Listenelementen, erzeugt durch SchaffDatDicho()
 xgrenz Grenzen fuer die graphische Darstellung

Details

Verwendet wird das zweite Listenelement von SchaffDatDicho(), naemlich die Matrix der Parameter. xgrenz: Voreinstellung c(-5,5). Falls die Voreinstellung geaendert wird, ist ein Vektor c(x,y) einzugeben.

Value

Graphische Darstellung

Author(s)

Paul Ruppen

Examples

```
list1=SchaffDatDicho(); PlotDich(list1)
[Package irt version 1.0 Index]
```

Funktionsbefehl

```
> PlotDich
function (listeDich , xgrenz = NULL)
{
  if (is.null(xgrenz))
    xgr = c(-5, 5)
  else xgr = xgrenz
  vek = listeDich [[1]][, 1]
  betas = listeDich [[2]]
  n = nrow(betas)
  icc = function(schwier , diskrim) {
    function(x) exp(diskrim * (x - schwer))/(1 + exp(diskrim *
      (x - schwer)))
  }
  plot(icc(betas[1, 1], betas[1, 2]), xlim = xgr, main = "ICCs",
    xlab = "Faehigkeiten", ylab = "Wahrscheinlichkeit korreketer Antwort")
  for (i in 2:n) {
    f = icc(betas[i, 1], betas[i, 2])
    curve(f, add = T, col = i)
  }
}
<environment: namespace:irt >
```

Werte der logistischen Verteilungsfunktion

Description

Es werden die Werte der logistischen Verteilungsfunktion berechnet (für die Komponenten einer Matrix), wobei Werte, die vom Programm auf 0 oder 1 gesetzt werden, von diesen weg um $x^{(1/2)}$ verschoben wird, mit der computerabhängigen kleinsten positiven Gleit-Komma-Zahl x .

Usage

```
probs(x)
```

Arguments

x Eine Matrix X

Details

Die Werte werden mit der R-Funktion `plogis` berechnet.

Value

Es wird eine Matrix geliefert

Author(s)

Paul Ruppen

References

Idee aus R-Paket `ltm` uebernommen (Rizopoulos, D. (2006, November). `ltm`: An R Package for Latent Variable Modeling and Item Response Theory Analysis. *Journal of Statistical Software*, 17, (5).

Examples

```
Z=matrix(c(1,1,1,1,-3,2,4,5),ncol=2)
betas=matrix(c(-3,0,1,1,1,1),ncol=2)
X=Z %*% t(betas)
probs(X)
```

Funktionsbefehl

```
> probs
function (x)
{
  pr = plogis(x)
  pr[pr == 1] = 1 - sqrt(.Machine$double.eps)
  pr[pr == 0] = sqrt(.Machine$double.eps)
  pr
}
<environment: namespace:irt >
```

EM-Algorithmus fuer die Schaetzung der Item-Parameter fuer dichotome, logistische IRT-Modelle nach Rizopoulos**Description**

Es werden die Item-Parameter des logistischen Zweiparameter-Modells von Birnbaum mit dem EM-Algorithmus von Rizopoulos im R-Paket ltm geschaetzt.

Usage

```
RizopoulosEM(X, startbetas, runden = 50, Knoten = 15, Protokoll = F)
```

Arguments

X	NXn-Matrix mit Antwortvektoren pro Person als Zeilen (N = Anzahl Personen, n = Anzahl Items)
startbetas	Start-Itemparameter fuer die Iteration (nX2-Matrix, erste Spalte Schwierigkeitsparameter, zweite Spalte Trennschärfeparameter, n = Anzahl Items)
runden	Anzahl Runden
Knoten	Anzahl GH-Knoten fuer die Gauss-Hermite-Integration
Protokoll	Bei TRUE wird ein Protokoll der Iterationen geliefert

Details

Im R-Paket wird der hier implementierte EM-Algorithmus im Befehl ltm.fit verwendet. Dort werden nach 15 Runden des EM-Algorithmus weitere Runden des BFGS-Algorithmus des R-Befehls optim() verwendet

Value

Bei Protokoll = T wird eine Liste mit den geschaetzten Parametern und den Ergebnissen der verschiedenen Iterationen geliefert, durch rep(0,2) getrennt. Bei Protokoll = F werden nur die geschaetzten Parameter als Matrix geliefert.

