

**SOZIALWISSENSCHAFTLICHES INSTITUT DER BUNDESWEHR**  
**Berichte** **Heft 9**

*Paul Schneider*

**Q U E R 3**

**Ein Programmsystem zur Deskriptivstatistik**

**München, Juni 1978**

Alle Rechte vorbehalten  
© 1978  
ISSN 0342-2569-9  
Druck: SOWI, München

Sozialwissenschaftliches  
Institut der Bundeswehr  
Schleißheimer Str. 418  
8000 München 45  
Tel. 089 / 251 702 1

Inhaltsverzeichnis	Seite
<b>1. EINLEITUNG</b> .....	<b>1</b>
<b>2. KURZER PROBLEMAUFRISS</b> .....	<b>2</b>
<b>3. DEFINITORISCHE ABGRENZUNGEN</b> .....	<b>8</b>
<b>4. PROGRAMMBESCHRÄNKUNGEN</b> .....	<b>10</b>
<b>5. KOMPATIBILITÄT FÜR VERSCHIEDENDE RECHENANLAGEN</b> .....	<b>12</b>
<b>6. ZUM PROGRAMMAUSDRUCK</b> .....	<b>12</b>
<b>7. EINGABEBESCHREIBUNG</b> .....	<b>17</b>
<b>7.1. Die verschiedenen Kartendecks</b> .....	<b>17</b>
<b>7.2. Abfolge der Kartendecks</b> .....	<b>18</b>
<b>7.3. Programmsteuerkarten</b> .....	<b>22</b>
<b>7.4. Verarbeitung von Missing Data</b> .....	<b>30</b>
<b>7.5. Datenkomprimierung</b> .....	<b>34</b>
<b>7.6. Gruppenbildung</b> .....	<b>35</b>
<b>7.7. Varianzanalyse und Scheffé-Tests</b> .....	<b>41</b>
<b>7.8. Zusammenfassung der Rohwerte         zu Klassen</b> .....	<b>45</b>
<b>7.9. Prüfung auf Normalverteilung</b> .....	<b>47</b>
<b>7.10. Mehrere Rechenläufe</b> .....	<b>49</b>
<b>7.11. Variable Labels</b> .....	<b>54</b>
<b>8. PROGRAMMVERSION A</b> .....	<b>56</b>
<b>9. WEITERGABE DES PROGRAMMS</b> .....	<b>58</b>
<b>Literaturverzeichnis</b> .....	<b>59</b>
<b>Anhang A: Auflistung des Quellenprogramms QUER3</b>	
<b>Anhang B: Auflistung des Quellenprogramms TEXT</b>	
<b>Anhang C: Missing Data Verarbeitung mittels T-Format</b>	
<b>Anhang D: Reduzierung des Kernspeicherbedarfs</b>	

## VORWORT

1976 hat das SOWI mit Heft 6 der Berichtsreihe erstmalig eine eigene Programmentwicklung im Bereich der Datenverarbeitung vorgelegt. Die große Nachfrage nach dem Itemanalyseprogramm – zum jetzigen Zeitpunkt wird gerade die zweite Auflage vorbereitet –, hat uns in der Absicht bestärkt, auch weitere Grundlagenentwicklungen als Ergebnis der praktischen Arbeit zu veröffentlichen.

Das als QUER 3 in diesem Heft präsentierte Programm zur Deskriptivstatistik dürfte vor allem für jene empirischen Sozialforscher von Interesse sein, die möglichst viele Informationen mit Hilfe eines einzigen Programms übersichtlich und leicht verarbeitbar erhalten möchten.

München, Juni 1978

Prof. Dr. Ralf Zoll

## 1. EINLEITUNG

Das hier vorgestellte Programm QUER3 ermittelt mehrdimensionale Häufigkeitsverteilungen, die in Form sog. Kreuztabellen dargestellt werden und errechnet die in den Sozialwissenschaften gebräuchlichsten Teststatistiken.

Da speziell für diesen Anwendungsbereich bereits eine Reihe erprobter und bewährter Programme existieren, von denen hier nur die SPSS-Routine CROSSTABS genannt werden soll, stellt sich die Frage, warum bei der heute bereits vorhandenen Unüberschaubarkeit des Marktes an Datenverarbeitungsprogrammen für statistische Zwecke ein weiteres Programm angeboten wird.

QUER3 entstand aus dem Bestreben, möglichst viele Informationen durch ein einziges Programm aus dem Datenmaterial herauszuziehen und die Ergebnisse so übersichtlich wie möglich darzustellen. Seine Vorteile liegen insbesondere in folgenden Bereichen:

- Die unabhängige Variable, die sich in den Tabellenzeilen manifestiert, kann in Form einer einzigen Variablen gegeben sein (zweidimensionale Häufigkeitsverteilung). Darüber hinaus lassen sich mittels spezieller Programmsteuerkarten die Informationen von maximal 10 unabhängigen Variablen zu *einer* aggregierten unabhängigen Variablen vereinen, ohne daß, wie bei CROSSTABS automatisch sämtliche Kombinationen der ursprünglichen unabhängigen Variablen in Tabellenform ausgedruckt werden (Näheres hierzu s. Kap. 2 Kurzer Problemaufriß und 7.6. Gruppenbildung).
- Für jede Tabelle lassen sich über die vom Benutzer definierten Gruppierungen einfache Varianzanalysen (feste Effekte) mit F-Tests auf Gleichheit der Mittelwerte berechnen. Neben den gängigen Signifikanztests wird außerdem die Praktische Signifikanz als Anteil vorhersagbarer Varianz ermittelt (s. Kap 7.7. Varianzanalyse).
- Um festzustellen, durch welche Mittelwerte ein statistisch signifikantes Ergebnis des F-Tests zustandekommt, können Scheffé-Tests auf paarweise Mittelwertvergleiche durchgeführt werden (s. Kap. 7.7. Varianzanalyse).

- Für jede Variable lassen sich Schiefe und Exzeß als globale Hinweise auf das Vorliegen einer Normalverteilung ermitteln (s. Kap. 7.9. Prüfung auf Normalverteilung).
- QUER3 ermöglicht auch einfache Häufigkeitsauszählungen.
- Sofern ein File angelegt wird, das alpha-numerische Bezeichnungen für alle Variablen (sog. Variable Labels) enthält, erfolgt der Ausdruck der Labels automatisch. Das Programm zur Erstellung des Textfiles ist im Anhang ausgedruckt. Existiert kein Textfile, können Variable Labels bei jedem Rechenlauf in Form sog. Textkarten eingelesen werden und erscheinen dann ebenfalls im Programmausdruck.

## 2. KURZER PROBLEMAUFRISS

Der erste Schritt bei der statistischen Analyse eines Datensatzes besteht meist in einer einfachen Häufigkeitsauszählung aller relevanten Variablen. Sie dient der Beschreibung der Stichprobe und ermöglicht Aussagen über die Verteilung der erhobenen Merkmale. Kennwerte der zentralen Tendenz, etwa das arithmetische Mittel oder der Median, sowie Variabilitätsmaße liefern weitere Aufschlüsse, sofern das jeweilige Datenniveau deren Ermittlung zuläßt.

Für die Überprüfung der einer Untersuchung zugrunde liegenden Hypothesen reichen aber diese eindimensionalen Beschreibungen nicht aus. Die in einer Stichprobe erhobenen Merkmale sind vielmehr miteinander in Beziehung zu setzen. Jeweils zweiseitige Zusammenhänge lassen sich am einfachsten durch sog. Kreuztabellen ermitteln, bei denen die Häufigkeiten der Merkmalsausprägung einer Variablen (A) in Abhängigkeit von den Merkmalsausprägungen einer anderen Variablen (B) ausgezählt werden. Man spricht hier auch von „zweidimensionalen“ Häufigkeiten.

Als Beispiel sei hier der (fiktive) Zusammenhang zwischen den Rohwerten in einem Leistungstest, wobei der Einfachheit halber nur eine Rohwertverteilung von 0 bis 5 Punkte angenommen werden soll, und der Schulbildung gezeigt. Tabellarisch ergibt sich folgendes Bild:

Tabelle 1: Rohwerteverteilung in einem (fiktiven) Leistungstest, aufgeschlüsselt nach der Schulbildung der Probanden.

Schulbildung	Rohwerte im Leistungstest						Summe
	0	1	2	3	4	5	
Hauptschule	0	34	10	12	19	13	88
Mittelschule	0	24	9	12	9	17	71
Gymnasium	0	12	14	10	16	23	75
	0	70	33	34	44	53	234

Die letzte Zeile dieser Tabelle zeigt die Verteilung der Werte im Leistungstest für die gesamte Stichprobe. Sie besteht im Beispiel aus 234 Personen. Davon erreichten 70 Probanden einen Leistungswert von 1, 33 brachten es auf einen Wert von 2 usw. Die Zahlen im eingerahmten Rechteck geben die Verhältnisse für die drei Untergruppen – Hauptschul-, Mittelschulabsolventen und Gymnasiasten – wieder. Den Leistungswert von 5 erreichten demnach 13 Hauptschüler, 17 Mittelschüler und 23 Gymnasiasten, insgesamt 53 Personen. Die Gruppe der Hauptschüler umfaßt 88 Probanden, die Zahl der Mittelschüler beträgt 71 und die Gymnasiasten belaufen sich auf 75 Individuen.

Die Häufigkeiten in den Zeilen vor dem Doppelstrich stellen also nichts anderes dar als einfache Häufigkeiten in den Kategorien der Variablen Leistungstest bezogen auf die jeweilige Untergruppe. Das Vorgehen bei der zweidimensionalen Häufigkeitsauszählung läßt sich demnach folgendermaßen beschreiben: aus der gesamten Stichprobe werden anhand einer Klassifikationsvariablen  $k$  Gruppen (bzw. Unterstichproben) gebildet. Für jede dieser  $k$  Gruppen sind dann die Häufigkeitsverteilungen für eine bestimmte Zahl  $m$  von sog. abhängigen Variablen zu ermitteln. Die Klassifikationsvariable, die für die Zusammensetzung der Gruppen maßgebend ist, wird im weiteren als „gruppenbestimmende Variable“ bezeichnet. Sie stellt gewissermaßen die unabhängige Variable dar. Die Zuweisung eines

Probanden bzw. Falles zu einer der  $k$  Gruppen erfolgt anhand seiner Merkmalsausprägung bzw. seines Scores in der gruppenbestimmenden Variablen. Dieses einfache Beispiel einer zweifachen Häufigkeitsauszählung läßt sich auf beliebig viele Dimensionen erweitern. Angenommen, es interessiert nicht nur die Verteilung der Leistungswerte bezogen auf die drei Schultypen, sondern darüber hinaus soll noch nach dem Geschlecht der Probanden aufgefächert werden. In diesem Fall haben wir es mit einer dreidimensionalen Häufigkeitsverteilung zu tun (Beispiel 2). Da auf dem Papier einigermaßen übersichtlich nur zwei Dimensionen darstellbar sind, müssen die Ergebnisse in zwei Tabellen untergebracht werden: die eine Tabelle zeigt die Verteilung der Leistungsdaten, getrennt nach dem Schultyp für die männlichen Probanden, die andere für die weiblichen. Im Beispiel sehen sie folgendermaßen aus:

Tabelle 2: Rohwerteverteilung der männlichen Probanden in einem fiktiven Leistungstest, getrennt nach der Schulbildung

Schulbildung	Rohwerte im Leistungstest						Summe
	0	1	2	3	4	5	
Hauptschule	0	23	4	3	13	8	51
Mittelschule	0	13	6	7	7	11	44
Gymnasium	0	2	6	4	7	11	30
	0	38	16	14	27	30	125

Tabelle 3: Rohwerteverteilung der weiblichen Probanden in einem fiktiven Leistungstest, getrennt nach der Schulbildung

Schulbildung	Rohwerte im Leistungstest						Summe
	0	1	2	3	4	5	
Hauptschule	0	11	6	9	6	5	37
Mittelschule	0	11	3	5	2	6	27
Gymnasium	0	10	8	6	9	12	45
	0	32	17	20	17	23	109

In der Stichprobe befanden sich demnach 125 Männer und 109 Frauen. Die Summen der einander entsprechenden Elemente von Tabellen 2 und 3 ergeben die Werte der Tabelle 1.<sup>1</sup>

Zum Abschluß noch ein Beispiel für eine vierdimensionale Häufigkeitsverteilung (Beispiel 3): Die Testleistungen, aufgeschlüsselt nach der Schulbildung und dem Geschlecht der Probanden sollen unter der Frage betrachtet werden, ob die jeweilige Person ein hohes oder niedriges Maß an Testangst aufweist. Die Bildung der Substichproben erfolgt hier durch drei gruppenbestimmende Variablen, den Schulabschluß, das Geschlecht und die Testangst, die auf einer (fiktiven) Skala 0 bis maximal 12 Punkte betragen kann. Niedrige Angstpegel seien durch Skalenscores von 0 bis 4, hohe Angstausprägungen durch Werte von 8 bis 12 definiert.

Wollte man nun jeden einzelnen Testangst-Score als Grundlage für eine Häufigkeitsauszählung heranziehen, dann wären insgesamt 20 Tabellen zu erstellen, nämlich je eine Tabelle Schulbildung gegen Leistungstest für männliche Probanden mit dem Testangstwert 0,1,2,3,4,8,9,10,11,12, also insgesamt 10 Verteilungen, sowie die gleiche Anzahl für weibliche Probanden. Faßt man dagegen die Angstscores 0 bis 4 („niedrige“ Testangst) sowie 8 bis 12 („hohe“ Testangst) jeweils zusammen, so reduziert sich der erforderliche Aufwand auf 4 Tabellen:

---

Testangst		niedrig		hoch	
		männlich	weiblich	männlich	weiblich
Geschlecht	Haupt.	a	b	c	d
	Mittel.				
Schulbildung	Gymnas.				

---

In den SPSS-Routinen FASTABS und CROSSTABS werden automatisch so viele zweidimensionale Tabellen erstellt, wie durch die Hierarchie der gruppenbildenden Variablen und deren Modalitäten erforderlich sind. Die PROCEDURE-Karte TABLES=LEISTG BY SCHULE BY SEX BY ANGST hätte im Beispiel 3 die Tabellen a bis d zur Folge (sofern die Testangst-Scores zu den beiden Gruppen zusammengefaßt werden). Im Programm

---

<sup>1</sup> Dies trifft jedoch nur dann zu, sofern die gruppenbestimmenden Variablen keine Missing Data aufweisen.

QUER3 besteht dieser Automatismus nicht. Jede gewünschte Gruppenbildung muß hier einzeln auf sog. Gruppenkarten spezifiziert werden. Im Beispiel 3 sind aus den gruppenbildenden Variablen Schulbildung, Geschlecht und Testangst drei Substichproben zu bilden, die sich folgendermaßen zusammensetzen: <sup>1</sup>

	Gruppe 1	Gruppe 2	Gruppe 3
Tabelle a)	HS, m, TA niedr.	MI, m, TA niedr.	GY, m, TA niedr.
Tabelle b)	HS, w, TA niedr.	MI, w, TA niedr.	GY, w, TA niedr.
Tabelle c)	HS, m, TA hoch	MI, m, TA hoch	GY, m, TA hoch
Tabelle d)	HS, w, TA hoch	MI, w, TA hoch	GY, w, TA hoch

Für jede Tabellenform a bis d sind eigene Programmsteuerkarten erforderlich. Sie definieren die jeweiligen Auswertungsschritte. Im folgenden wird hier von *Rechenläufen* gesprochen. In einem Rechenlauf werden alle m Variablen gegen die jeweiligen Substichproben ausgezählt. Die bildlichen Darstellungen zeigen diesen Sachverhalt verkürzt für eine der m Variablen, die als „Testleistung“ bezeichnet wurde.

Selbstverständlich lassen sich mit diesen drei gruppenbildenden Variablen auch andere Substichproben definieren. Bei den Tabellen e bis g handelt es sich um zwei Gruppen, die folgende Tabellenbilder ergeben (Beispiel 4):

	Gruppe 1	Gruppe 2
Tabelle e)	TA niedr., HS, m	TA hoch, HS, m
Tabelle f)	TA niedr., MS, m	TA hoch, MS, m
Tabelle g)	TA niedr., GY, m	TA hoch, GY, m

Geschlecht	Schulbildung	männlich		
		Hauptschule	Mittelschule	Gymnasium
Testangst	Leistung	e	f	g
	niedr.			
hoch				

<sup>1</sup> Abkürzungen: HS = Hauptschüler; MS = Mittelschüler, GY = Gymnasiasten; m = männlich, W = weiblich; TA hoch = Testangst hoch; TA niedr. = Testangst niedrig

Die Ergebnisse des Beispiels 3, für die insgesamt vier Arten von Tabellen erforderlich waren, lassen sich durch entsprechende Gruppenbildung zu einer einzigen Tabelle vereinigen, die dann etwa folgendermaßen aussähe (Beispiel 5):

Gruppe	Zusammensetzung der Gruppe	Testleistung
1	HS, m, TA niedr.	
2	MS, m, TA niedr.	
3	GY, m, TA niedr.	
4	HS, w, TA niedr.	
5	MS, w, TA niedr.	
6	GY, w, TA niedr.	
7	HS, m, TA hoch	
8	MS, m, TA hoch	
9	GY, m, TA hoch	
10	HS, w, TA hoch	
11	MS, w, TA hoch	
12	GY, w, TA hoch	

Wir haben es hier mit 12 Gruppen zu tun (die Zahl der vom Programm akzeptierten Gruppen beträgt maximal 20). Für jede der  $m$  spezifizierten Variablen – in den Beispielen haben wir nur jeweils die Variable Testleistung betrachtet – würden in diesem Fall die Häufigkeitsverteilungen über die 12 Gruppen ermittelt. Allerdings ergäbe die Gesamtsumme jeder Tabelle (wie auch in den Beispielen 3 und 4) nicht mehr die Zahl der Probanden des Datensatzes, da bei den genannten Gruppierungen alle Probanden mit Testangstwerten im Bereich von 5 - 7 Punkte außer Betracht bleiben.

Anzumerken ist, daß die  $k$  Gruppen *echte* Teilmengen des Gesamtdatensatzes darstellen müssen. Die Substichproben sind also so zu definieren, daß jeder Proband eindeutig einer oder keiner dieser Gruppen zugeordnet werden kann. Eine ausführliche Beschreibung der für die Gruppenbildung erforderlichen Programmsteuerkarten erfolgt im Kapitel 7.6. „Gruppenbildung“.

Neben den absoluten Häufigkeiten, mit denen die Kategorien einer Variablen auftreten, berechnet das Programm die arithmetischen Mittel und Standardabweichungen in den  $k$  Gruppen sowie in der Gesamtstichprobe. Die Entscheidung über die Zulässigkeit der genannten Parameter aufgrund des Datenniveaus der jeweiligen Variablen bleibt dabei dem Benutzer überlassen.

### **3. DEFINITORISCHE ABGRENZUNGEN**

Zum besseren Verständnis der Eingabebeschreibung für das Programm werden die dort verwendeten Begriffe im folgenden kurz beschrieben.

#### **Gruppe bzw. Unterstichprobe**

Der gesamte Datensatz wird in sog. Gruppen untergliedert. Für jede dieser  $k$  Gruppen werden Häufigkeitsverteilungen für alle im Eingabeformat benannten Variablen erstellt. Im Programmausdruck sind die Gruppen durch die Tabellenzeilen repräsentiert. Synonym zum Begriff Gruppe werden die Termini Unter- bzw. Substichprobe verwandt.

#### **Gruppenbestimmende Variable(n)**

Die Variable(n), mittels derer die Zuweisung eines Falles (Untersuchungseinheit, Proband etc.) zu einer der  $k$  Gruppen erfolgen soll, wird (werden) als „gruppenbestimmende“ Variable(n) bezeichnet. Die Gruppenzuteilung wird anhand der Merkmalsausprägungen des jeweiligen Falles in der (den) gruppenbestimmenden Variablen vorgenommen. Sie stellen gewissermaßen die unabhängigen Variablen dar.

#### **Abhängige Variable(n)**

Damit sind diejenigen Variablen gemeint, deren Häufigkeitsverteilungen über die Gruppen hinweg zu ermitteln sind. Die im Eingabeformat spezifizierten  $m$  Variablen setzen sich somit aus  $k$  gruppenbestimmenden und  $m$  abhängigen Variablen zusammen.

#### **Spannweite**

Eine Variable umfaßt  $j$  Merkmalsausprägungen, die durch positive oder negative Zahlen repräsentiert sind. Der Zahlenbereich zwischen minimaler und maximaler Merkmalsausprägung wird als „Spannweite“ bezeichnet.

Sie errechnet sich folgendermaßen: Maximaler Zahlenwert, den die jeweilige Variable im gesamten Datensatz annimmt minus minimaler Zahlenwert plus 1. Die Spannweite wird vom Programm ermittelt, braucht also nicht eingegeben zu werden. Sofern die Kategorien der abhängigen Variablen zu jeweils  $p$  Werte umfassende Klassen zusammengefaßt werden, läßt sich programmtechnisch die Spannweite um den Faktor  $p$  verkleinern. Es ergibt sich dann folgender Berechnungsmodus:

$(\text{Größter Wert} - \text{kleinster Wert})/p + 1$ .

### Rechenlauf

Ein Rechenlauf beinhaltet die Häufigkeitsauszählungen aller abhängigen Variablen in den vom Benutzer erstellten Gruppen. Sollen die abhängigen Variablen gegen eine neue Gruppierung bzw. sollen neue abhängige Variable gegen die gleiche Gruppierung quergezählt werden, sind weitere Rechenläufe erforderlich.

Diese lassen sich als sog. Kettenjobs durchführen.

### Eingabedaten – Ausgangsdaten

In Verbindung mit der Missing-Data Verarbeitung werden die Daten auf einen Platten-Zwischenspeicher (Scratch File) gebracht, von dem sie dann zur Erstellung der jeweiligen Auswertungsschritte in den Kernspeicher übertragen werden. Als „Ausgangsdaten“ wird die Originaldatei bezeichnet, wie sie entweder als Lochkartensatz vorliegt oder aber auf einem Band bzw. einem Plattenbereich (Permanent File) gespeichert ist. Der Begriff „Eingabedaten“ bezieht sich auf die Daten des Scratch File.

Zwei Fälle lassen sich unterscheiden:

- a. Eingabedatei und Ausgangsdatei sind identisch.

Die Originaldaten werden mit dem vom Benutzer erstellten Format für die Eingabedaten gelesen und zwischengespeichert. Hier ist nur *eine* Formatanweisung erforderlich: Format für Eingabedaten (s. Eingabebeschreibung).

- b. Die Ausgangsdaten werden in komprimierter Form, d.h. ohne zu überspringende Lochkartenspalten bzw. Lochkarten zwischengespeichert (s. Kap. 7.5. Datenkomprimierung). Hier sind beide Dateien mittels zweier verschiedener Formatanweisungen zu definieren: Format für

Eingabedaten (komprimierte Daten) und Format für Ausgangsdaten (Originaldaten).

### Default

Sofern der Benutzer einen Parameter, der einen Defaultwert aufweist, nicht spezifiziert, also ungelocht läßt, tritt automatisch dieser Defaultwert in Kraft.

## 4. PROGRAMMBESCHRÄNKUNGEN

- Die Zahl der Variablen (gruppenbestimmende und abhängige Variablen) darf maximal 100 betragen.
- Die Zahl der Gruppen ist auf maximal 20 begrenzt.
- Zur Definition der k Gruppen dürfen maximal 10 gruppenbestimmende Variable herangezogen werden.
- Sofern die Daten auf Platten- oder Bandspeicher vorliegen, ist dafür zu sorgen, daß die Recordlänge nicht größer als 80 ist. Die Daten auf dem externen Speicher müssen also sog. Lochkarten-Image aufweisen. Diese Beschränkung ist jedoch nur dann obligatorisch, wenn für jede Variable ein eigener Missing Data Code eingelesen wird (in diesem Fall werden die Missing Data Codes auf den Steuerkarten benannt, diese aber im Eingabeformat für die Daten gelesen).
- Die maximal zulässige Spannweite richtet sich nach drei Parametern:
  - a. Konstante KONST (s.u.)
  - b. Zahl der Gruppen (auf Parameterkarte mit IGR bezeichnet)
  - c. Zahl der Variablen (auf Parameterkarte mit MV bezeichnet).
 Die zulässige Spannweite für die MV Variablen beträgt:  
 Zulässige Spannweite =  $KONST / (MV \cdot IGR)$ , darf jedoch 200 nicht überschreiten.  
 Variable, deren Spannweite größer als die zulässige Spannweite ist, werden nicht ausgezählt. Statt dessen erscheint die Nachricht:  
 VARIABLE ... SPANNWEITE ZU GROSS.  
 Der Faktor KONST ist auf 40000 gesetzt. Damit wird ein Kernspeicherfeld von 40000 Worten definiert. Sofern dadurch die Speicherkapazität

überschritten wird, ist die Dimensionierung des Feldes IH (Programmkarte mit Label QUE00110) zu erniedrigen (z.B. auf DIMENSION IH (20000)). Die Programmkonstante NDIM (Label QUE00310) ist auf den gleichen Wert zu reduzieren (im Beispiel: NDIM = 20000).

- Die Feldbreite der eingelesenen Variablen darf höchstens 4 betragen. Die Variablen dürfen somit aus maximal 4 Ziffern bestehen (s. Kapitel Missing Data Verarbeitung. Das maximal zulässige Eingabeformat beträgt demnach F4.0 bzw. F4.1, F4.2).
- Variablen mit Nachkommastellen werden im Ausdruck der Häufigkeiten auf die vor dem Komma stehende Zahl abgerundet. Die Berechnung von Mittelwert und Standardabweichung erfolgt aber anhand der Originalwerte.  
Falls sich die Meßwerte nur in den Nachkommastellen voneinander unterscheiden (falls sie beispielsweise alle im Bereich 2.00 bis 2.99 liegen), sind sie vor dem Programmablauf mit einem Faktor 10 bzw. 100 zu multiplizieren. (Diese Datenmanipulation kann in der Subroutine DATV vorgenommen werden). Mittelwerte und Standardabweichungen vergrößern sich damit ebenfalls um den Faktor 10 bzw. 100, Schiefe und Exzeß bleiben unbeeinflusst.
- Variable mit negativen Scores werden akzeptiert. Bei den verschiedenen Rechengängen und beim Ausdruck der Häufigkeitsverteilungen wird das Vorzeichen berücksichtigt.
- Nach einem abgeschlossenen Rechenlauf erfolgt ein Programmrücksprung zum Lesebefehl der Parameterkarte. Aus diesem Grund lassen sich beliebig viele Rechenläufe in ein und demselben Job durchführen. Enthält eine Parameterkarte keine Angaben über die Zahl der Fälle (NP) und die Zahl der Variablen (MV), so wird damit das Jobende signalisiert. Vor der Abschlußkarte (end-of-file-Karte) muß deshalb eine Leerkarte liegen. Andernfalls wird die End-of-file-Karte als Parameterkarte interpretiert, was einen Jobabbruch durch das System mit entsprechender Meldung hervorruft. Die bis hierhin berechneten Ergebnisse werden dadurch allerdings nicht beeinflusst.

## 5. KOMPATIBILITÄT DES PROGRAMMS FÜR VERSCHIEDENE RECHENANLAGEN

Um das Programm für Rechenanlagen der verschiedenen Hersteller kompatibel zu gestalten, werden zur Programmierung nur Standard FORTRAN IV Statements verwendet. Es ist auf einer IBM 370/148 ausgetestet, der Übertragung auf andere Anlagen dürften aber keine größeren Schwierigkeiten entgegenstehen. Erfahrungen mit dem Programm liegen auch für CDC-Rechner vor. Bei diesen Maschinentypen, die eine andere Byte- und Wortlänge verwenden als IBM-Rechner, ist darauf zu achten, daß die Statements REAL\*8 und INTEREGER\*2 aus dem Hauptprogramm sowie der Subroutine SCHEX entfernt werden. Die Missing-Data-Verarbeitung läßt sich hier wesentlich vereinfachen, da Blanks im F-Format intern mit -0. verschlüsselt werden. Die Abfrage auf -0. erfolgt mit Hilfe der SIGN Funktion. Dadurch erübrigt sich einer der beiden Plattenzwischenpeicher (s. Kap. 8 Programmversion A).

## 6. ZUM PROGRAMMAUSDRUCK

Für jede der in die Auswertung einbezogenen  $m$  Variablen werden die Häufigkeiten für jede einzelne Kategorie bzw. Modalität der Variablen in den  $k$  Unterstichproben sowie in der Gesamtstichprobe ermittelt. Die Zeilen der Tabellen stellen somit die Gruppen, die Spalten die jeweiligen Kategorien der auszuzählenden Variablen dar. Die Zahl der die Substichproben konstituierenden Fälle sowie das arithmetische Mittel und die Standardabweichung für die Gruppe bilden den Anfang der ersten Zeile einer Gruppe. Diese sind zur leichteren Identifikation der sich darin befindlichen Zahlen durch die Buchstaben A bis E gekennzeichnet. Zeile A gibt die absoluten Häufigkeiten wieder, während die Zeilen B, C und D relative Häufigkeiten enthalten: Zeile B ist auf die jeweilige Gruppensumme, Zeile C auf die betreffende Spaltensumme und Zeile D auf die Tabellengesamtsumme prozentuiert. Die Ausgabe der genannten Prozentwerte erfolgt auf Wunsch des Benutzers, wobei die einzelnen Berechnungsmodi beliebig miteinander kombiniert werden können. Zeile E enthält die für die Berechnung des Chi-Quadrat-Wertes ermittelten Erwartungshäufigkeiten. Sie werden nur ausgedruckt, sofern sich der Chi-Quadrat-Test anwenden läßt (s.u.).

Das Programm enthält eine Option zur Durchführung *einfacher Varianzanalysen* zwischen den vom Benutzer konstituierten Gruppen. Der Programmausdruck beinhaltet in diesem Fall zusätzlich folgende Kennwerte:<sup>1</sup>

Mittlere Abweichungsquadrate „zwischen“ und „innerhalb“;

Freiheitsgrade des Zählers und Nenners des F-Quotienten; F-Wert, Irrtumswahrscheinlichkeit alpha für F (in Prozent) sowie verbale Beschreibung des Prüfergebnisses: GLEICHHEIT MITTELWERTE = NEIN (sofern alpha kleiner/gleich 5 %) bzw. GLEICHHEIT MITTELWERTE = JA (bei Irrtumswahrscheinlichkeiten größer 5 %).

Praktische Signifikanz: Omega-Quadrat nach Hays.

Liegen nur zwei Gruppen vor, werden zusätzlich die Prüfstatistik T des T-Tests, sowie zur Schätzung der praktischen Signifikanz des Quadrat des punktbiserialen Korrelationskoeffizienten  $r_{pb}^2$  und das nach Levy berechnete Klassifikationsmaß U. ausgedruckt (s. Kap. Varianzanalyse).

Barlett-Test auf Gleichheit der Varianzen: Chi-Quadrat, Freiheitsgrade, Irrtumswahrscheinlichkeit sowie Prüfergebnis: GLEICHHEIT VARIANZEN = NEIN (Irrtumswahrscheinlichkeit kleiner/gleich 5 %) bzw. GLEICHHEIT VARIANZEN = JA (bei Irrtumswahrscheinlichkeit größer 5 %).

Unter folgenden Voraussetzungen wird der Chi-Quadrat-Test für Mehrfeldertafeln durchgeführt;

- (1) weniger als 20 % der Zellen sind mit Häufigkeiten kleiner oder gleich 5 besetzt,
- (2) die Zahl der Zeilen (Unterstichproben) ist kleiner oder gleich 10,
- (3) die Zahl der Spalten (Kategorien der auszuzählenden Variablen nach einer eventuellen Zusammenfassung einzelner Kategorien zu Klassen) ist kleiner oder gleich 10.

Sind diese Bedingungen erfüllt, so erscheinen im Programmausdruck:

errechneter Chi-Quadrat-Wert, Freiheitsgrade, Irrtumswahrscheinlichkeit alpha in Prozent, verbales Testergebnis:

GLEICHE VERTEILUNG IN GRUPPEN = JA (sofern alpha

<sup>1</sup> Verweise auf die Literatur zu den verschiedenen Kennwerten erfolgen im Kap. 7.7. Varianzanalyse. Für die weniger gebräuchlichen Koeffizienten sind dort auch Formeln wiedergegeben.

Die Irrtumswahrscheinlichkeiten (alpha) werden über die FUNKTION PRBF von Veldman errechnet.

größer 5 %) bzw. GLEICHE VERTEILUNG IN GRUPPEN  
= NEIN (bei Irrtumswahrscheinlichkeit kleiner/gleich 5 %).

Sofern Voraussetzungen (2) und (3) erfüllt sind, werden folgende Kontingenzmaße berechnet:

Kontingenzkoeffizient und korrigierter Kontingenzkoeffizient<sup>1</sup>, V nach Cramer<sup>2</sup>.

Als Maß für predictive association erscheinen die Koeffizienten lambda asymmetrisch und lambda symmetrisch<sup>3</sup>.

---

<sup>1</sup> Kontingenzkoeffizient  $C = \sqrt{\text{Chi}^2 / (N + \text{Chi}^2)}$

Maximaler Kontingenzk.  $C_{\max} = \sqrt{(L-1)/L}$

wobei: L = kleinerer der beiden Werte Z (Zahl der Zeilen)  
oder S (Zahl der Spalten)

Für quadratische Kontingenztafeln gilt  $L = Z = S$ .

Korrigierter Kontingenzkoeffizient  $C_{\text{kor}} = (C \cdot 100) / C_{\max}$

(s. Sachs, 1971, S. 372)

<sup>2</sup> V nach Cramer:  $V = \sqrt{\text{Chi}^2 / (N(L-1))}$

im Falle einer Vierfeldertafel gilt:  $V = \text{Phi} = r_{\text{pbis}}$  (punkt-biseriale Korrr.)