Author(s)

Paul Ruppen

References

EM.R im R-Paketes ltm

Rizopoulos, D. (2006, November). ltm: An R Package for Latent Variable Modeling and Item Response Theory Analysis. Journal of Statistical Software, 17, (5).

Examples

```
X=SchaffDatDicho();X=X[[1]][,2:6];b=c(0,0,0,0,0);a=c(1,1,1,1,1);betas=cbind(b,a)
RizopoulosEM(X,betas)
[Package irt version 1.0 Index]
```

Funktionsbefehl

```
> RizopoulosEM
function (X, startbetas , runden = 50, Knoten = 15, Protokoll = F)
{
  p = ncol(X)
  startbetas1 = startbetas
  Z = GHModKnot(Knoten)
  q. = ncol(Z)
  Xre = Xredu(X)
  X = Xre$datenred
  obs = Xre$freqsXi
  mX = 1 - X
  ghw = gh[[Knoten]]$w * 1/sqrt(pi)
```

```

betas1 = startbetas1
runde = 0
for (it in 1:runden) {
  pr = probs(Z %*% t(betas1))
  qr = 1 - pr
  p.xz = exp(X %*% t(log(pr)) + mX %*% t(log(qr)))
  p.x = c(p.xz %*% ghw)
  p.zx = (p.xz/p.x) * obs
  f = ghw * colSums(p.zx)
  nb = matrix(0, p, q.)
  dvar = pr * qr
  for (i in 1:p) {
    Y = outer(X[, i], pr[, i], "-")
    sc = colSums((p.zx * Y) %*% (Z * ghw))
    hes = crossprod(Z, (dvar[, i] * f) * Z)
    nb[i, ] <- betas1[i, ] + solve(hes, sc)
  }
  betas1 = nb
  if (Protokoll == T) {
    betas2 = cbind(-betas1[, 1]/betas1[, 2], betas1[,
      2])
    startbetas = rbind(startbetas, rep(0, 2), betas2)
  }
}
betas1[, 1] = -betas1[, 1]/betas1[, 2]
colnames(betas1) = c("Schwierigkeit", "Trennschaerfe")
if (Protokoll == T) {
  prot = list(Parameter = betas1, Protokoll = startbetas)
  prot
}
else betas1
}
<environment: namespace:irt >

```

Schaffung von dichotomen Antwortevektoren**Description**

Es wird eine Matrix mit den Faehigkeitsparametern als erster Spalte und je einem dichotomen Antwortevektor pro Zeile geliefert, der auf Grund des Faehigkeitsparameters und der gewaehlten ICCs erzeugt wird. Fuer die Faehigkeitsparameter kann aus drei Verteilungsfamilien gewaehlt werden: Standardnormalverteilung, uniforme Verteilung oder Chi-Quadrat-Verteilung, die um die Anzahl der Freiheitsgrade nach links verschoben wird.

Usage

```
SchaffDatDicho(N = 1500, n = 5, paraMatrix = NULL, distr = 0, unif = NULL, df = NULL,
standard = F)
```

Arguments

N	Anzahl Antwortevektoren
n	Anzahl Items
paraMatrix	(n,2)-Matrix mit den Parametern - Schwierigkeiten, Trennschaerfen, n = Anzahl Items; Voreinstellung: Trennschaerfen 1; Schwierigkeiten symmetrisch um 0 im Abstand von 1 verteilt
distr	distr=0 (Standardnormalverteilung); distr=1 (Uniforme Verteilung); distr=2 (Chi-Quadrat-Verteilung)
unif	Intervallgrenzen der uniformen Verteilung. Voreinstellung: -2.5,2.5
df	Freiheitsgrade fuer Chi-Quadrat-Verteilung. Voreinstellung df=2
standard	Bei T wird bei der Chi-Quadrat-Verteilung eine Standardisierung vorgenommen