(s. Hays, 1970, S. 604 ff., Nie et al, 1975, S. 224)

<sup>3</sup> Mit diesen Indizes läßt sich ermitteln, wie stark sich die Fehlerwahrscheinlichkeit proportional verringert, wenn man weiß, in welche Kategorie der Variablen A ein Fall gehört und man aufgrund dieser Information schätzt, in welche Kategorie der Variablen B er wahrscheinlich fallen wird. Reduziert die Information über Kategorie A die Fehlerwahrscheinlichkeit nicht, so ergibt sich ein Lambda von 0. Ein Index von 1 besagt dagegen, daß der Schluß von Variable A auf Variable B fehlerfrei ist.

Asymmetrisches Lambda:

Lambda(a): aus der Zugehörigkeit zu einer der k Substichproben oder Gruppen wird auf die abhängige Variable geschlossen.

Lambda(b): aus der abhängigen Variablen wird auf die Zugehörigkeit zu einer der k Substichproben geschlossen.

Symmetrisches Lambda:

Lambda(ab) gibt Auskunft darüber, wie stark sich die Fehlerwahrscheinlichkeit proportional verringert durch die Kenntnis der Kategorie von Variable A oder Variable B (d.h. Substichprobe oder abhängige Variable). Lambda (ab) liegt immer zwischen Lambda(a) und Lambda(b).

(Formel s. Hays, 1970, S. 608 ff., Nie et al, 1975, S. 225)

Eine weitere Programmoption erlaubt die *Prüfung der Häufigkeitsverteilungen* (in den einzelnen Gruppen sowie in der Gesamtstichprobe) *auf Normalität* mittels der dritten und vierten Verteilungsmomente „Schiefe“ und „Exzeß“.

Der Ausdruck enthält folgende Daten:

Schiefe, T-Wert für die Schiefe, Irrtumswahrscheinlichkeit alpha sowie Prüfergebnis NORMAL (sofern alpha größer 5 %).

Exzeß, T-Wert für den Exzeß, Irrtumswahrscheinlichkeit sowie Prüfergebnis NORMAL (alpha größer 5 %) bzw. blank (alpha kleiner/gleich 5 %).

*Scheffé-Tests* für paarweise Mittelwertvergleiche lassen sich unter folgenden Bedingungen auf Wunsch des Benutzers berechnen:

- Zahl der Gruppen größer als 2 und kleiner/gleich 10,
- Ergebnis der Varianzanalyse mindestens auf dem 5 %-Niveau signifikant.

Die Darstellung der Testergebnisse erfolgt in Form einer Matrix mit k Zeilen (wobei k = Zahl der Gruppen) und k-1 Spalten. Die Matrix enthält die errechneten F-Werte für jeden paarweisen Mittelwertvergleich sowie bei Vorliegen von Signifikanz den Hinweis SIGNIF.

Tabelle 4 zeigt das eingangs besprochene Beispiel 1 (s. Tabelle 1) mit vollem Programmausdruck (sämtliche Prozentuierungsmöglichkeiten, Varianzanalyse, Verteilungscharakteristika Schiefe und Exzeß, Scheffé-Tests).

Wie bereits erwähnt, finden in einer Tabelle maximal 15 Scores der abhängigen Variablen Platz. Übersteigt die Spannweite den Wert 15, so sind zwei und mehr Tabellen erforderlich. In diesen Fällen werden Scores, die in der Gesamtstichprobe eine Häufigkeit von Null aufweisen, nicht ausgedruckt, was eine Platzeinsparung mit sich bringt. Enthält die abhängige Variable z.B. nur die Rohwerte 1 bis 5 und 99, so werden die zwischen 5 und 99 liegenden Scores, deren Häufigkeit dann Null beträgt, im Ausdruck nicht aufgeführt.

VARIABLE 8 LEISTUNGSTEST  
GRUPPE ANZAHL

GRUPPE	LEISTUNGSTEST	ANZAHL	MITTEL- WERT	STANDARD- ABWEICH.	HÄUFIGKEITEN				
					1	2	3	4	5
1 HAUPTSCHULE	A	88	2.63	1.534	34	10	12	19	13
	B	100.0			38.6	11.4	13.6	21.6	14.8
	C				48.6	30.3	35.3	43.2	24.5
	D				14.5	4.3	5.1	8.1	5.6
	E				26.3	12.4	12.8	16.5	19.9
2 MITTELSCHULE	A	71	2.80	1.600	24	9	12	9	17
	B	100.0			33.8	12.7	16.9	12.7	23.9
	C				34.3	27.3	35.3	20.5	32.1
	D				10.3	3.8	5.1	3.8	7.3
	E				21.2	10.0	10.3	13.4	16.1
3 GYMNASIUM	A	75	3.32	1.481	12	14	10	16	23
	B	100.0			16.0	18.7	13.3	21.3	30.7
	C				17.0	42.4	29.4	36.4	43.4
	D				5.1	6.0	4.3	6.8	9.8
	E				22.4	10.6	10.9	14.1	17.0
B		234	2.90	1.560	70	33	34	44	53
		100.0			29.9	14.1	14.5	18.8	22.6

VARIANZANALYSE. MAO(ZWISCHEN) = 10.28 MAO(INNERHALB) = 2.36 DF(ZAEHLER) = 2 DF(NENNER) = 231

F = 4.347 IRRTUMSWAHRSCHEINLICHKEIT (ALPHA) = 1.39 PROZENT

GLEICHHEIT MITTELWERTE = NEIN

BARLETT-TEST AUF GLEICHHEIT DER VARIANZEN. OMEGA-QUADRAT (HAYS) = 0.0278

CHI-QUADRAT = 16.154 DF = 8 IRRTUMSWAHRSCHEINLICHK. (ALPHA) = 4.09 PROZENT GLEICHE VERTEIL. IN GRUPPEN = JA

KONTINGENZKOEFFIZIENT C = 0.245 KORRIGIERTER KONTINGENZKOEFFIZIENT = 0.311

INDEX OF PREDICTIVE ASSOCIATION. LAMBDA(A) = 0.067 LAMBDA(B) = 0.096 LAMBDA(AB) = 0.081 V (CRAMER) = 0.186

GRUPPE	SCHIEFE/ T-WERT	ALPHA/ ERGEBNIS	EXZESS/ T-WERT	ALPHA/ ERGEBNIS	SCHEFFE-TEST PAARWEISE MITTELW.-VERGLEICHE KRIT F*(IGR-1) = 6.07	
					1	2
1 HAUPTSCHULE	0.24	0.634	-1.52	0.005		
2 MITTELSCHULE	0.92	NORMAL	-2.91	0.009	0.53	
3 GYMNASIUM	0.17	0.559	-1.56	0.015	8.27	4.13
	0.60	NORMAL	-2.68	0.000	SIGNIF	
	-0.28	0.674	-1.40			
	-0.99	NORMAL	-2.48			
	0.05	0.764	-1.53			
	0.30	NORMAL	-4.79			

Das Programm bietet die Möglichkeit, *einfache Häufigkeitsauszählungen* vornehmen zu lassen. In diesem Fall erscheinen in der Ausgabe die Häufigkeiten für den gesamten Datensatz, sowie Mittelwert und Standardabweichung (s. Häufigkeitstabelle: die letzten beiden Zeilen vor dem Doppelstrich). Die Verteilungsmaße Schiefe und Exzeß lassen sich auf Wunsch des Benutzers mitberechnen.

## 7. EINGABEBESCHREIBUNG

### 7.1. Die verschiedenen Kartendecks

Ein vollständiger Job enthält maximal folgende vier Kartendecks:

- a) Systemsteuerkarten (Job Control Karten). Zahl und Aufbau der Systemsteuerkarten sind abhängig vom jeweiligen Maschinentyp.
- b) Programmdeck. Mit diesem Begriff werden die Lochkarten, die die Programmstatements enthalten, bezeichnet.
- c) Programmsteuerkarten (s.u.)
- d) Datenkarten. Sie enthalten die Daten, für die die Programmläufe durchzuführen sind.

Systemsteuerkarten sind für jeden Job, Programmsteuerkarten für jeden Rechenlauf eines Jobs zu lochen und über den Kartenleser einzugeben. Für die Eingabe des Programmes bestehen mehrere Möglichkeiten:

1. Das Programm liegt in Lochkartenform vor (BCD-Code oder EBCDIC-Code) und wird ebenfalls eingelesen. Damit ist der FORTRAN-Compiler aufzurufen, der das Programm in Maschinenform übersetzt. Anschließend wird es in den Kernspeicher geladen und gestartet.  
Nachteil: QUER3 umfaßt ca. 1200 Lochkarten, die Einleseprozedur ist deshalb umständlich und zeitaufwendig. Das Übersetzen des Programms benötigt Maschinenzeit, die in der Regel vom Benutzer zu bezahlen ist.
2. Sofern mit QUER3 häufiger gearbeitet werden soll, empfiehlt es sich, es in absoluter (binärer) Form auf Band, oder falls es die Möglichkeiten des jeweiligen Rechenzentrums erlauben, auf Magnetplatte zu speichern. Damit entfällt der Compilierungsvorgang.

3. Es besteht ferner die Möglichkeit, das Programm in Lochkartenform auf Platte oder Band zu übernehmen. Damit erspart man sich zwar die Eingabe über den Kartenleser, nicht aber die Compilierungszeit.
4. Das Programmdeck läßt sich verkleinern, indem man sich eine Binärversion von QUER3 ausstanzen läßt. Es ist dann zwar für jeden Job über den Kartenleser einzugeben, der Compilierungsvorgang entfällt aber.

Ähnlich verhält es sich mit den Daten. Diese liegen entweder in Form von Lochkarten vor oder sind auf einer Platten-/Banddatei im BCD- oder EBCDI-Code gespeichert. **QUER3 ist nicht für Binärdaten ausgelegt!**

## 7.2. Abfolge der Kartendecks

Im folgenden wird der für einen Job erforderliche Aufbau der Lochkartendecks sowie deren Abfolge kurz beschrieben. Der Aufbau der Systemsteuerkarten wird für eine IBM 370/148 bzw. CYBER 175 demonstriert.

Fall 1: Programm und Daten werden über den Lochkartenleser eingegeben

- Abfolge
- a) Systemsteuerkarten
  - b) Programmkarten  
(evtl. 2. Teil der Systemsteuerkarten, s.u.)
  - c) Programmsteuerkarten
  - d) Datenkarten

IBM:

- a)  $\left\{ \begin{array}{l} //Job-Karte \\ // EXEC FORTHCLG \\ //FORT.SYSIN DD * \end{array} \right.$
- b) Programmkarten
- a)  $\left\{ \begin{array}{l} //GO.FT15F001 DD UNIT=..., VOL=SER..., \\ // DISP=OLD, DSN=Name der Datei für Variable \\ labels<sup>1</sup> \\ //GO.FT08F001 DD UNIT=SYSDA,SPACE=(...,...), \\ // DSN=&XXXX \\ //GO.FT09F001 DD UNIT=SYSDA,SPACE=(...,...), \\ // DSN=&YYYY \\ //GO.SYSIN DD * \end{array} \right.$

---

<sup>1</sup>Diese Karte ist nur erforderlich, falls die Textkarten von der externen Datei mit der logischen File-Nummer 15 gelesen werden.

- c) Programmsteuerkarten
- d) Datenkarten

- CYBER
- a) {
    - JOB-Karte
    - USER-Karte
    - CHARGE-Karte
    - FTN(A,T)
    - LGO.
    - EOR (End-of-record-Karte; Lochung 7/8/9  
in Spalte 1)
  - b) {
    - PROGRAM QUER3 (INPUT,OUTPUT,TAPE 5=INPUT,  
TAPE6=OUTPUT,TAPE15,TAPE8,TAPE9)<sup>1</sup>
    - Programmkarten
    - EOR
  - c) Programmsteuerkarten
  - d) Datenkarten

Fall 2: Das Programm wird über Karten, die Daten von einem externen Speicher eingegeben.

- IBM
- a) wie Fall 1
  - b) Programmkarten
  - a) {
    - ./GO.FT08F001 DD (weiter wie Fall 1)
    - //GO.FT09F001 DD (weiter wie Fall 1)
    - //GO.FT10F001 DD UNIT=...,VOL=SER=...,  
// DISP=OLD,DSN=Dateiname<sup>2</sup>
    - //GO.FT15 DD...(s. Fall 1)
    - //GO.SYSIN DD\*
  - c) Programmsteuerkarten

---

<sup>1</sup> Die PROGRAM-Karte ist bei CDC-Rechnern Bestandteil des Programms. Sie liegt am Anfang des Programmdecks, also vor der Karte mit dem Label QUE00010. Wird das Programm gespeichert, so ist die PROGRAM-Karte ebenfalls auf die Platte bzw. das Band zu bringen. Damit auch Daten von Platte/Band gelesen werden können, empfiehlt es sich, auf der PROGRAM-Karte auch TAPE10 zu spezifizieren. Sofern Programmversion A (s. Kap. 8) benutzt wird, entfällt TAPE9.

<sup>2</sup> Als Eingabeeinheit für Daten ist in diesem Fall auf der Parameterkarte (s.u.) eine 10 anzugeben.

CYBER

- a) {
  - JOB-Karte
  - USER-Karte
  - CHARGE-Karte
  - FTN(A,T)
  - GET(TAPE10=Dateiname)
  - LGO.
  - EOR
- b) {
  - PROGRAM QUER3 (INPUT, OUTPUT,  
 TAPE5=INPUT,TAPE6=OUTPUT,TAPE8,  
 TAPE9,TAPE10,TAPE15)<sup>1</sup>
  - Programmkarten
  - EOR
- c) Programmsteuerkarten

Fall 3: Das Programm befindet sich mit dem Label QUER3 in absoluter Form auf Platte oder Band (IBM: JOBLIB). Die Daten werden ebenfalls von Platte oder Band gelesen.

IBM

- //JOB-Karte
- //JOBLIB-Karte
- // EXEC PGM=QUER3
- //FT08F001 DD UNIT=SYSDA,SPACE=(...,),
- // DSN=&xxxx
- a) // FT09F001 DD UNIT=SYSDA,SPACE=(...,),
- // DSN=&yyyy
- //FT10F001 DD UNIT=..., VOL=SER=...,
- // DISP=OLD,
- // DSN=Dateiname der Einheit, die die Daten enthält.<sup>1</sup>
- //FT15F001 DD UNIT=...,VOL=SER=...,DISP=OLD,
- // DSN=Name der Datei für variable labels.<sup>1</sup>
- //FT06F001 DD SYSOUT=A
- //FT05 F001 DD \*
- c) Programmsteuerkarten

---

<sup>1</sup> Als Eingabeeinheit für Daten ist in diesem Fall auf der Parameterkarte (s.u.) eine 10 anzugeben.

CYBER

a) { JOB-Karte  
 USER-Karte  
 CHARGE-Karte  
 GET(QUER3)  
 GET(TAPE10=Dateiname)<sup>1</sup>  
 QUER3.  
 EOR

c) Programmsteuerkarten

(Über den weiteren Verlauf der einzugebenden Karten s. Kap. 7.10. Mehrere Rechenabläufe).

**Anmerkung:**

QUER3 enthält ein Unterprogramm, SUBROUTINE DATV, das zur Datenmanipulation verwendet werden kann. Die entsprechenden Programmkarten sind vom Benutzer zu erstellen und in die Subroutine, die sich am Ende des Programmdecks befindet, einzufügen.

Dieses Unterprogramm wird *vor* den Lesebefehlen der Programmsteuerkarten aufgerufen. Sofern das Programm und die Daten über Lochkarten zugeführt werden, verändert sich daher die Reihenfolge der Eingabe:

wenn SUBROUTINE DATV besetzt ist, müssen die Datenkarten stets *vor* der ersten *Programmsteuerkarte* liegen.

(Liegt das Programm bereits auf einem externen Speicher, ist die Subroutine nicht mehr zugänglich. Erforderliche Datenmanipulationen müssen in diesem Fall in einem von QUER3 getrennten Programm vorgenommen werden).

---

<sup>1</sup> Als Eingabeeinheit für Daten ist in diesem Fall auf der Parameterkarte (s.u.) eine 10 anzugeben.

### 7.3. Programmsteuerkarten

Um das Programm starten zu können, werden bestimmte Informationen benötigt wie z.B. der Umfang der Stichprobe, die Zahl der zur Auswertung vorgesehenen Variablen, ob Missing Data vorliegen etc. Das Programm erhält die benötigten Angaben über sog. Programmsteuerkarten.

QUER3 kennt insgesamt 9 verschiedene Typen von Programmsteuerkarten. Die eingeklammerten Karten werden nur benötigt, falls auf der Parameterkarte eine entsprechende Angabe erfolgt.

#### 1. Parameterkarte

Jeder Rechenlauf wird durch eine Parameterkarte eingeleitet. Sie enthält u.a. Angaben über die Zahl der Fälle in der Gesamtstichprobe, die Zahl der Variablen, die logische File-Nummer der Eingabeeinheit (Kartenleser oder externer Speicher), den Missing Data Schlüssel und sie definiert die Zahl der gruppenbestimmenden Variablen und die Zahl der Substichproben.

#### 2. Formatkarte(n) für die Eingabedaten

Mit Hilfe der Formatangaben wird dem System mitgeteilt, welche Daten (Variablen) in die Auswertung aufzunehmen sind.

#### ( 3. ) Formatkarte(n) für die Ausgangsdaten

Diese Formatanweisung wird nur benötigt, sofern die Originaldaten in komprimierter Form zwischengespeichert werden.