Details

Voreinstellungen: Anzahl Antwortevektoren (N): 1500; Anzahl Items (n): 5; Um 0 mit gleichem Abstand von 1 symmetrisch verteilte Schwierigkeitsparameter und Trennschaerfeparameter von 1. Bei gerader Anzahl Items und automatischer Generierung von paraMatrix, liegt kein Schwierigkeitsparameter auf 0. Bei ungerader Anzahl Items, liegt der mittlere Parameter auf 0. Die Parameter werden zur Berechnung der Werte p der logistischen Funktion verwendet. Es werden mit den Wahrscheinlichkeiten p die Werte bernoulliverteilter Zufallsvariablen berechnet.

Value

Es wird eine Liste erzeugt.

Listenelement 1:

Matrix; erste Spalte Faehigkeitsvektor, andere Spalten zeilenweise Antwortevektoren

Listenelement 2:

Matrix der Schwierigkeiten und der Trennschaerfen, die zur Berechnung verwendet wurde

Author(s)

Paul Ruppen

Examples

```
SchaffDatDicho()
```

```
SchaffDatDicho(distr=1)
```

```
SchaffDatDicho(distr=2,df=3)
```

```
[Package irt version 1.0 Index]
```

Funktionsbefehl

```
> SchaffDatDicho
function (N = 1500, n = 5, paraMatrix = NULL, distr = 0, unif = NULL,
        df = NULL, standard = F)
{
  if (distr == 0)
    pers = rnorm(N)
  if (distr == 1) {
    if (is.null(unif))
      pers = runif(N, -2, 2)
    else pers = runif(N, unif[1], unif[2])
  }
  if (distr == 2) {
    if (is.null(df))
      pers = (rchisq(N, 2) - 2)
    else pers = (rchisq(N, df) - df)
    if (standard == T)
      pers = (pers - mean(pers))/sd(pers)
  }
  daten = pers
  if (is.null(paraMatrix)) {
    mat1 = matrix(rep(NA, 2 * n), ncol = 2)
    for (i in 1:n) mat1[, 2] = 1
    is.wholenumber = function(x, tol = .Machine$double.eps^0.5) {
      abs(x - round(x)) < tol
    }
    if (is.wholenumber(n/2))
      mat1[, 1] = seq(-n/2 + 0.5, n/2, 1)
    else mat1[, 1] = seq(trunc(-n/2), trunc(n/2), 1)
  }
  else mat1 = paraMatrix
  for (i in 1:n) {
    f = icc(mat1[i, 1], mat1[i, 2])
    daten = cbind(daten, variableDich(f, pers))
  }
  ret = list(Daten = daten, Matrix = mat1)
  ret
}
<environment: namespace:irt >
```

Item-Parameter-Schaetzung mit empirischen Haeufigkeiten auf den Werten der suffizienten Statistiken**Description**

Es werden die Item-Parameter des logistischen Zweiparameter-Modells von Birnbaum mit dem EM-Algorithmus des ltm-Paketes (Rizopoulos) geschaezt, wobei nicht mit der Gauss-Hermite-Integration gearbeitet wird. Als Knoten werden die Werte der suffizienten Statistiken $\text{sum}(a_j x_j)$ mit den vorlaeufigen a_j verwendet und als Gewichte die relativen Haeufigkeiten der Antwortvektoren mit dem dementsprechenden Wert $\text{sum}(a_j x_j)$ ($1 \leq j \leq n$, $n = \text{Anzahl Items}$).

Usage

```
SufstatEM(X, startbetas, runden = 50, Protokoll = F, empVert = F)
```

Arguments

X NXn-Matrix mit Antwortvektoren pro Person als Zeilen ($N = \text{Anzahl Personen}$, $n = \text{Anzahl Items}$)

startbetas Start-Item-Parameter (nX2-Matrix, erste Spalte Schwierigkeitsparameter, zweite Spalte Trennschärfeparameter, $n = \text{Anzahl Items}$)

runden Anzahl EM-Iterations-Runden

Protokoll Bei TRUE wird ein Protokoll der EM-Iterationen geliefert

empVert Bei TRUE wird eine Liste mit den geschaezten Item-Parametern und einer Matrix mit den Werten der suffizienten Statistiken, den standardisierten Statistiken, den relativen Haeufigkeiten und der geschaezten W-Funktion geliefert.

Value

Es wird eine Matrix mit den Trennschaerfe und Faehigkeitsparametern ausgegeben. Bei Protokoll=T eine Liste der Schaetzung der Parameter und mit den Ergebnissen aller EM-Iterationen. Bei empVert=T eine Liste mit den Parameter und der geschaezten Wahrscheinlichkeitsfunktion auf den Stuetzpunkten der letzten Runde

Author(s)

Paul Ruppen

References

Bock, R. & Aitkin, M. (1981). Marginal Maximum Likelihood Estimation of Item Parameters: Application of an EM Algorithm. *Psychometrika*, 46, 443-459.

Birnbaum, A. (1968). Some Latent Trait Models and their Use in Inferring an Examinee's Ability. In F. Lord & M. Novick (Hrsg.), *Statistical Theories of Mental Test Scores* (S. 397-479). Reading: Addison-Wesley.

EM.r Befehl im R-Paket ltm (Rizopoulos)

Examples

```
X=SchaffDatDicho();X=X[[1]][,2:6];b=c(0,0,0,0,0);a=c(1,1,1,1,1);betas=cbind(b,a)
SufstatEM(X,betas);SufstatEM(X,betas,empVert=T);SufstatEM(X,betas,Protokoll=T)
[Package irt version 1.0 Index]
```

Funktionsbefehl

```
> SufstatEM
function (X, startbetas , runden = 50, Protokoll = F, empVert = F)
{
  p = ncol(X)
  N = nrow(X)
  Xre = Xredu(X)
```

```

X2 = Xre$datenred
obs = Xre$freqsXi
mX = 1 - X2
obs = Xre$freqsXi
startbetas1 = startbetas
weighted.sd = function(x, w) {
  sqrt(1/N * sum(w * (x - weighted.mean(x, w))^2))
}
for (it in 1:runden) {
  Z1 = X2 %*% startbetas1[, 2]
  Z2 = Z1
  Z1 = (Z1 - weighted.mean(Z1, obs))/weighted.sd(Z1, obs)
  q = nrow(Z1)
  Z = cbind(rep(1, q), Z1)
  betas1 = startbetas1
  pr = probs(Z %*% t(betas1))
  qr = 1 - pr
  mX = 1 - X2
  p.xz = exp(X2 %*% t(log(pr)) + mX %*% t(log(qr)))
  p.x = c(p.xz %*% obs/N)
  p.zx = (p.xz/p.x) * obs
  f = obs * colSums(p.zx)/N
  nb = matrix(0, p, 2)
  dvar = pr * qr
  for (i in 1:p) {
    Y = outer(X2[, i], pr[, i], "-")
    sc = colSums((p.zx * Y) %*% (Z * obs/N))
    hes = crossprod(Z, (dvar[, i] * f) * Z)
    nb[i, ] = betas1[i, ] + solve(hes, sc)
  }
  betas1 = nb
  if (Protokoll == T) {
    betas2 = cbind(-betas1[, 1]/betas1[, 2], betas1[,
      2])
    startbetas = rbind(startbetas, rep(0, 2), betas2)
  }
  startbetas1 = betas1
}
betas1[, 1] = -betas1[, 1]/betas1[, 2]
colnames(betas1) = c("Schwierigkeit", "Trennschaerfe")
if (Protokoll == T) {
  prot = list(Parameter = betas1, Protokoll = startbetas)
  prot
}
else betas1
if (empVert == T) {
  EmpVert = list(Parameter = betas1, Empirische_Verteilung = cbind(Z2,
    Z1, obs/N, f/N))
  colnames(EmpVert$Empirische_Verteilung) = c("Faehig",
    "standard_Faehig", "rel_Haeuf", "geschaetzt_rel_Haeuf")
  EmpVert
}
else betas1
} <environment: namespace:irt >

```

Schaffung einer dichotomen Variable

Description

Fuer einen Faehigkeitsvektor der Laenge N werden Werte bernoulliverteilter Zufallsvariablen mit dem Parameter p berechnet. p wird mit Hilfe einer strikt monoton steigenden Funktion berechnet, welche Faehigkeitsparametern Zahlen p zwischen 0 und 1 zuordnet. Die Funktion wird im Befehl SchaffDatDicho() des irt-Paketes verwendet. Dort wird die ICC-Funktion des irt-Paketes fuer die Zuordnung der p verwendet.