#### ( 4. ) Missing Data Karte(n)

Missing Data Karten werden erwartet, wenn für jede in die Auswertung aufgenommene Variable ein eigener Missing Data Code gilt.

#### ( 5. ) Textkarten für Variablen

Um den Computerausdruck übersichtlicher zu gestalten, kann für jede Variable ein alphanumerischer Text eingelesen werden, der dann in den Tabellen erscheint.

## 6. Gruppentext- und Gruppenkarten

Diese Karten dienen zur Bestimmung der Substichproben (Gruppen).

### (7.) Klassenkarte(n)

Mit Hilfe der Angaben auf der (den) Klassenkarte(n) lassen sich mehrere einzelne Merkmalsausprägungen einer abhängigen Variablen zu Merkmalsklassen zusammenlegen.

### (8.) Varianzanalysekarte(n)

Sie kennzeichnen all jene Variablen, für die eine einfache Varianzanalyse mit F-Test über die mittels der Gruppenkarten definierten Substichproben durchzuführen ist.

### (9.) Verteilungskarte(n)

Für sämtliche auf dieser (diesen) Karte(n) benannten Variablen werden die Verteilungsstatistiken Schiefe und Exzeß ermittelt.

Im folgenden werden die Programmsteuerkarten und die darauf vorzunehmenden Lochungen in einer kurz gefaßten Eingabebeschreibung dargestellt. Für eine ausführlichere Beschreibung einzelner Sachverhalte erfolgen an gegebener Stelle Verweise auf spätere Abschnitte.

## STEUERKARTEN FÜR PROGRAMM QUER 3

### 1. Parameterkarte

Spalte(n) Name<sup>1</sup>

1 - 5	NP	Zahl der Fälle in der Gesamtstichprobe
7 - 9	MV	Zahl der Variablen (einschließlich der bis zu 10 gruppenbestimmenden Variablen); Maximal 100
11 - 12	IT	Nummer der File-Einheit, von der die Daten gelesen werden. Default: IT = 5 (entspricht einer Dateneingabe über Lochkarten)
14	IFO	Zahl der Formatkarten für Eingabedaten. Maximal 5. Default: IFO=1 (s. 2. Formatkarte(n) für Eingabedaten)
16	MISD	Missing Data Code zur Definition fehlender Daten blank: nur Blanks (ungelochte Felder) werden als Missing Data (MD) behandelt <ol style="list-style-type: none"> <li>1: Null und Blank werden als MD behandelt</li> <li>2: Für jede Variable wird ein eigener MD-Code gelesen. <b>Missing Data Karten</b> sind erforderlich.</li> <li>3: Es sind keine MD vorhanden. In den Daten erscheinende Blanks werden als Nullen gewertet.</li> </ol>
17	KDAT	blank: ohne Wirkung <ol style="list-style-type: none"> <li>1: Die Ausgangsdaten werden komprimiert. <b>Formatkarte(n) für Ausgangsdaten</b> erforderlich</li> </ol>
18	JFO	sofern KDAT=blank: ohne Wirkung sofern KDAT=1: Zahl der Formatkarten für die Ausgangsdaten Maximal 5. Default: IFO=1
19 - 20	ITREN	Anzahl der gruppenbestimmenden Variablen. Maximal 10 (s. Kap. 7.6. Gruppenbildung)
22 - 23	IGR	Zahl der Gruppen bzw. Substichproben. Maximal 20. Sofern einfache Häufigkeitsauszählungen durchgeführt werden sollen: IGR = blank

---

<sup>1</sup> Diese Bezeichnungen werden für den jeweiligen Parameter im Programm verwendet.

- 25 ITX blank: ohne Wirkung  
 1: Für jede der MV Variablen wird eine Textkarte gelesen und in den Tabellen ausgedruckt  
 2: Textkarten befinden sich auf Datei mit log. File-Nummer 15
- 27 IVAR blank: ohne Wirkung  
 1: Es werden Varianzanalysen, evt. auch Scheffé-Tests für paarweise Mittelwertvergleiche bei den auf der (den) **Varianzanalysekarten** bezeichneten Variablen durchgeführt
- 28 IVER blank: ohne Wirkung  
 1: Für jede Gruppe sowie für die Gesamtstichprobe werden die Verteilungsmaße Schiefe und Exzeß berechnet. Auf einer bzw. zwei **Verteilungskarten** ist anzugeben, für welche Variablen dies gelten soll
- 29 KLASS blank: ohne Wirkung  
 1: Die Scores der abhängigen Variablen werden zu Klassen zusammengefaßt. Auf einer bzw. zwei **Klassenkarten** ist anzugeben, für welche Variable dies gelten soll bzw. in welchen Schritten die Klassenbildung erfolgen soll (Anmerkung: Mittelwert und Standardabweichung werden stets aus den Originaldaten errechnet!)
- 30 IAL Signifikanzniveau für Schiefe und Exzeß  
 blank: alpha = 0.05  
 1: alpha = 0.01  
 2: alpha = 0.001
- 31 IREL blank: ohne Wirkung  
 1: Relativwerte bezogen auf Zeilensummen werden ausgedruckt
- 32 JREL blank: ohne Wirkung  
 1: Relativwerte bezogen auf Spaltensummen werden ausgedruckt
- 33 KREL blank: ohne Wirkung  
 1: Relativwerte bezogen auf die Gesamtsumme werden ausgedruckt

- 35 IQ            blank: es wird ein neuer Datensatz eingelesen und verarbeitet. Abfolge der Programmsteuerkarten: 1 bis 9
- 1: Es wird kein neuer Datensatz für den jeweiligen Rechenlauf gelesen. Die Häufigkeitsauszählungen für den Rechenlauf werden mit schon gelesenen Daten erstellt; die MV Variablen sind die gleichen wie im Lauf vorher. Es werden nur andere Gruppierungen vorgenommen bzw. andere Variable (aus dem Set von MV Variablen) dienen zur Bestimmung der Gruppen.  
In diesem Fall dürfen **keine** Formatkarten, Missing Data Karten und Textkarten dem Steuerkartenpaket beigelegt werden! (s. Kap. Mehrere Programmabläufe)
- 37 IREW        blank: ohne Wirkung
- 1: Die mit dem Parameter IT spezifizierte externe Einheit wird im jeweiligen Rechenlauf auf den Datenanfang zurückgesetzt. Dies ist erforderlich, falls der Datensatz neu eingelesen werden muß.  
(Gilt nicht, sofern IT = 5, d.h. Dateneingabe über Kartenleser)
- 38 - 80        beliebiger Text, erscheint in der Ausgabe

## 2. Formatkarte(n) für Eingabedaten (sofern IQ = 1: keine Formatkarte(n)!)

F-Format für die MV Variablen, also einschließlich aller gruppenbestimmenden Variablen. Maximal sind 5 Formatkarten zulässig. Spalten 1 bis 80 können gelocht werden. Sofern IQ = 1 (Spalte 35 der Parameterkarte) dürfen keine Formatkarten dem Steuerkartenpaket beigelegt werden!

## 3. Sofern KDAT = 1: Formatkarte(n) für die Ausgangsdaten

Sofern die Ausgangsdaten komprimiert werden sollen (Spalte 17 der Parameterkarte: KDAT = 1): **A-Format** für die MV Variablen, wie sie im Originaldatensatz vorliegen. Maximal 5 Formatkarten. Spalten 1 bis 80 können gelocht werden. Die Formatkarten (2) für Eingabedaten enthalten in diesem Fall die Formatangaben für die komprimierten Daten. Die Originaldaten werden mit dem Format (3) gelesen und mit dem unter (2) spezifizierten Format, das vom Programm in ein A-Format umgewandelt wird, auf einen Plattenspeicher geschrieben.

Sofern IQ = 1: es dürfen keine Formatkarten für Ausgangsdaten beigegeben werden (s. Kap. 7.5. Datenkomprimierung)

#### 4. Sofern MISD = 2: Missing Data Karte(n)

Für jede der MV Variablen wird ein eigener Missing Data Code angegeben. Der entsprechende Code ist in die Spalte(n) zu lochen, auf der (denen) sich die jeweilige Variable auf den Datenträgern befindet (die Missing Data Karten werden mit dem Format (2) gelesen). Es sind so viele Missing Data Karten erforderlich, wie Datenkarten pro Fall durch das Format für Eingabedaten (2) spezifiziert worden sind (s. Kap. 7.4. Verarbeitung von Missing Data).

Sofern IQ = 1: es dürfen keine Missing Data Karten beigegeben werden!

#### 5. Sofern ITX = 1: Textkarten für Variablen

Es folgen MV Textkarten. Für den alpha-numerischen Text einer Variablen stehen die Spalten 1 bis 80 zur Verfügung.

Sofern IQ = 1: es dürfen keine Textkarten beigegeben werden! (s. Kap. 7.11. Variable Labels)

#### 6. Sofern IGR größer 0: Text für Gruppenbezeichnung und Gruppenkarten

Für jede der IGR Gruppen sind eine Gruppentextkarte sowie ITREN Gruppenkarten zu lochen.

##### a) Gruppentextkarte für Gruppe j (j = 1 ... IGR)

Spalten	1 - 15	Kurzbezeichnung der Gruppe; erscheint in den ausgedruckten Tabellen
	17 - 80	ausführlicher Text für die jeweilige Gruppe

##### b) Gruppenkarten

Für jede der IGR Gruppen sind ITREN Gruppenkarten zu lochen.

Spalten	1 - 4	NO(K)	lfd. Nr der jeweiligen gruppenbestimmenden Variablen
	5 - 10	GU(J,K)	kleinster Wert, den die gruppenbestimmende Variable k annehmen darf, um der Gruppe j zugewiesen zu werden
	11 - 15	GO (J,K)	größter Wert, den die gruppenbestimmende Variable k annehmen darf, um der Gruppe j zugewiesen zu werden
	16 - 20	G3(J,K)	zusätzliche Codes, die die gruppenbestimmende Variable k annehmen darf, um der Gruppe j zugewiesen zu werden. (Sofern die Gruppenzuweisung
	21 - 25	G4(J,K)	

durch das Intervall GU bis GO  
exakt definiert ist, bleiben G3  
und G4 blank)

27 - 34 ITE(J,K) alpha-numerische Bezeichnung  
für diesen Code der gruppen-  
bestimmenden Variablen (kann  
blank bleiben)

**Sofern IGR = blank und ITREN = blank:** einfache Häufigkeitsauszählungen mit dem gesamten Datensatz: Textkarte für Gruppenbezeichnung und Gruppenkarten entfallen.

**Sofern IGR = blank und ITREN größer Null:** einfache Häufigkeitsauszählung, jedoch mit reduziertem Datensatz. Nur solche Probanden werden berücksichtigt, deren Werte in der (den) gruppenbestimmenden Variablen den durch GU, GO, ev. G3 und G4 spezifizierten Gruppen entsprechen. In diesem Fall sind **eine** Gruppentextkarte und insgesamt ITREN Gruppenkarten erforderlich.

*Anmerkung:* Die Reihenfolge der gruppenbestimmenden Variablen auf den ITREN Gruppenkarten einer Gruppe (NO(K), wobei  $k = 1 \dots ITREN$ ) muß in allen Gruppen die gleiche sein! (s. Kap. 7.6. Gruppenbildung).

#### 7. Sofern KLASS = 1: **Klassenkarte(n)**

Die Rohwerte mancher oder aller Variablen werden zu Rohwerteklassen zusammengefaßt (gruppierte Häufigkeiten). Beginnend vom kleinsten Rohwert, den die Variable j aufweist, werden jeweils k Werte zur gleichen Klasse zusammengefaßt. K muß auf der (den) Klassenkarte(n) – sofern mehr als 80 Variable eingegeben werden, sind 2 Klassenkarten erforderlich – für jede Variable spezifiziert werden. Der entsprechende Code ist in die Spalte zu lochen, deren Nummer der laufenden Variablen-Nummer entspricht, für die eine Rohwertzusammenlegung durchgeführt werden soll. (Bei mehr als 80 Variable steht die Spalte 1 der 2. Klassenkarte für die Variable 81 usw.)

Codes für die Klassenkarte(n):

- blank oder 1: keine Klassenzusammenlegung; Häufigkeitszählung anhand der Originalwerte
- 2: beginnend vom kleinsten Rohwert, den die Variable j aufweist, werden jeweils 2 aufeinanderfolgende (ganzzahlige) Merkmalsausprägungen zu einer Klasse zusammengefaßt
- 3.....9: beginnend vom kleinsten Rohwert, den die Variable j aufweist, werden jeweils 3.....9 aufeinanderfolgende Merkmalsausprägungen zu einer Klasse zusammengefaßt

0: 10 aufeinanderfolgende Merkmalsausprägungen werden zu einer Klasse zusammengefaßt

*Anmerkung:* Die Berechnung von Mittelwert, Standardabweichung, ev. Schiefe und Exzeß erfolgt stets anhand der ungruppierten Häufigkeiten (s. Kap. 7.8. Zusammenfassung der Rohwerte zu Klassen).

### 8. Sofern IVAR = 1: Varianzanalysekarte(n)

Für manche der MV Variablen werden Varianzanalysen (F-Tests) und ev. Scheffé-Tests für paarweise Mittelwertvergleiche durchgeführt.

#### a) Varianzanalyse

Jede Spalte der Varianzanalysekarte(n), deren Nummer einer laufenden Variablennummer entspricht, für die eine Varianzanalyse berechnet werden soll, ist mit einer Lochung 1 zu versehen. Bei mehr als 80 Variablen sind 2 Varianzanalysekarten erforderlich.

Sofern für sämtliche Variable (außer der (den) gruppenbestimmenden Variablen) Varianzanalysen durchzuführen sind: Spalte 1 der Varianzanalysekarte ist mit 9 zu lochen. Sofern MV größer als 80, ist eine zweite Varianzanalysekarte, die dann allerdings blank bleibt, anzufügen.

#### b) Varianzanalyse und Scheffé-Tests

Für Variablen, bei denen zusätzlich Scheffé-Tests für paarweise Mittelwertvergleiche interessieren, ist an der entsprechenden Stelle der Varianzanalysekarte(n) eine 2 zu lochen.

Sofern für sämtliche Variable (Ausnahmen wie bei a)) zusätzlich Scheffé-Tests gerechnet werden sollen: Spalte 1 der Varianzanalysekarte ist mit 8 zu lochen.

(Bei mehr als 80 Variablen bleibt die dann erforderliche zweite Varianzanalysekarte blank).

Bei Durchführung von Scheffé-Tests werden automatisch Schiefe und Exzeß für die betreffende Variable ermittelt (s. Kap. 7.7. Varianzanalyse und Scheffé-Tests)

### 9. Sofern IVER = 1: Verteilungskarte(n)

Für manche der MV Variablen werden die Verteilungsparameter Schiefe und Exzeß sowie deren T-Werte und Signifikanzniveaus errechnet. Jede Spalte der Verteilungskarte(n), deren Nummer einer laufenden Variablennummer entspricht, für die Schiefe und Exzeß interessieren, ist mit einer Lochung 1 zu versehen.

(Bei mehr als 80 Variablen sind 2 Verteilungskarten erforderlich).

Sofern für sämtliche Variable (außer der (den) gruppenbestimmenden Variablen) die Verteilungsmaße erwünscht sind: Spalte 1 der Verteilungskarte ist mit 9 zu lochen.

(Die ev. erforderliche zweite Verteilungskarte bleibt dann blank; s. Kap. 7.9. Prüfung auf Normalverteilung).

## 10. Sofern Dateneingabe über Lochkarten: **Datenkarten** für die NP Pro-

Anschließend eröffnen sich folgende Möglichkeiten:

- a) ein weiterer Rechenlauf soll vorgenommen werden. Abfolge der Steuerkarten (1) bis (9), ev. (1), (6) bis (9);  
(s. Kap. 7.10. mehrere Rechenläufe)
- b) Programmende: Vor der für die jeweilige Rechenanlage gültigen Endkarte (end-of-file-Karte) muß eine Leerkarte liegen. Sie signalisiert das Programmende. Wird sie vergessen, wird ein neuer Programmlauf gestartet, der vom System abgebrochen wird, da keine weiteren Steuerkarten folgen.

### **Benötigte externe Einheiten:**

Kanal 5 (Input – Kartenleser)

Kanal 6 (Output – Schnelldrucker)

Kanal 8 (Plattenzwischenspeicher, scratch file zur binären Zwischenspeicherung der Daten)

Kanal 9 (Plattenzwischenspeicher, scratch file. Die Daten jedes Falles werden auf diesen Plattenbereich zweimal, zwecks Missing Data Verarbeitung, geschrieben.)

ev. Kanal 15

    Datei enthält Variable Labels (Permanent File)

### **7.4. Verarbeitung von Missing Data (MD)**

Das Programm ist für IBM-Maschinen geschrieben und auf einer IBM 370/148 ausgetestet worden. Da bei diesen Anlagen ungelochte Felder (blanks) sowohl im F- als auch im I-Format intern als Nullen dargestellt werden, ist eine direkte Unterscheidung zwischen Nullen und Blanks nicht möglich. (Bei Maschinen, die Blanks mit -0 bzw. .0. verschlüsseln, dient zur Unterscheidung mit Null die ISIGN- bzw. SIGN-Funktion). IBM-FORTRAN gestattet das zweimalige Lesen einer Karte bzw. eines Records mit Hilfe des T-Formats. Um das Programm aber auch für Rechenanlagen anderer Hersteller kompatibel zu gestalten, wird zur Verarbeitung von Blanks eine andere Strategie eingeschlagen:

Das Programm wandelt das vom Benutzer spezifizierte F-Format für die Dateneingabe in ein A-Format um. Die Daten jedes Probanden bzw. Falles

werden daraufhin *zweimal* mit diesem A-Format auf einen Plattenzwischenpeicher (scratch file) geschrieben. (Im A-Format gelesene Blanks werden bekannterweise maschinenintern von Nullen unterschieden). In der Eingabe-Routine wird der erste Record jedes Falles von diesem Plattenbereich im F-, der zweite im A-Format gelesen. Sofern im F-Datenvektor eine Null auftaucht, wird im entsprechenden Element des A-Vektors geprüft, ob es sich dabei um ein Blank handelt. Trifft dies zu, so bekommt das Element des F-Vektors den neuen Wert  $-1.0E06$  zugewiesen. Bei der Missing-Data-Verarbeitung wird dann auf diesen Wert hin abgefragt.