Usage

```
variableDich(fun, pers)
```

Arguments

- fun Strikt monoton steigende Funktion, gemaess der den Faehigkeitsparametern Werte zwischen 0 und 1 zugeordnet werden.
- pers Vektor mit Faehigkeitsparametern

Value

Vektor der Laenge N mit Komponenten 0 und 1.

Author(s)

Paul Ruppen

Examples

```
pers=rnorm(50,0,1);variableDich(icc(1,2), pers)
[Package irt version 1.0 Index]
```

Funktionsbefehl

```
> variableDich
function (fun, pers)
{
  N = length(pers)
  var1 = rep(NA, N)
  for (i in 1:N) var1[i] = rbinom(1, 1, fun(pers[i]))
  var1
}
<environment: namespace:irt >
```

Funktion zur Reduktion der Antwortematrix**Description**

Die Funktion reduziert die Antwortematrix X, so dass nur mehr unterschiedliche Antwortvektoren vorkommen. Zudem wird eine Häufigkeitsverteilung der Antwortvektoren berechnet.

Usage

```
Xredu(X)
```

Arguments

X Matrix von dichotomen Antworten. Darf nur nur 0 und 1 enthalten.

Value

Es wird eine Liste ausgegeben:

\$datenred

enthaelt die reduzierte Antwortematrix

\$freqsXi

enthaelt die Haeufigkeiten der Antwortvektoren

Note

Etliches wurde aus der Funktion ltm.fit des R-Paktes ltm entnommen, s. Rizopoulos, D. (2006, November). ltm: An R Package for Latent Variable Modeling and Item Response Theory Analysis. Journal of Statistical Software, 17, (5).

Author(s)

Paul Ruppen

Examples

```
X=matrix(c(0,1,1,0,1,1,1,1,0,0,1,1,0,1,1,0,1,1,0,1),ncol=5,byrow=T)
```

```
Xredu(X)
```

```
[Package irt version 1.0 Index]
```

Funktionsbefehl

```
> Xredu = function (X)
{
  p = ncol(X)
  pats = apply(X, 1, paste, collapse = "")
  freqs = table(pats)
  obs = as.vector(freqs)
  cbind(names(freqs))
  f = function(X) {
    nX = nchar(X)
    out = substring(X, 1:nX, 1:nX)
    out = out[out != "A"]
    out[out == "N"] <- NA
    out
  }
  X = apply(cbind(names(freqs)), 1, f)
  nX = nchar(cbind(names(freqs)))
  out = substring(cbind(names(freqs))[1, ], 1:nX[1, 1], 1:nX[1, 1])
  out = out[out != "A"]
  out[out == "N"] <- NA
  X = as.numeric(t(X))
  dim(X) = c(length(freqs), p)
  list(datenred = X, freqsXi = obs)
} <environment: namespace:irt >
```

7 Liste der Symbole

\mathbb{N}_n^*	Menge der natürlichen Zahlen von 1 bis n
$\mathbb{E}(X)$	Erwartungswert der Zufallsvariable X
$1\{x \text{ ist ein } \varphi\}$	$:= \begin{cases} 1 & \text{wenn } x \text{ ein } \varphi \text{ ist} \\ 0 & \text{sonst} \end{cases}$
$(a_{ij})_{i,j}^{s,g}$	Matrix $\in \mathbb{R}^{s \times g}$ mit den Komponenten a_{ij} , wobei i von 1 bis s läuft und j von 1 bis g
$(f_k)_k^q$	Vektor $\in \mathbb{R}^{q \times 1}$, wobei k von 1 bis q läuft. Vektoren sind damit einspaltige Matrizen. Statt $(f_k)_k^q \in \mathbb{R}^{q \times 1}$ wird im Text $(f_k)_k^q \in \mathbb{R}^q$ geschrieben.
N	Anzahl Personen oder Fähigkeitsparameter
n	Anzahl Items
$X_{i,j}$	Zufallsvariable mit Funktionswerten $x_{i,j}$ (Antworten der Person i auf die Variable j ; Variable = Item; Funktionswert = Realisation)
$x_{i,j}$	Realisation der Zufallsvariable $X_{i,j}$ ($x_{i,j} \in \{0, 1\}$)
\mathbf{X}_i	Zufallsvektor mit Realisationen \mathbf{x}_i ($\mathbf{X}_i = (X_{i,1}, \dots, X_{i,n})^\top$)
\mathbf{x}_i	Vektor der Antworten der Person i (Realisation des Zufallsvektors \mathbf{X}_i ; $\mathbf{x}_i \in \{0, 1\}^n$; $\mathbf{x}_i = (x_{i,1}, \dots, x_{i,n})^\top$; n Anzahl Items)
\mathbf{X}	Zufallsmatrix mit Realisationen \mathbf{x} (Matrix der Antworten der N Personen auf die n Variablen; $\mathbf{X} = (\mathbf{X}_1, \dots, \mathbf{X}_N)^\top$)
\mathbf{x}	Matrix der Antwortvektoren (Realisation der Zufallsmatrix \mathbf{X} ; $\mathbf{x} \in \{0, 1\}^{N \times n}$; $\mathbf{x} = (\mathbf{x}_1, \dots, \mathbf{x}_N)^\top$)
Θ_i	Zufallsvariable mit Fähigkeiten θ_i als Realisation
θ_i	Realisation der Zufallsvariable Θ_i (Fähigkeiten; $\theta_i \in \mathbb{R}$)
Θ	Zufallsvektor mit den Fähigkeiten θ als Realisation.
θ	Realisation des Zufallsvektors Θ (Fähigkeiten; $\theta \in \mathbb{R}^N$; $\theta = (\theta_1, \dots, \theta_N)^\top$)
Δ_j	Item-Parameter-Zufallsvektor mit Werten δ_j
δ_j	Realisation des Zufallsvektors Δ_j (Itemparameter; $\delta_j \in \mathbb{R}^2$; $\delta_j = (\alpha_j, \beta_j)^\top$)
Δ	Item-Parameter-Zufallsmatrix mit Werten δ
δ	Realisation der Zufallsmatrix Δ (Itemparameter; $\delta \in \mathbb{R}^{n \times 2}$; $\delta = (\delta_1, \dots, \delta_n)^\top$)
α_j	Trennschärfeparameter für das j -te Item ($\alpha_j \in \mathbb{R}$)
β_j	Schwierigkeitsparameter für das j -te Item ($\beta_j \in \mathbb{R}$)
\mathbb{P}	gemeinsame Verteilung von $(\mathbf{X}, \Theta, \Delta)$
$P(\mathbf{x}_i \theta_i, \delta)$	$:= \mathbb{P}(\mathbf{X}_i = \mathbf{x}_i \Theta_i = \theta_i, \Delta = \delta)$
Z_k	Stützpunkte bei der näherungsweisen Integration (i. A. GH-Stützpunkte)
o	Verkettung von Funktionen

IRT	Item Response Theorie
ICC	Item Characteristic Curve (charakteristische Item-Kurve)
JMLE	Joint Maximum Likelihood Estimation
LL	Loglikelihood
ML	Maximum Likelihood
MMLE	Marginal Maximum Likelihood Estimation