Da die Vektoren, die die Eingabedaten aufnehmen, mit einfacher Genauigkeit ausgelegt sind (d.h. REAL\*4), darf die Feldlänge des A-Formats nicht größer als 4 sein. Das bedeutet, daß jede Variable höchstens 4 Lochfelder umfaßt, das Eingabeformat demnach eine maximale Feldbreite von 4 aufzuweisen hat (F4.0, F4.1 bzw. F4.2). Sind größere Feldbreiten erforderlich, muß der Eingabevektor Y im Hauptprogramm sowie X in SUBROUTINE WAND auf REAL\*8 gesetzt werden. Damit ist gleichzeitig aber auch die Wortlänge der Variablen BLEER, CLEER und DLEER zu vergrößern!

Maschinentypen, die eine Unterscheidung von Nullen und Blanks im F-Format gestatten (wie z.B. CDC 6500 und Cyber 175) benötigen die oben geschilderte zeitaufwendige Missing-Data-Routine nicht. Bei der im Anhang wiedergegebenen Auflistung des Quellenprogramms sind die Programmstatements, die bei diesen Maschinentypen entfallen können, besonders gekennzeichnet (s. Kap. Programmversion A).

Die Missing Data Routine ermöglicht vier Verarbeitungsstrategien:

- a) Ungelochte Spalten (sog. Blanks) werden als MD behandelt: Parameter MISD (Sp. 16 der Parameterkarte) = blank oder null.

Sofern in den Datenkarten (bzw. Karten-Images bei Dateneingabe von externen Speichern) eines Probanden ein ungelochtes Element auftaucht, wird die Variable, die sich in den jeweiligen Spalten manifestiert für diesen Probanden als Missing Datum gewertet.

- b) Sowohl ungelochte als auch mit Null gelochte Spalten werden als MD behandelt: Parameter MISD = 1

- c) Es sind keine Missing Data vorhanden: Parameter MISD = 3.

Im gesamten Datensatz sind keine Informationen enthalten, die als MD zu behandeln sind. Sofern Blanks auftauchen, werden sie wie Nullen gewertet und als solche in die Berechnungen aufgenommen.

Wird eine der unter a) bis c) genannten Optionen gewählt, so gilt die jeweilige MD-Verarbeitungsroutine für alle im Datensatz auszuwertenden Variablen. Unterscheiden sich die Variablen jedoch hinsichtlich ihres Codes für Missing Data, so ist die nachfolgende Option zu wählen:

- d) Für jede Variable wird ein eigener MD-Code gelesen: Parameter MISD = 2. In diesem Fall sind dem Steuerkartenpaket sog. *Missing Data Karten* beizugeben, auf denen der für jede Variable geltende MD-Code zu lochen ist. Die entsprechende Lochung für Variable j ist an die Stelle der MD-Karte(n) (rechtsbündig) zu setzen, auf der sich Variable j auf den Datenkarten bzw. Kartenimages befindet. Erscheint im Datensatz für Variable j eine derartige Lochung oder ein ungelochtes Feld, so wird dies als Missing Datum gewertet. Falls bei Variablen nur ungelochte Spalten als Missing Data gelten sollen, bleiben die entsprechenden Spalten auf der (den) Missing Data Karte(n) blank. Die Missing Data Karte(n) werden mit dem Format für die Eingabedaten gelesen. Daraus geht hervor, daß so viele MD-Karten erforderlich sind wie Datenkarten (bzw. Karten-Images) pro Fall vorliegen. Sofern die Ausgangsdaten komprimiert werden (s. Kap. Datenkomprimierung), richtet sich die Verspaltung der Lochungen für die MD-Karten nach den komprimierten Daten.

Das Vorgehen bei der Erstellung von Missing Data Karten sei an zwei Beispielen erläutert. Insgesamt 10 Variable sind auszuwerten. Für jeden Probanden sind 3 Lochkarten erforderlich; die interessierenden Variablen befinden sich auf den Karten 1 und 2. Die Daten sind mit folgendem Format zu lesen:

(4X,2F1.0,5X,3F2.0/7X,4F1.0,F2.0/)

Folgende Codes gelten als Missing Data:

Variable	Code
1	9 und blank
2	blank
3	99 und blank
4	1 und blank
5	blank
6	9 und blank
7	4
8	blank
9	blank
10	98 und blank

Beispiel a) keine Datenkomprimierung

(1) Parameterkarte: MISD = 2

(2) Formatkarte für Eingabedaten: (4X,2F1.0,5X,3F2.0/7X,4F1.0,F2.0/)

(4) Missing Data Karten (insgesamt 3 MD-Karten werden benötigt)

	Spalten	Lochung
Karte 1	5	9
	6	blank
	12 - 13	99
	14 - 15	01 oder 1 (rechtsbündig)
	16 - 17	blank
Karte 2	8	9
	9	4
	10	blank
	11	blank
	12 - 13	98
Karte 3	1 - 80	blank

weitere Steuerkarten s. Eingabebeschreibung

Beispiel b) Daten werden komprimiert

(1) Parameterkarte: MISD = 2, KDAT = 1

(2) Formatkarte für Eingabedaten: (2F1.0,3F2.0,4F1.0,F2.0)

(3) Formatkarte für Ausgangsdaten: (4X,2A1,5X,3A2/7X,4A1,A2/)

(4) Missing Data Karte (hier: 1 Karte. Verspaltung bezieht sich auf :  
Format (2))

<u>Spalten</u>	<u>Lochung</u>
1	9
2	blank
3 - 4	99
5 - 6	01 oder 1 (rechtsbündig)
7 - 8	blank
9	9
10	4
11	blank
12	blank
13 - 14	98

Weitere Steuerkarten s. Eingabebeschreibung

### 7.5. Datenkomprimierung

Die für die MD-Verarbeitung eingeschlagene Strategie benötigt einen Zwischenspeicher, der die Daten jedes Probanden bzw. Falles zweimal aufnimmt. Sofern es sich um eine umfangreiche Stichprobe handelt, müssen daher größere Plattenbereiche reserviert werden. Der Plattenbedarf vergrößert sich zusätzlich, wenn jeder Fall mehrere Lochkarten umfaßt. Da die Daten mit dem Eingabeformat (das, wie bereits beschrieben, in ein A-Format umgesetzt wird) gelesen und auf Platte gebracht werden, muß auch Platz für alle nicht zur Auswertung kommenden, d.h. vom Format überlesenen Variablen reserviert werden. Ein Beispiel möge dies verdeutlichen. Angenommen, die Daten jedes Probanden befinden sich auf insgesamt 10 aufeinanderfolgenden Lochkarten (bzw. Lochkarten-Images). Zur Auswertung sollen aber nur 10 Variable gelangen, von denen 5 auf der dritten Karte, die anderen 5 auf der vierten Karte abgelocht sind. Das Eingabeformat sähe dann etwa folgendermaßen aus:

(//5X,F1.0,2X,F2.0,3F1.0/7X,F1.0,2X,3F1.0,5X,F1.0/////). Die Daten werden im entsprechenden A-Format auf den Zwischenspeicher übertragen. Dabei wird für jeden Probanden Platz für 10 Karten beansprucht, der größtenteils nicht ausgefüllt wird.

Sofern die Speicherkapazität der Plattenbereiche beschränkt ist, empfiehlt es sich, die Ausgangsdatei zu komprimieren. Das bedeutet, daß auf dem Scratch File nur die aktuell in die Berechnungen eingehenden Variablen

abgespeichert werden. Dazu definiert man auf der Parameterkarte den Parameter KDAT als 1 und fügt dem Steuerkartenpaket eine oder mehrere (je nach dem Parameter JFO) Formatkarten für die *Ausgangsdatei* (A-Format!) bei. Die Daten werden mit diesem Format gelesen, aber mit dem Format für die *Eingabedatei* (F-Format, vom Programm in ein A-Format transformiert) auf den Zwischenspeicher geschrieben. Im Beispiel würde das Format für die Ausgangsdatei lauten:

(//5X,A1,2X,A2,3A1/7X,A1,2X,3A1,5X,A1/////). Das Format für die Eingabedatei wäre dann: (F1.0,F2.0,8F1.0). Mit dem entsprechenden A-Format werden die Daten dann, zweimal für jeden Probanden, auf der Platte gespeichert.

## 7.6. Gruppenbildung

Aus der Gesamtdatei lassen sich mit Hilfe von Maximal 10 sog. gruppenbestimmenden Variablen bis zu 20 Unterstichproben bilden, wobei für jede auszuzählende (abhängige) Variable Häufigkeitsverteilungen für diese Gruppen erstellt werden. Die Gruppenbildung wird mit sog. Gruppenkarten vorgenommen. Eine Gruppe ist dadurch definiert, daß eine bzw. bis zu 10 gruppenbestimmende Variable entweder in zu spezifizierende Intervalle fallen oder ganz bestimmte Codes aufweisen. Der Parameter ITREN (Parameterkarte Spalten 19 - 20) definiert die Zahl der gruppenbestimmenden Variablen, IGR (Spalten 22 - 23) legt die Zahl der Gruppen bzw. Unterstichproben fest.

Für jede der IGR Gruppen ist zuerst eine Textkarte zu lochen. Der Text erscheint in den Tabellen. Dann folgen für jede der ITREN gruppenbestimmenden Variablen Karten, die festlegen, welche Codes die jeweilige Variable aufweisen darf, um zur Gruppe j (wobei j=1..IGR) zu gehören.

## Anordnung der Gruppentext- und Gruppenkarten

## a) Gruppentextkarte für Gruppe j

Spalten	1 - 15	Kurzbezeichnung der jeweiligen Gruppe
	17 - 80	Ausführliche Kennzeichnung der Gruppe (kann blank bleiben)

## b) Gruppenkarten

Spalten	1 - 4	NO(K)	laufende Nummer der jeweiligen gruppenbestimmenden Variablen
	5 - 10	GU(J,K)	kleinster Wert, den die gruppenbestimmende Variable K annehmen darf, um der Gruppe J zugewiesen zu werden
	11 - 15	GO(J,K)	größter Wert, den die gruppenbestimmende Variable K annehmen darf, um der Gruppe J zugewiesen zu werden
	16 - 20	G3(J,K)	weitere zusätzliche Codes, die die gruppenbestimmende Variable K annehmen darf, um der Gruppe J zugewiesen zu werden;
	21 - 25	G4(J,K)	(sofern die Gruppenzuweisung durch das Intervall GU bis GO exakt definiert ist, bleiben G3 und G4 blank)
	27 - 34	ITE(J,K)	alpha-numerische Bezeichnung für diese(n) Code(s) der gruppenbestimmenden Variablen (kann blank bleiben).

Sofern für sämtliche gruppenbestimmenden Variablen gilt: die Codeziffern der gruppenbestimmenden Variablen mit der laufenden Nummer k (wobei  $k = 1 \dots \text{ITREN}$ ) liegen im Bereich GU(J,K) einschließlich GO(J,K) bzw. sind identisch mit dem Wert G3(J,K) oder G4(J,K) – so wird der betreffende Fall der Gruppe j zugewiesen.

Die Zuordnung eines Probanden zu einer der IGR Gruppen läßt sich am einfachsten für den Fall einer einzigen gruppenbestimmenden Variablen verdeutlichen (ITREN = 1):

Hier sind insgesamt IGR-Gruppentext- und IGR Gruppenkarten erforderlich. Die erste Gruppenkarte definiert die 1. Substichprobe. Alle Probanden, die in der Variablen mit der laufenden Nummer NO (gruppenbestimmende Variable) Codes aufweisen, die auf dieser ersten Gruppenkarte spezifiziert sind, d.h. deren gruppenbestimmende Variable Merkmalsausprägungen zeigt, die im Bereich GU(1,1) bis GO(1,1) liegen oder den Wert G3(1,1) bzw. G4(1,1) enthalten ( die Indizes stehen für die 1. Gruppe, 1. gruppenbestimmende Variable), werden der 1. Substichprobe zugewiesen.

Die Bestimmung der 2. Substichprobe erfolgt über die 2. Gruppenkarte. Wenn der in der Variablen mit der laufenden Nummer NO (wobei NO identisch sein muß mit der Angabe auf der ersten und allen folgenden Gruppenkarten) gefundene Code im Bereich GU(2,1) bis GO(2,1) liegt (die Indizes stehen hier für 2. Gruppe, 1. gruppenbestimmende Variable), oder einen Wert von G3(2,1) bzw. G4(2,1) aufweist, gehört der jeweilige Proband in die 2. Gruppe.

Um bei zwei gruppenbestimmenden Variablen (ITREN = 2) eine einzelne Gruppe zu definieren, benötigt man 2 Gruppenkarten. Die erste Gruppenkarte legt die Codes für die erste gruppenbestimmende Variable fest, die zweite Gruppenkarte spezifiziert die zulässigen Merkmalsausprägungen der 2. gruppenbestimmenden Variable.

Ein Fall wird der 1. Gruppe zugewiesen, sofern sich die folgende Beziehung als zutreffend erweist:

Der Code in der gruppenbestimmenden Variablen mit der laufenden Nummer NO(1) (= erste gruppenbest. Variable) liegt im Bereich GU(1,1) bis GO(1,1) (Indizes stehen für 1. Gruppe, 1. gruppenbestimmende Variable) oder hat den Wert G3(1,1) bzw. G4(1,1) *und*

der Code in der gruppenbestimmenden Variablen mit der laufenden Nummer NO(2) (= zweite gruppenbest. Variable) liegt im Bereich GU(1,2) bis GO(1,2) (Indizes stehen für 1. Gruppe, 2. gruppenbest. Variable) oder hat den Wert G3(1,2) bzw. G4(1,2).

Für die zweite Gruppe lauten die Parameter:

1. Gruppenkarte: NO(1), GU(2,1), GO(2,1), G3(2,1), G4(2,1)
2. Gruppenkarte: NO(2), GU(2,2), GO(2,2), G3(2,2), G4(2,2), usw.

NO(K), GU (J,K) und GO(J,K) müssen angegeben werden, G3(J,K) und G4(J,K) nur dann, wenn der Bereich GU bis GO zur Definition der Gruppe nicht ausreicht. Andernfalls bleiben sie blank. Die Gruppentextkarten sind obligatorisch, man kann dafür aber Leerkarten begeben. In diesem Fall werden die Gruppen im Tabellenausdruck lediglich mit den laufenden Nummern 1 bis IGR gekennzeichnet.

Ein Anwendungsbeispiel möge die Vorgehensweise bei der Gruppenbildung verdeutlichen. Aus einer Datei sollen 4 Untergruppen gebildet werden (d.h. IGR = 4), und zwar anhand der 3 gruppenbestimmenden Variablen

– Konfession (laufende Variablen-Nummer sei 1)

Codes: 1 = katholisch 2 = evangelisch 3 = sonstige

– Parteipräferenz (laufende Variablen-Nummer sei 5)

Codes: 1 = SPD 2 = CDU 3 = CSU 4 = FDP 5 = sonstige

– Geschlecht (laufende Variablen-Nummer sei 7)

Codes: 1 = männlich 2 = weiblich

Die vier Gruppen sollen sich aus Probanden zusammensetzen, die folgende Bedingungen erfüllen:

Gruppe 1:	katholisch,	CDU/CSU-Präferenz,	männlich
2:	katholisch,	SPD/FDP-Präferenz,	männlich
3:	katholisch,	CDU/CSU-Präferenz,	weiblich
4:	katholisch,	SPD/FDP-Präferenz,	weiblich

Daraus ergeben sich folgende 16 Gruppentext- und Gruppenkarten:

Karte	Spalte	1	4	10	15	20	25	30	
		NO	GU	GO	G3	G4			
1. Gruppe	1	KATH, CDU, MAEENL					ausführl. Text		
	2	1	1	1			KATHOL. (1)		
	3	5	2	3			CDU, CSU		
	4	7	1	1			MAEENL.		
2. Gruppe	5	KAT, SPD/FDP, MAE							
	6	1	1	1			KATHOL.		
	7	5	1	1	4		SPD/FDP (2)		
3. Gruppe	8	7	1	1			MAEENL.		
	9	KATH, CDU, WEIBL.							
	10	1	1	1			KATHOL.		
	11	5	2	3			CDU, CSU		
4. Gruppe	12	7	2	2			WEIBL.		
	13	KAT, SPD/FDP, WEI							
	14	1	1	1			KATHOL.		
	15	5	1	1	4		SPD/FDP		
	16	7	2	2			WEIBL.		