Literatur

- Agresti, A. (1990). *Categorical Data Analysis*. New York: John Wiley.
- Agresti, A. (2002). *Categorical Data Analysis* (2. Aufl.). New York: John Wiley.
- Andersen, E. (1972). The Numerical Solution of a Set of Conditional Estimation Equations. *Journal of the Royal Statistical Society, 34, Series B*, 42-54.
- Baker, F. B. (2001). *The Basics of Item Response Theory*. University of Maryland: ERIC Clearinghouse on Assessment and Evaluation.
- Baker, F. B. & Kim, S.-H. (2004). *Item Response Theory: Parameter Estimation Techniques*. Basel: Dekker.
- Bauer, H. (2002). *Wahrscheinlichkeitstheorie* (5. Aufl.). Berlin: De Gruyter.
- Beyer, W. H. (1987). *Standard Mathematical Tables* (28th Aufl.). Boca Raton: CRC Press.
- Birnbaum, A. (1968). Some Latent Trait Models and their Use in Inferring an Examinee's Ability. In F. Lord & M. Novick (Hrsg.), *Statistical Theories of Mental Test Scores* (S. 397-479). Reading: Addison-Wesley.
- Bock, R. & Aitkin, M. (1981). Marginal Maximum Likelihood Estimation of Item Parameters: Application of an EM Algorithm. *Psychometrika, 46*, 443-459.
- Bock, R. & Lieberman, M. (1970). Fitting a Response Model for n Dichotomously Scored Items. *Psychometrika, 35*, 179-197.
- Cauchy, A. (1882-1974). *Cours d'analyse*. Paris: Gauthier-Villars et fils. (Oeuvres complètes d'Augustin Cauchy. Série 2, tome 3 , URL:<http://catalogue.bnf.fr/ark:/12148/cb302073189>)
- Cronbach, L. J. (1951). Coefficient Alpha and the Internal Structure of Tests. *Psychometrika, 16* (3), 297-334.
- Dempster, A., Laird, N. & Rubin, D. (1977). Maximum Likelihood from Incomplete Data via the EM Algorithm (with Discussion). *Journal of the Royal Statistical Society, Series B, 39*, 1-38.
- Dümbgen, L. (2009). *Lineare Modelle und Regression I-II*. Bern: Vorlesungsskript Universität Bern.
- eFunda. (2014). *Abscissas and Weights of Gauss-Hermite Integration*. Sunnyvale, CA: eFunda, Inc. (URL: <http://www.efunda.com/math/num/integration/findgausshermite.cfm>; konsultiert am 11. September 2014)
- Fischer, G. H. & Molenaar, I. W. (Hrsg.). (1995). *Rasch Models: Foundations, Recent Developments and Applications*. Berlin: Springer.
- Fox, J.-P. (2010). *Bayesian Item Response Modeling: Theory and Applications*. Berlin: Springer.
- Gautschi, W. (2004). *Orthogonal Polynomials: Computation and Approximation*. New York: Oxford University Press.
- Goodman, L. A. & Kruskal, W. H. (1954). Measures of Association for Cross Classifications. *Journal of the American Statistical Association* (49), 732-764.
- Guttman, L. (1950a). The Basis for Scalogram Analysis. In S. Stouffer, L. Guttman, E. Suchman, P. Lazarsfeld, S. Star & J. Clausen (Hrsg.), *Measurement and prediction. Studies in social psychology in World War II*. Princeton: Princeton University Press.
- Guttman, L. (1950b). The Principal Components of Scale Analysis. In S. Stouffer, L. Guttman, E. Suchman, P. Lazarsfeld, S. Star & J. Clausen (Hrsg.), *Measurement and prediction. Studies in social psychology in World War II*. Princeton: Princeton University Press.
- Guttman, L. (1950c). Relation of Scalogram Analysis to other Techniques. In S. Stouffer, L. Guttman, E. Suchman, P. Lazarsfeld, S. Star & J. Clausen (Hrsg.), *Measurement and prediction. Studies in social psychology in World War II*. Princeton: Princeton University Press.
- Hosmer, D. W. & Lemeshow, S. (2000). *Applied Logistic Regression*. New York: Wiley.
- International Association for the Evaluation of Educational Achievement (IEA). (2009). *TIMSS - Advanced 2008 Technical Report* (A. Arora, P. Foy, M. O. Martin & I. V. Mullis, Hrsg.). Boston: TIMSS & PIRLS International Study Center, Lynch School of Education, Boston College.
- Kendall, M. (1938). A new measure of rank correlation. *Biometrika, 30*, 81-93.