Im Beispiel wäre es durchaus möglich, zusätzlich die vier entsprechenden Gruppen für die Konfession evangelisch zu bilden. Damit ergäbe sich ein Parameter IGR von 8. (Beachte: maximale Gruppenzahl = 20). Die gleiche Wirkung läßt sich durch einen zweiten Rechenlauf mit 4 Gruppen erzielen, wobei immer der Code für katholisch (1) durch evangelisch (2) zu ersetzen ist (s. Kap. Mehrere Rechenläufe).

<sup>1</sup> Sofern nur eine einzige Codeposition einer Variablen zur Gruppe j gehört, ist dieser Code sowohl kleinster (GU) also auch größter (GO) Wert der gruppenbestimmenden Variablen.

<sup>2</sup> Da hier die gruppenbildenden Codes nicht kontinuierlich aneinander anschließen, wird als GU und GO der Code 1 (SPD) und als zusätzlicher Code 4 (FDP) angegeben. Würde man GU=1 und GO=4 wählen, so würden auch Probanden mit CDU und CSU Präferenz in diese Gruppe fallen. (Man hätte hier auch GU=4 und GO=4, und G3 als 1 definieren können.)

Zur Vermeidung fehlerhafter Gruppenbildung ist folgendes zu beachten:

1. Bei den Gruppen muß es sich stets um echte Teilmengen handeln, d.h. jedes Individuum darf nur einer einzigen – oder keiner – Gruppe angehören.  
Würde man im obigen Beispiel die 2. Gruppe definieren als „kath., CDU/FDP, männlich“, dann erhielte man fehlerhafte Resultate, da ein katholischer, männlicher Proband mit CDU Präferenz dann sowohl der 1. als auch der 2. Gruppe zuzuordnen wäre. (Das Programm würde ihn der 1. Gruppe zuweisen).
2. Alle IGR Gruppen müssen sich aus den gleichen ITREN gruppenbildenden Variablen zusammensetzen. Folgende Gruppenbildung ist demnach unzulässig:  
Gruppe 1 katholisch, CDU-Präferenz, weiblich  
Gruppe 2 evangelisch, SPD-Präferenz, Volksschüler  
Geschlecht und Schulbildung sind verschiedene Variablen.
3. Die Reihenfolge der gruppenbildenden Variablen auf den ITREN Gruppenkarten einer Gruppe (d.h. NO(K), wobei  $K = 1 \dots ITREN$ ) muß in allen Gruppen die gleiche sein. Unzulässig ist es, wenn NO (Spalten 1 bis 4 der Gruppenkarten) in der ersten Gruppe mit 1 - 5 - 7 (Karten 2,3,4), in der 2. Gruppe (Karten 6,7,8) dagegen mit 5 - 1 - 7 definiert ist. In einem solchen Fall wird der Rechenlauf mit der Fehlermeldung „DIE NUMMERN DER TRENNVARIABLEN STIMMEN IN DEN VERSCHIEDENEN GRUPPEN NICHT ÜBEREIN. PROGRAMMABRUCH.“ abgebrochen.

Im folgenden seien die Gruppenkarten für das Beispiel 4, Tabelle e des Abschnitts 2 dargestellt.

Gruppenbestimmende Variable:

- Schulbildung (laufende Variablen-Nummer sei 6)  
Codes: 1 = Hauptschule (HS) 2 = Mittelschule (MS) 3 = Gymnasium (GY)
- Geschlecht (laufende Variablen-Nummer sei 25)  
Codes: 1 = männlich (M) 2 = weiblich (W)
- Testangst (laufende Variablen-Nummer sei 30)  
Codes: 0 bis 4 niedrige Testangst (TAnied)  
5 bis 7  
8 - 12 hohe Testangst (TAhoch)

Spalten	4	10	15	20	25	30	35
	.....+	.....+	.....+	.....+	.....+	.....+	.....+
	NO	GU	GO				
	GRUPPE 1			TAniedr.,	maennl.	GS	
	30	0	4				
	25	1	1				
	6	1	1				
	GRUPPE 2			TA hoch,	maennl.	GS	
	30	8	12				
	25	1	1				
	6	1	1				
	.....+	.....+	.....+	.....+	.....+	.....+	.....+

Der Aufwand an Eingabekarten für die Gruppenbildung verringert sich bei Vorliegen von nur einer gruppenbestimmenden Variablen. Hier sind für jede Unterstichprobe eine Gruppentextkarte und *eine* Gruppenkarte erforderlich.

Einfache Häufigkeitsauszählungen über den gesamten Datensatz (sog. Grundauszählungen) benötigen keine Gruppentext- und Gruppenkarten (Parameterkarte: IGR = blank, ITREN = blank).

Allgemein läßt sich die Zahl der benötigten Gruppentext- und Gruppenkarten nach der Formel  $Zahl = IGR + ITREN \cdot IGR$  ermitteln.

### 7.7. Varianzanalyse und Scheffé-Tests

Das Programm berechnet für jede Variable automatisch Gruppenmittelwerte und -standardabweichungen, sowie Mittelwert und Standardabweichung in der Gesamtstichprobe. Die Entscheidung darüber, ob diese Maße im Hinblick auf das der jeweiligen Variablen zugrundeliegende Datenniveau interpretationsfähig sind, liegt beim Benutzer. Bei Vorliegen von intervall-(oder ev.

auch ordinal-) skalierten Daten interessiert häufig, ob die Mittelwertunterschiede in den einzelnen Gruppen statistisch signifikant sind.

Zu diesem Zweck führt das Programm auf Wunsch einfache Varianzanalysen für unabhängige Stichproben durch. Voraussetzung ist, daß der Parameter IVAR (Spalte 27 der Parameterkarte) mit 1 gelocht und eine (bzw. bei Vorliegen von mehr als 80 Variablen: zwei) Varianzanalysekarte(n) dem Steuerkartenpaket beigelegt ist, aus der (denen) hervorgeht, für welche Variablen Varianzanalysen zu berechnen sind (Lochung 1 an der Stelle der Varianzanalysekarte(n), deren Spaltennummer der laufenden Nummer einer Variablen entspricht, für die eine Varianzanalyse vorgenommen werden soll).

Die Computerausdrucke enthalten in diesem Fall den F-Wert der Analyse, ferner die mittleren Abweichungsquadrate „zwischen“ den und „innerhalb“ der Gruppen, Freiheitsgrade des Zählers und des Nenners, die Irrtumswahrscheinlichkeit alpha (in Prozent) für den gefundenen F-Wert sowie einen Hinweis darauf, ob die Mittelwerte als gleich zu betrachten sind, d.h. ob die Nullhypothese „die IGR Unterstichproben stammen aus der gleichen Grundgesamtheit und haben den gleichen Mittelwert“, beizubehalten oder zu verwerfen ist. Sofern sich eine Irrtumswahrscheinlichkeit von 5% und kleiner errechnet, erscheint der Text: GLEICHHEIT DER MITTELWERTE = NEIN, andernfalls lautet die verbale Beschreibung des Testergebnisses GLEICHHEIT DER MITTELWERTE = JA.

Da der F-Test Varianzhomogenität voraussetzt, wird gleichzeitig mit der Varianzanalyse der Barlett-Test auf Gleichheit der Varianzen durchgeführt<sup>1</sup>. Im Ausdruck findet sich als Testergebnis der Chi-Quadratwert, die prozentuale Irrtumswahrscheinlichkeit, sowie die Mitteilung „GLEICHHEIT DER VARIANZEN = JA“ bzw. „GLEICHHEIT DER VARIANZEN = NEIN“. Zur Schätzung der praktischen Signifikanz wird das Maß Omega-Quadrat nach Hays berechnet und ausgedruckt.<sup>2</sup>

---

<sup>1</sup> vgl. Sachs, S., Statistische Auswertungsmethoden, 1971, S. 384 ff.

<sup>2</sup> vgl. Hays, W.L., 1969, S. 382  
vgl. auch Bredenkamp, J., 1969 und  
Bredenkamp, J., 1972, S. 48 ff.

Für den Fall zweier Gruppen werden zusätzlich die Prüfgröße T des T-Tests sowie zur Schätzung der praktischen Signifikanz das Quadrat des punktbiserialen Korrelationskoeffizienten  $r_{pb}^2$  und das nach Levy berechnete Klassifikationsmaß U ermittelt.<sup>1</sup>

Ein signifikantes varianzanalytisches Testergebnis besagt, daß nicht alle IGR Gruppenmittelwerte als gleich bzw. als aus der gleichen Grundgesamtheit stammend, betrachtet werden können. Zumindest zwei Mittelwerte unterscheiden sich voneinander. Das Programm bietet eine Option, mit der bestimmt werden kann, zwischen welchen der insgesamt möglichen  $\binom{IGR}{2} = \frac{IGR \cdot (IGR-1)}{2}$  Zweierkombinationen von Mittelwerten Unterschiede bestehen. Zu diesem Zweck ist die jeweilige Spalte der Varianzanalysekarte, deren Spaltennummer der laufenden Variablennummer entspricht, mit einer Lochung 2 zu versehen. Es werden dann sämtliche Mittelwertkombinationen mit Hilfe des Scheffe-Tests analysiert.<sup>2</sup> Diese Tests werden allerdings nur durchgeführt, sofern mehr als zwei Gruppen bzw. weniger als 11 Unterstichproben vorliegen und die Varianzanalyse ein signifikantes Ergebnis erbrachte.

---

<sup>1</sup> vgl. Bredenkamp, 1972, S. 48 ff.

$$r_{pb}^2 = \frac{T^2}{T^2 + N-2}$$

Das Testergebnis U liefert eine Schätzung darüber, bei wieviel Prozent der Probanden eine korrekte Zuweisung zu einer der beiden Gruppen aufgrund der Datenverteilung vorgenommen werden kann. Im Fall gleicher Mittelwerte (T=0) resultiert ein Wert von 50 % richtiger Klassifikationen, was der Zufallserwartung bei zwei Gruppen entspricht.

Berechnung des Wertes U:  $z = T/\sqrt{N}$ . Die Fläche unter der Normalverteilungskurve, von minus Unendlich bis zum Abszissenpunkt z, ergibt die relative Häufigkeit für die richtigen Klassifikationen.

<sup>2</sup> vgl. Sachs, 1972, S. 391 ff.

vgl. Diehl, J.M. und Kohr, H.U., 1977, S. 89 ff.

Die Nullhypothese: die Mittelwerte von Gruppe a und b sind gleich, wird zurückgewiesen, sofern

$$(\bar{x}_a - \bar{x}_b)^2 / (MQ_{\text{inn}} \left( \frac{1}{n_a} + \frac{1}{n_b} \right)) \geq (IGR-1) * F_{(IGR-1, N-IGR, \alpha)}$$

Im Computerausdruck erscheint, zusätzlich zur Häufigkeitsverteilung mit Mittelwert und Standardabweichungen für jede Gruppe für die betreffende Variable folgende Tabelle (im Beispiel für 4 Gruppen):

GRUPPE <sup>1</sup>	SCHEFFE-TEST PAARWEISE MITTELWERTVERGLEICHE		
	KRIT F*(IGR-1) = 7.95		
	1	2	3
1 HAUPTSCHULE			
2 MITTELSCHULE	7.46		
3 GYMNASIUM	6.25	7.80	
4 UNIVERSIT.	9.50	6.80	3.78
	SIGNIF		

Testergebnisse größer oder gleich dem kritischen F-Wert (5 %, 1 % und 0.1 % Niveau je nach Angabe auf der Parameterkarte) multipliziert mit der um 1 verminderten Zahl der Gruppen erscheinen mit dem Hinweis SIGNIF.<sup>2</sup>

Die Ergebnisse des Beispiels besagen: die in den Gruppen 1 und 4 vorliegenden Stichproben entstammen Grundgesamtheiten mit unterschiedlichen Mittelwerten, oder anders ausgedrückt: das statistisch signifikante Testergebnis der Varianzanalyse kommt zustande, weil sich die Gruppen „Probanden mit Hauptschulabschluß“ und Probanden mit Universitätsabschluß“ in der hier zur Betrachtung stehenden Variablen voneinander unterscheiden. Über die Richtung des Unterschiedes geben die Gruppenmittelwerte Auskunft.

<sup>1</sup> Um die Druckzeilen voll auszunutzen, werden an dieser Stelle die Verteilungsparameter Schiefe und Exzeß für jede Gruppe mit aufgeführt (s. Prüfung auf Normalverteilung).

<sup>2</sup> Die Berechnung des kritischen F-Wertes erfolgt über eine Annäherungsfunktion. Insbesondere bei weniger als 100 Freiheitsgraden im Nenner (d.h. Gesamtzahl der Fälle kleiner 100) und hohem Signifikanzniveau (0.1 %) weichen die errechneten kritischen F-Werte von den tatsächlichen stark ab. Die Abweichungen vergrößern sich zusätzlich durch Multiplikation des kritischen F mit der um 1 verminderten Zahl der Gruppen. So errechnet sich beispielsweise  $F(9,50,0.001)$  zu 4.05, während der exakte Wert 3.82 beträgt. Die Unterschiede reduzieren sich bei hohen Freiheitsgraden im Nenner.  $F(9, 100, 0.001)$  errechnet beträgt 3.47, der tatsächliche Wert ist 3.44. Es empfiehlt sich deshalb in den genannten Fällen den im Tabellenkopf ausgedruckten Parameter  $F(KRIT)*(IGR-1)$  anhand der in den meisten Statistikbüchern vorhandenen F-Verteilungen zu überprüfen.

## 7.8. Zusammenfassung der Rohwerte zu Klassen

Für jede Merkmalsausprägung einer Variablen, d.h. für jeden ganzzahligen Rohwert werden die Häufigkeiten in den Untergruppen sowie in der gesamten Stichprobe ermittelt und ausgedruckt. Mitunter ist es aus Gründen der Übersichtlichkeit sinnvoll, die Rohwerte zu Rohwertklassen zusammenzufassen. Dies ist insbesondere bei Skalen der Fall, wo es nicht so sehr interessiert, wie häufig ein einzelner Score auftritt, sondern wie stark bestimmte Klassenintervalle besetzt sind. Das Programm bietet hierzu die Möglichkeit,  $n$  stetig aufeinanderfolgende Merkmalsausprägungen (wobei  $n = 1, 2, \dots, 10$ ) zu jeweils einer Merkmalsklasse zusammenzufassen.

Hierzu ist der Parameter KLASS (Spalte 29 der Parameterkarte) mit 1 zu lochen. Auf der (den) später folgenden Klassenkarte(n) ist für jede Variable anzugeben, wieviele aufeinanderfolgende Merkmalsausprägungen zu einer Klasse vereinigt werden sollen. Jede Spalte dieser Karte(n), deren Nummer der laufenden Nummer einer zu gruppierenden Variablen entspricht, ist mit dem jeweiligen Klassencode zu versehen. Dabei bedeuten die Lochungen 2, 3...9, daß 2, 3...9 Merkmalsausprägungen zusammengefaßt werden. Die Ziffer 0 ist zu lochen, wenn in Zehnerschritten gruppiert werden soll. Diejenigen Spalten der Klassenkarte(n), die nicht zu gruppierende Variablen repräsentieren, bleiben blank oder sind mit der Lochung 1 zu versehen.

Der Gruppierungsalgorithmus läuft folgendermaßen ab: Zuerst wird für die betreffende Variable der kleinste in der gesamten Stichprobe vorkommende Score ermittelt. Ausgehend von diesem Minimalwert werden dann die  $n-1$  (wobei  $n = 2 \dots 10$ , je nach Definition auf der Klassenkarte) aufeinanderfolgenden ganzen Zahlen zur 1. Klasse vereinigt. Der daran anschließende Wert sowie die  $n-1$  auf ihn folgenden Werte ergeben die Klasse 2 usw. Zur Verdeutlichung ein Beispiel:

Angenommen in der Gesamtstichprobe liegt für die Variable mit der laufenden Nr. 25 (dh. die 25. im Format spezifizierte Variable) folgende Rohwertverteilung vor:

Score	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Häufigkeit	0	0	0	3	0	2	1	4	7	3	1	0	2	1	0

Jeweils vier aufeinanderfolgende Merkmalsausprägungen sind zusammenzufassen. Hierzu ist die Spalte 25 der Klassenkarte mit der Lochung 4 zu versehen. Den Minimalwert dieser Variablen bestimmt das Programm mit 3. Dieser Wert und die drei darauffolgenden Merkmalsausprägungen, also die Scores 3,4,5 und 6 bilden die erste Klasse. In die zweite Klasse fallen die Scores 7,8,9,10. Die dritte Klasse ergibt sich aus den Werten 11,12,13 und dem hier nicht vorkommenden Score 14. Als Überschriften werden die Klassenmitten ausgedruckt. Somit ergibt sich folgende Tabelle:

Score	<u>4.5</u>	<u>8.5</u>	<u>12.5</u>
Häufigkeit	6	15	3

Mittelwert und Standardabweichung, die ja in den Häufigkeitstabellen ebenfalls aufgeführt sind, berechnen sich jedoch stets nach den ungruppierten Originaldaten.

Die Zusammenfassung zu Klassen hat folgende Vorteile:

- a) Sofern eine Variable mehr als 15 Merkmalsausprägungen aufweist, reicht die Papierbreite des Schnelldruckers (132 Druckpositionen) nicht mehr aus. Es sind dann zwei bzw. noch mehr Tabellen erforderlich, die untereinander angeordnet sind. Der hierdurch entstehende vermehrte Papierbedarf lässt sich durch geeignete Klassenzusammenlegung reduzieren.
- b) Die zulässige Spannweite richtet sich nach der Zahl der Variablen sowie der Zahl der Unterstichproben. Das bedeutet, je mehr Gruppen und Variablen in die Auswertung eingehen, desto stärker reduziert sich die zulässige Spannweite. Angenommen Variable 9 weist eine Spannweite von 100 auf, zulässig ist aber im konkreten Fall nur eine Spannweite von 30. Die Auswertung dieser Variablen wird dann unterbleiben. An ihrer Stelle erscheint der Hinweis:

VARIABLE 9 SPANNWEITE ZU GROSS. Die Zusammenfassung der Rohwerte zu Klassen bewirkt eine Neuberechnung der Spannweite aufgrund der dann vorliegenden Klassenzahl. Durch Vereinigung von jeweils 4 Werten reduziert sich somit im Beispiel die Spannweite von 100 auf 25. Die Variable kann somit ausgewertet werden.

## 7.9. Prüfung auf Normalverteilung

Eine Reihe statistischer Auswertungsverfahren setzt normalverteilte Daten voraus. Abweichungen von dieser Verteilungsform können, je nach der Sensibilität der jeweiligen Methode gegenüber Verteilungsanomalien, zu Verzerrungen der Ergebnisse führen. Eine Überprüfung dieser Frage erscheint somit, zumindest für die zentralen Variablen einer Untersuchung angebracht.

Der Vergleich einer empirischen mit einer theoretischen Häufigkeitsverteilung läßt sich mit dem Chi-Quadrat-Test durchführen. Dieses Verfahren ist jedoch mit relativ großen Rechenaufwand verbunden. Deshalb werden hier als Maße für die Abweichungen der Verteilung von der Normalverteilung die dritten und vierten Verteilungsmomente, Schiefe und Exzeß herangezogen. Mit dem Schiefewert lassen sich Unregelmäßigkeiten in der Symmetrie feststellen, während der Exzeß darüber Aufschluß gibt, ob die Verteilung zu flach- bzw. zu steilgipfelig ist.<sup>1</sup>

Bei normalverteilten Daten nehmen Schiefe und Exzeß den Wert Null an. Positive Schiefewerte ergeben sich bei linkssteilen, d.h. linksgipfeligen Häufigkeitsverteilungen, negative Werte weisen auf rechtssteile Verteilungen hin. Positiver Exzeß resultiert, sofern die empirische Verteilung im Vergleich zur Normalverteilung einen steileren Gipfel aufweist. Negativer Exzeß stellt sich demnach bei Flachgipfeligkeit ein.

Um die statistische Aussagekraft der errechneten Schiefe- bzw. Exzeßparameter zu ermitteln, werden über die jeweiligen Standardfehler  $S_s$  bzw.  $S_e$  T-Werte errechnet, aus denen sich die Irrtumswahrscheinlichkeit alpha bestimmen läßt. Erreicht bzw. unterschreitet die so gefundene Irrtumswahrscheinlichkeit das vom Benutzer spezifizierte Signifikanzniveau  $p$  (5 %, 1 % und 0.1 % sind als Programmoptionen vorgesehen), so ist der jeweilige

$$\begin{array}{ll}
 \text{Schiefe} = \frac{\sum(x_i - \bar{x})^3}{n s^3} & \text{Exzeß} = \frac{\sum(x_i - \bar{x})^4}{n s^4} - 3 \\
 S_s = \sqrt{\frac{6}{n}} & S_e = \sqrt{\frac{6}{n}} \cdot 2 \\
 T_s = \frac{\text{Schiefe}}{S_s} & T_e = \frac{\text{Exzeß}}{S_e}
 \end{array}$$

vgl. Lienert, G., 1969, S. 172, 173.

Schiefe- bzw. Exzeßwert als signifikant zu betrachten. Die Daten der entsprechenden Variablen folgen in diesem Fall nicht einer Normalverteilung. Anderfalls ist die Nullhypothese normalverteilter Daten beizubehalten.

Um in den Programmzweig der Verteilungsüberprüfung zu gelangen, ist der Parameter IVER (Spalte 28 auf der Parameterkarte) mit 1 zu lochen. Über die später folgende(n) Verteilungskarte(n) (sofern mehr als 80 Variable formatiert wurden, benötigt man zwei Verteilungskarten) wird spezifiziert, für welche Variablen Schiefe und Exzeß zu berechnen sind. Alle Spalten dieser Karte(n), deren Spaltennummer den laufenden Nummern von Variablen entsprechen, für die die Verteilungsform interessiert, sind mit 1 zu lochen.

Der Computerausdruck zeigt dann, zusätzlich zur Häufigkeitsverteilung, folgende Tabelle (im Beispiel für zwei Gruppen):

GRUPPE	SCHIEFE/ T-WERT	ALPHA/ ERGEBNIS	EXZESS/ T-WERT	ALPHA/ ERGEBNIS
1	0.82	0.001	-0.08	0.865
TESTGRUPPE 1	3.43		-0.16	NORMAL
2	0.56	0.001	-0.48	0.127
TESTGRUPPE 2	3.51		-1.52	NORMAL
	0.63	0.000	-0.38	0.150
	4.76		-1.42	NORMAL

Die Tabelle ist folgendermaßen zu interpretieren:

Der Wert für die Schiefe in Gruppe 1 beträgt 0.82. Daraus errechnet sich T zu 3.43. (Die Zahl der Fälle  $n_1$  in der Gruppe 1 ist aus der Häufigkeitstabelle, die hier nicht mit aufgeführt ist, ersichtlich. Im Beispiel beträgt sie 105.) Unter Berücksichtigung von  $n_1 - 1 = 104$  Freiheitsgraden ergibt sich eine Irrtumswahrscheinlichkeit von 0.001. Das Ergebnis ist signifikant, die Verteilung ist linksschief. Normalverteilung kann nicht angenommen werden. Für den Exzeß gelten die entsprechenden Beziehungen. Nicht signifikante Schiefe- bzw. Exzeßwerte sind unter der Rubrik ERGEBNIS mit NORMAL gekennzeichnet.

Die letzte Doppelzeile gibt die Verhältnisse in der Gesamtstichprobe wieder.

### 7.10. Mehrere Rechenläufe – Parameter IQ

Das Paket der Steuerkarten regelt die Analyse eines Datensatzes anhand der angegebenen Gruppenbildung. Ist damit die Auswertung beendet, so ist vor die für den jeweiligen Maschinentyp erforderliche Endkarte (end-of-file) eine Leerkarte zu legen. Sie signalisiert dem Programm das Ende der Berechnungen. Andernfalls erfolgt ein Rücksprung zum Lesebefehl der Parameterkarte. Es lassen sich somit mehrere Auswertungsschritte (Rechenläufe) im gleichen Job durchführen. Dies ist vorteilhaft, weil dadurch erneutes Einlesen des Programmes, Compilieren und Laden vermieden wird. Dazu ist aber ein neues Steuerkartenpaket erforderlich, das entweder, bei Eingabe der Daten mittels Lochkarten, nach der letzten Datenkarte zu liegen kommt, oder, bei Eingabe der Daten von Platte oder Band, an das Steuerkartenpaket des vorherigen Laufs anschließt.

Das Programm sieht für den zweiten und die folgenden Rechenläufe drei Fälle vor:

#### 1. Es wird ein neuer Datensatz gelesen und verarbeitet.

Hier werden also in den verschiedenen Programmläufen unterschiedliche Stichproben ausgewertet. Die Daten befinden sich auf verschiedenen Lochkartensätzen bzw. auf einem Platten- oder Bandspeicher mit unterschiedlichen logischen Filenamen bzw. Data-Set-Namen.

#### a) Eingabe der Daten über Lochkarten

Abfolge der Steuerkarten:

- |         |   |
|---------|---|
| 1. Lauf | (1) Parameterkarte (IT = blank oder 5)              |
|         | (2) Formatkarte(n) (F-Format)                       |
|         | (3) ev. Formatkarte(n) für Ausgangsdaten (A-Format) |
|         | (4) ev. Missing Data Karte(n)                       |
|         | (5) ev. Textkarten für Variablenbezeichnungen       |
|         | (6) Gruppenkarten und Gruppentextkarten             |
|         | (7) ev. Klassenkarte(n)                             |
|         | (8) ev. Varianzanalysekarte(n)                      |
|         | (9) ev. Verteilungskarte(n)                         |
|         | (10) Datenkarten                                    |

Für jeden weiteren Lauf Abfolge (1) bis (10)

## b) Eingabe der Daten von Platte oder Band

Der Parameter IT auf der Parameterkarte enthält hier die logische File Nummer der Platten- bzw. Banddatei. Die verschiedenen Rechenläufe müssen durch unterschiedliche File-Nummern IT gekennzeichnet sein.<sup>1</sup>

Abfolge der Steuerkarten:

1. Lauf (1) Parameterkarte (IT=10 oder jede andere Nummer zwischen 1 und 99 außer 5,6,8 u. 9)  
(2) bis (9)
  2. Lauf (1) Parameterkarte (IT=11)  
(2) bis (9)
  3. Lauf (1) Parameterkarte (IT=12)  
(2) bis (9)
- usw.

2. Es werden **neue Daten** verarbeitet, die sich aber im **gleichen Datensatz** wie im Lauf vorher befinden.

Wie bereits geschildert, können in einem Rechenlauf maximal 100 Variable gleichzeitig verarbeitet werden. Beinhaltet eine Stichprobe eine größere Variablenzahl je Proband bzw. Fall, so muß sie in Unterabschnitte mit jeweils maximal 100 Variable gegliedert werden. Dieser Fall sei an einem Beispiel verdeutlicht:

Angenommen es handelt sich um eine Stichprobe mit 150 Variablen je Proband, die auf jeweils 3 Lochkarten untergebracht sind. Es gelte folgender Codeplan:

---

<sup>1</sup> Bei IBM-Anlagen sähe die Zuordnung der log. File-Nummern zu den ihnen entsprechenden Dateien folgendermaßen aus:  
 //GO.FT10F001 DD UNIT=SYSDA,VOL=SER=...,DISP=OLD,  
 // DSN=DATEI1  
 //GO.FT11F001 DD UNIT=SYSDA,VOL=SER=...,DISP=OLD,  
 // DSN=DATEI2  
 bzw. sofern das Programm bereits in binärer Form auf einen externen Speicher vorliegt:  
 //FT10F001 DD UNIT=SYSDA,VOL=SER=...,DISP=OLD,DSN=DATEI1  
 //FT11F001 DD UNIT=SYSDA,VOL=SER=...,DISP=OLD,DSN=DATEI2

- Karte 1 Spalten 1- 4 Kartenkennung und laufende Nummer der Probanden  
5-80 insgesamt 76 einspaltige Variable
- Karte 2 Spalten 1- 4 Kartenkennung und laufende Nummer des Probanden  
5-24 insgesamt 10 je zweispaltige Variable  
25-80 56 einspaltige Variable
- Karte 3 Spalten 1- 4 Kartenkennung und laufende Nummer des Probanden  
5-12 insgesamt 8 je einspaltige Variable.

Die erste Variable, die sich auf Karte 1 Spalte 5 befindet, diene zur Gruppenbildung. Die restlichen Items sollen gegen sie ausgezählt werden.

#### a) Eingabe der Daten über Lochkarten

Sofern die Daten nur auf Lochkarten vorliegen, ergeben sich Probleme, da sie ab dem zweiten Rechenlauf mit einer neuen Formatanweisung wieder gelesen werden müssen. Abhilfe läßt sich schaffen, indem man

- entweder die Daten vorher — über die jeweiligen Systemroutinen oder per Programm — auf einen Platten- oder Bandspeicher bringt (Permanent File), Eingabe s. unter 2 b,
- oder sie mittels der im Programm vorgesehenen SUBROUTINE DATV auf Platte zwischenspeichert (Scratch File). Diese Möglichkeit steht allerdings nur bei Eingabe des Programms in Form eines Kartendecks offen. Die hierfür benötigten Programmstatements sind vom Benutzer zu erstellen und in die sich am Ende des Programms befindliche SUBROUTINE DATV einzufügen.

Das Unterprogramm läßt sich folgendermaßen standardisieren:

```

SOUBROUTINE DATV +)
DIMENSION KK(200) → (für maximal 10 Karten je Proband)
K = ..... → (Zahl der Karten je Proband bzw. Fall)
N = ..... → (Zahl der Probanden in der Stichprobe)
L = K * 20
DO 10 I = 1,N
READ (5,1) (KK(J),J=1,L)
1 FORMAT (20A4)
10 WRITE (10,1) (KK(J),J=1,L)
REWIND 10
RETURN +)
END +)

```

Die mit +) gekennzeichneten Karten befinden sich bereits im Programm.

**Abfolge der Steuerkarten:**

1. Lauf (10) Datenkarten (SUBROUTINE DATV wird vor den Lesebefehlen der Steuerkarten aufgerufen)
- (1) Parameterkarte (IT=10)
  - (2) Formatkarte(n); im Beispiel  
(4X,76F1.0/4X,10F2.0,14F1.0/)
  - (3) bis (9)
2. Lauf (1) Parameterkarte (IT=10,IREW=1)
- (2) Formatkarte(n); im Beispiel:  
(4X,F1.0/38X,42F1.0/4X,8F1.0)
  - (3) bis (9)

entsprechend für alle weiteren Läufe.

Der Parameter IREW muß ab dem 2. Lauf auf 1 gesetzt werden, da der externe Datenspeicher für jeden neuen Lauf auf den Anfang der Datei zurückgesetzt werden muß.

**b) Eingabe der Daten von Platte oder Band****Abfolge der Steuerkarten:**

1. Lauf (1) Parameterkarte (IT=10)
- (2) Formatkarte(n) (wie im Beispiel 2a, 1. Lauf)
  - (3) bis (9)
2. Lauf (1) Parameterkarte (IT=10,IREW=1)
- (2) Formatkarte(n) (wie im Beispiel 2a, 2. Lauf)
  - (3) bis (9)

entsprechend für alle weiteren Läufe.

3. Es werden die gleichen Daten verarbeitet wie im Lauf vorher, sie werden nur anders gruppiert bzw. eine oder mehrere andere Variable aus dem gleichen Datensatz dienen zur Bestimmung der Gruppen.

Mitunter sollen die Daten hinsichtlich verschiedener gruppenbestimmender Variablen nacheinander quantifiziert werden. So will man etwa die im Beispiel angeführten Daten nicht nur gegen die erste, sondern auch noch gegen die zweite Variable auszählen. Dieser Fall läßt sich prinzipiell genauso behandeln wie Fall 2. Dies wäre jedoch mit unnötigem Arbeitsaufwand verbunden, insbesondere wenn Textkarten für die einzelnen Items gelesen

werden. Die Textkarten wären dann nämlich jedem Steuerkartenpaket beizugeben. Durch den Parameter IQ (Spalte 35 der Parameterkarte) läßt sich dies vermeiden. Ein IQ von 1 signalisiert dem Programm, daß für den speziellen Programmablauf weder Format-, noch Missing Data- noch Textkarten zu lesen sind (also Karten (2) bis (5)), weil diese inhaltlich identisch mit den Angaben des vorherigen Laufes sind.

#### a) Eingabe der Daten über Lochkarten

In diesem Fall müssen die Daten nicht wie bei 2a vom Benutzer auf eine externe Einheit geschrieben werden. Dies wird hier vom Programm besorgt.

Abfolge der Steuerkarten:

1. Lauf
  - (1) Parameterkarte (IT=blank oder 5)
  - (2) Formatkarte(n) (wie im Beispiel 2a, 1.Lauf)
  - (3) bis (9)
  - (10) Datenkarten
2. Lauf: Auszählung gegen Variable 2
  - (1) Parameterkarte (IT=blank, IQ=1)
  - (6) bis (9)

Für ev. weitere Rechenläufe gilt die gleiche Abfolge der Steuerkarten wie im Lauf 2. Man erspart sich im Beispiel dadurch das nochmalige Ablocken von 100 Textkarten.

#### b) Eingabe der Daten von Platte oder Band

Abfolge der Steuerkarten:

1. Lauf
  - (1) Parameterkarte (IT=10)
  - (2) Formatkarte(n) (wie im Beispiel 2a)
  - (3) bis (9)
2. Lauf
  - (1) Parameterkarte (IT=10,IQ=1)
  - (6) bis (9)

usw.

Soll für die restlichen 50 Variablen des Beispiels ebenso verfahren werden, so ergeben sich 2 weitere Rechenläufe:

---

(Anmerkung: Es hat keine Auswirkung auf die Ergebnisse, wenn in den genannten Beispielen der Parameter IREW immer auf 1 gesetzt wird. Es gibt jedoch Fälle, wo ein Zurücksetzen der Eingabeeinheit auf den Datenanfang unerwünscht ist.)