- Lawley, D. (1944). On problems connected with item selection and test construction. *Proceedings of the Royal Society of Edinburgh*, 61, 273-287.
- Lazersfeld, P. F. (1950). The Logical and Mathematical Foundation of Latent Structure Analysis. In S. Stouffer, L. Guttman, E. Suchman, P. Lazarsfeld, S. Star & J. Clausen (Hrsg.), *Measurement and prediction. Studies in social psychology in World War II*. Princeton: Princeton University Press.
- Lord, F. M. & Novick, M. R. (1968). *Statistical Theories of Mental Test Scores*. Reading (Mass.): Addison-Wesley.
- Mislevy, R. (1986). Bayes model estimation in item response models. *Psychometrika*, 51, 177-195.
- Müller, P. H. (1991). *Lexikon der Stochastik*. Berlin: Akademie Verlag.
- Moeschlin, O. (1998). *Bedingte Erwartungswerte und Verteilungen*. Hagen: Fernuniversität Hagen.
- Mokken, R. (1971). *A Theory and Procedure of Scale Analysis: With applications in political research*. Den Haag: Mouton.
- Neyman, J. & Scott, E. (1948). Consistent Estimates Based on Partially Consistent Observations. *Econometrica*, 16, 1-32.
- OECD. (2009). *PISA 2006 - Technical Report*. Paris: OECD Publishing.
- Perline, R., Wright, B. D. & Wainer, H. (1979). The Rasch Model as Additive Conjoint Measurement. *Applied Psychological Measurement*, 3 (2), 237-255.
- Pfanzagl, J. (1994). *Parametric statistical theory*. Berlin: Walter de Gruyter.
- R Development Core Team. (2006). *R: A Language and Environment for Statistical Computing*. Wien: R Foundation for Statistical Computing; URL <http://www.R-projekt.org/>.
- Rasch, G. (1980). *Probabilistic Models for some Intelligence and Attainment Tests*. Chicago: The University of Chicago Press. (Erweiterte Auflage der Ausgabe des Danish Institute for Educational Research (1960) in Kopenhagen)
- Rigdon, S. & Tsutakawa, R. (1983). Parameter estimation in latent trait models. *Psychometrika*, 48, 567-574.
- Rizopoulos, D. (2006, November). ltm: An R Package for Latent Variable Modeling and Item Response Theory Analysis. *Journal of Statistical Software*, 17 (5).
- Samejima, F. (1969). Estimation of Latent Ability using a Response Pattern of Graded Scores. *Psychometrika Monograph Supplement*, 17.
- Samejima, F. (1997a). Departure from normal Assumptions: a Promise for future Psychometrics with substantive mathematical Modelling. *Psychometrika*, 62 (4), 471-493.
- Samejima, F. (1997b). Graded Response Model. In W. J. van der Linden & R. K. Hambleton (Hrsg.), *Handbook of Modern Item Response Theory* (S. 85-100). Berlin: Springer.
- Stegmüller, W. (1970). *Probleme und Resultate der Wissenschaftstheorie und Analytischen Philosophie: Erfahrung, Festsetzung, Hypothese und Einfachheit in der wissenschaftlichen Begriffs- und Theorienbildung* (Bd. 2 - Theorie und Erfahrung). Berlin: Springer. (Studienausgabe Teil A)
- Swaminathan, H. & Gifford, J. A. (1982). Bayesian estimation in the Rasch model. *Journal of Educational Statistics*, 7, 175-191.
- Swaminathan, H. & Gifford, J. A. (1985). Bayesian Estimation in the Two-Parameter Logistic Model. *Psychometrika*, 50 (3), 349-364.
- van der Linden, W. J. & Hambleton, R. K. (Hrsg.). (1997). *Handbook of Modern Item Response Theory*. New York: Springer.
- Wu, J. (1983). On the Convergence Properties of the EM Algorithm. *The Annals of Statistics*, 11 (1), 95-103.