3. Lauf: Auszählung der Variablen 101 bis 150 gegen Variable 1  
(entspricht 2b)
- (1) Parameterkarte (IT=10,IQ=blank,IREW=1)
  - (2) Formatkarte(n); im Beispiel:  
(4X,2F1.0/38X,42F1.0/4X,8F1.0)
  - (3) bis (9)
4. Lauf: Auszählung der Variablen 101 bis 150 gegen Variable 2:
- (1) Parameterkarte (IT=10,IQ=1)
  - (6) bis (9)

### 7.11. Alpha-numerische Bezeichnungen der Variablen (Variable Labels)

Um sich im Programmausdruck leichter zurechzufinden, empfiehlt es sich, die einzelnen Tabellen mit alpha-numerischen Bezeichnungen, sog. Variable Labels zu versehen. Diese Labels sind in Form von Textkarten dem Paket der Programmsteuerkarten beizugeben. Der Parameter ITX (Spalte 25 der Parameterkarte) ist in diesem Fall mit 1 zu lochen. Das Zusammenstellen der Textkarten kann, zumal bei Auswertung zahlreicher Variablen, mitunter mühsam und zeitaufwendig werden. Sofern ein Datensatz häufiger mit QUER3 bearbeitet werden soll, empfiehlt es sich deshalb, die Textkarten *für sämtliche Variablen* der Stichprobe auf einen externen Speicher zu bringen (Permanent File). Bei entsprechender Spezifikation des Parameters ITX (Lochung 2) werden die Labels der in der aktuellen Auswertung berücksichtigten Variablen von diesem Speicher mit der logischen File-Nummer 15 gelesen und in den Tabellen ausgedruckt.

Die Auswahl der jeweiligen Labels erfolgt nicht anhand der laufenden Variablen-Nummern, sondern orientiert sich am Eingabeformat für die Daten (bzw. dem Eingabeformat für die Ausgangsdaten). Aus diesem Grunde genügt es nicht, das Textkartenpaket mittels der Übertragungsroutinen des jeweiligen Betriebssystems auf den Speicher bringen zu lassen: vielmehr ist hierfür ein eigenes kurzes Programm (Programm TEXT) erforderlich, das im Anhang aufgelistet ist. Die Übertragung des Textes auf das Permanent File wird im folgenden Beispiel erläutert:

Der Stichprobe soll folgender Codeplan zugrundeliegen:

Karte 1: Spalte           1    Kartenkennung  
                           2 - 4   laufende Nummer des Probanden  
                           5    Geschlecht  
                           6 - 7   Alter  
                           8    Schulabschluß  
                           9 - 10   blank  
                          11 - 80   70 einspaltige Variable

Karte 2: Spalte           1    Kartenkennung  
                           2 - 4   laufende Nummer  
                           5 - 10   blank  
                          11 - 30   10 zweispaltige Variable

Da über die Kartenkennung und laufende Nummer sicherlich keine Häufigkeitsauszählungen durchzuführen sind, beträgt die Zahl der insgesamt in die Stichprobe aufgenommenen Variablen 83. Die Labels für diese 83 Variablen sind zu lochen, wobei für jede Variable eine Karte mit 80 Spalten zur Verfügung steht. Dem Programm TEXT ist die Zahl der Variablen mitzuteilen, sowie die Zahl der erforderlichen Formatkarten (Hier: 1). Darauf folgt die Formatierung der Variablen mittels der A-Spezifikation, im Beispiel (4X,A1,A2,A1,2X,70A1/10X,10A2). Die Textkarten werden sodann in eine Kernspeichermatrix KK(Zahl der Variablen, 80) eingelesen. Der Übertragungsvorgang auf das Permanent File erfolgt über das A-Format derart, daß die einzelnen Zeichen eines Labels immer an der gleichen durch das Format bezeichneten, Stelle des logischen Records<sup>1</sup> zu stehen kommen. Die Labels werden folgendermaßen gespeichert:

Speicherstelle (entspricht der Spalte auf den Datenkarten)

	1	2	3	4	5	6	7	8	9	10	11	...	80/	1	2	3	...	10	11	12	13	...	29	30	...	80
Log.Record 1					G	A		S			V	...	V						V	V	V					
2					E	L		C			A		A						A	A	A					
3					S	T		H			R		R						R	R	R					
4					C	E		U			I		I						I	I	I					
5					H	R		L			A		A						A	A	A					
6					L			A			B		B						B	B	B					
7					E			B			L		L						L	L	L					
8					C			S			E		E						E	E	E					
9					H			C					K						L	M	Z					
10					T			H			A															
11								L																		
12								U																		
13								S																		
14								S																		
...																										
80																										

<sup>1</sup> Für IBM-Anlagen: Die physikalische Recordlänge LRECL im DCB-Parameter muß 80 betragen

Sollen nun mit QUER3 beispielsweise die Variablen der 2. Karte gegen das Geschlecht ausgezählt werden, so ergibt sich das folgende Eingabeformat für Daten: (4X,F1.0/10X,10F2.0). Dieses wird programmintern in ein A-Format umgewandelt: (4X,A1/10X,10A2). Die auf den entsprechenden Speicherstellen des Permanent Files sich befindlichen Zeichen werden gelesen und durch Kippen der Matrix wieder in die ursprüngliche Reihenfolge gebracht. Mit diesem Datensatz lassen sich jetzt beliebig viele Auswertungsschritte vornehmen, ohne daß immer wieder Textkarten einzugeben sind.

## 8. PROGRAMMVERSION A

Rechner, bei denen im F-Format gelesene Blanks (ungelochte Felder) intern mit -0. dargestellt werden, benötigen die im Programm QUER3 eingeschlagene zeitaufwendige Strategie der Missing Data Verarbeitung nicht. Eine Abfrage auf Blanks läßt sich hier direkt mit der SIGN-Funktion durchführen. Sofern die Abfrage: IF (X .EQ. 0. .AND. SIGN(1.,X) .LT. 0.) als „wahr“ erkannt wird, handelt es sich bei der Variablen mit der Bezeichnung X um ein ungelochtes Feld.

Damit entfallen eine ganze Reihe von Programmstatements. Die entsprechenden Karten, deren Label im folgenden aufgeführt sind, werden mit DELETE bezeichnet. Sie sind aus dem Programmdeck ersatzlos zu entfernen. Karten, die zu verändern sind, erhalten den Hinweis CHANGE sowie den neuen Inhalt der jeweiligen Programmkarte. Sind mehrere aufeinanderfolgende Karten auszuscheiden, wird dies der Kürze halber mit QUExxxxx bis QUEyyyyy DELETE bezeichnet. Alle Karten ab Labelnummer xxxxx (einschl.) bis yyyy (einschl.) sind aus dem Programmpaket herauszunehmen.

Die Parameter KDAT und JFO haben in der Programmversion A keine Wirkung mehr. Damit entfallen auch Formatkarten für Ausgangsdaten sowie das Scratch File mit der logischen Filenummer 9.

LABEL	Veränderung	ev. neuer Karteninhalt
QUE00090	CHANGE	DIMENSION AMIS(100),I19(10), FMA(400)
*QUE00160 bis QUE00190	DELETE	
QUE00760 bis QUE00780	DELETE	
QUE01340 bis QUE01400	DELETE	
QUE01420 bis QUE01450	DELETE	
QUE01460	CHANGE	READ (5,FM) (AMIS(J),J=1,MV)
QUE01480	CHANGE	IF(AMIS(J).EQ.0..AND.SIGN(1., AMIS(J)).LT.0.)AMIS(J) = FBLANK
QUE01490	DELETE	
QUE01510	DELETE	
QUE01544 bis QUE01546	DELETE	
QUE01570	DELETE	
QUE01620	CHANGE	READ(5,6) NO(K),GU(J,K),GO(J,K), G3(J,K),G4(J,K),BTEXT(J,K)
QUE01630	CHANGE	6 FORMAT (I4, 1X,4F5.0,1X,A8)
QUE01650	CHANGE	IF(G3(J,K).EQ.0..AND.SIGN(1., G3(J,K)).LT.0.)G3(J,K) = FBLANK
QUE01660	CHANGE	IF(G4(J,K).EQ.0..AND.SIGN(1., G4(J,K)).LT.0.)G4(J,K) = FBLANK
QUE01680 bis QUE01780	DELETE	
QUE02740 bis QUE02840	DELETE	
QUE02860	CHANGE	READ(IT,FM) (X(J), J = 1,MV)
QUE02870	DELETE	
QUE02900 bis QUE02940	DELETE	
QUE03190	DELETE	
QUE09970 bis QUE10120	DELETE	
*QUE10330	DELETE	

Die mit \* gekennzeichneten Programmstatements haben nichts mit der Missing-Data Verarbeitung zu tun. Sie sind nur dann zu entfernen, falls die für den jeweiligen Maschinentyp verwendete FORTRAN-Version die Statements REAL\*8 und INTEGER\*2 zur Umdefinition der Wortlänge nicht zulässt (z.B. bei CDC-Rechnern).

## **9. WEITERGABE DES PROGRAMMS**

Das FORTRAN-Quellenprogramm kann von öffentlichen Institutionen unter folgenden Bedingungen bezogen werden:

- es wird nicht ohne Erlaubnis an Dritte weitergegeben;
- Veränderungen werden nur nach Absprache mit dem Programmautor vorgenommen;
- die durch den Versand entstehenden Kosten sowie die Kosten für Lochkarten und Rechenzeit für Testläufe müssen von der anfordernden Institution übernommen werden.

Obwohl das Programm sorgfältig ausgetestet wurde, kann keine Garantie für fehlerfreien Lauf gegeben werden.

**LITERATURVERZEICHNIS**

- Bredenkamp, J.*, Über Maße der praktischen Signifikanz, in *Zs für Psychologie*, Bd. 177, 1969
- , *Der Signifikanztest in der psychologischen Forschung*, Frankfurt/M., 1972
- Diebl, J.M., Kobr, H.-U.*, *Durchführungsanleitungen für statistische Test*, Weinheim, 1977
- Hays, W., L.*, *Statistics*, London, 1970
- Lienert, G.*, *Testaufbau und Testanalyse*, Weinheim, 1969
- Nie, N.H., Hull, C.H., Jenkins, J.G., Steinbrenner, K. & Bent, D.H.*, *Statistical Package for the Social Sciences*, 3. Aufl., McGraw Hill, 1975
- Sachs, L.*, *Statistische Auswertungsmethoden*, 3. Aufl., Berlin, 1971
- Veldman, D.J.*, *FORTTRAN Programming for the Behavioral Sciences*, New York, 1967

```

C QUER3 MIT SCHIEFE UND EXZESS
C BENOETIGTE ZWISCHENSPEICHER PLATTENEINHEITEN 8 UND 9
C FUER CDC REAL*8 UND INTEGER*2 ENTFERNEN AUS HAUPTPROGRAMM, SUBROUTINES
C WAND, AFORM, SCHEX
  DIMENSION FM(400),TEXT(100,80),TX(20,2),IV(100),
  AKL(100),X(101),SU(20,100),SU2(20,100),AM(20),SMALL(100),
  BBIG(100),ZU(100),KU(200),NN(20),H(15),ISP(200),KVAR(100),FE(10,10)
  C, FMI(18),FM2(18),GVAR(100),JH(15),KS(100),TE(6)
  DIMENSION AMIS(100),BMIS(100),FMA(400),Y(101),I19(10)
  DIMENSION SCHEF(10,10),KES(100)
  DIMENSION IH(40000)
  DIMENSION GU(20,10),GO(20,10),G3(20,10),G4(20,10),NO(20),
  ABTEXT(20,10),LI(20),TY(20,17),MAXZ(10),MAXS(10),NR(20,10)
  DIMENSION IES(100),XM(400),X3(20,100),X4(20,100),EX(21),SC(21),TS(
  A21),TEX(21),SSS(21),SEE(21),ALS(21),AEE(21),TESCH(21),TEEX(21)
  REAL*8 TESCH,TEEX
  REAL*8 TX,FMI,FM2,FZ,STR,ALEER,TE
  REAL*8 G5,G6,BTEXT
  INTEGER*2 IH
  DIMENSION MA(15),MG(200),AB(15),AD(15),AC(15)
  COMMON S,NN
  COMMON/ALF/ALPHA
  DATA ALEER,BLEER,LEER/8H ,4H ,1H /
  DATA CLEER,DLEER/2H-0,3H -0/
  DATA I19/IH1,IH2,IH3,IH4,IH5,IH6,IH7,IH8,IH9,IH0/
  DATA JAA,NEIN/4H JA ,4HNEIN/
  DATA FMI(1),FMI(2),FMI(18),FZ/5H(1H+ ,4H42X,1H),7H6(1H-),/
  DATA FM2(1),FM2(2),FM2(18),STR/5H(1H+ ,4H42X,1H),7H6(1H=),/
  DATA SIG1,SIG2/4HSIGN,2HIF/
  DATA IDMAX,MDMAX,JDMAX/20,100,200/
  NDIM = 40000
  FBLANK = -1.0E06
  CALL DATV
  NQ = 0
500 READ (5,1) NP,MV,IT,IFO,MISD,KDAT,JFO,ITREN,IGR,ITX,IVAR,IVER,
  AKLASS,IAL,IREL,JREL,IQ,IREW,(TE(J),J=1,6)
  1 FORMAT (I5,1X,I3,1X,I2,2(IX,I1),2I1,2(I2,IX),I1,IX,7I1,2(IX,I1),
  A5A8,A3)
  IF (NQ .EQ. 0) IQ = 0
  NQ = NQ +1

```

```

QUE00010
QUE00020
QUE00030
QUE00040
QUE00050
QUE00060
QUE00070
QUE00080
QUE00090
QUE00100
QUE00110
QUE00120
QUE00130
QUE00140
QUE00150
QUE00160
QUE00170
QUE00180
QUE00190
QUE00200
QUE00210
QUE00220
QUE00230
QUE00240
QUE00250
QUE00260
QUE00270
QUE00280
QUE00290
QUE00300
QUE00310
QUE00320
QUE00330
QUE00340
QUE00350
QUE00360
QUE00370
QUE00380
QUE00390
QUE00400

```

```

IF (NP .LE. 0 .AND. MV .LE. 0) GO TO 600
DO 58 J = 1, MV
KVAR(J) = 0
KL(J) = 1
58 IV(J) = 0
IF (IT .LE. 0) IT = 5
IF (IT .EQ. 5) IREW = 0
IF (IREW .GT. 0) REWIND IT
ALPHA = .05
IF (IAL .EQ. 1) ALPHA = .01
IF (IAL .EQ. 2) ALPHA = 0.001
WRITE (6,2) (TE(J), J=1,6), NP, MV, IT, IFO, MISD, ITREN, IGR, ITX, IVAR,
AKLASS, IREL, JREL, KREL, IQ, IREW, IVER, KDAT
2 FORMAT (1H1, 15HQVERAUSAEHLUNG, 3X, 5A8, A3/
A1H, 41H** SOZIALWISS. INSTITUT DER BUNDESWEHR **//
11H0, 20HZAHL DER FAELLE =, I5/1H, 20HZAHL DER VARIABLEN =, I5/
21H, 20HEINGABEEINHEIT =, I5/1H, 20HZAHL FORMATKARTEN =, I5/
31H, 20MISSING DATA CODE =, I5/1H, 20HZAHL GRUPPENVAR. =, I5/
41H, 20HZAHL DER GRUPPEN =, I5/1H, 20HTEXTKARTEN =, I5/
51H, 20HVARIANZANALYSE =, I5/1H, 20HKLASSENZUSAMMENFAS. =, I5/
51H, 20HRELATIVWERTE ZEILE =, I5/1H, 20HRELATIVWERTE SPALTE =, I5/
71H, 20HRELATIVWERTE GESAMT =, I5/1H, 20HWIEDERHOL. FAKTOR IQ =, I5/
81H, 20HREWIND INPUT TAPE =, I5/1H, 20HSCHEIFE U. EXZESS =, I5/
91H, 20HKDAT =, I5//)
WRITE (6,75)
75 FORMAT (1H0, 50HFOLGEMDE PROGRAMMSTEUERKARTEN WERDEN NEBEN DER PAR,
A20HAMETERKARTE ERWARTET/1H, 4HZAH, 2X, 9HKARTENART/1H, 55(1H-))
IA = IVAR + IVER + KLASS
IE = 1
IF (MV .GT. 80) IE = 2
IF (IQ .EQ. 1) GO TO 101
JJ = MAX0(1, IFO)
WRITE (6,76) JJ
76 FORMAT (1H, 13, 3X, 43HFORMATKARTE(N) FUER EINGABEDATEN (F-FORMAT))
JJ = MAX0(1, JFO)
IF (KDAT .EQ. 1) WRITE (6,77) JJ
77 FORMAT (1H, 13, 3X, 44HFORMATKARTE(N) FUER AUSGANGSDATEN (A-FORMAT))
IF (MISD .EQ. 2) WRITE (6,78)
78 FORMAT (1H, 6X, 76HMISSINGDATAKARTE(N) (SO VIELE WIE DATENKARTEN JE
QUE00410
QUE00420
QUE00430
QUE00440
QUE00450
QUE00460
QUE00470
QUE00480
QUE00490
QUE00500
QUE00510
QUE00520
QUE00530
QUE00540
QUE00550
QUE00560
QUE00570
QUE00580
QUE00590
QUE00600
QUE00610
QUE00620
QUE00630
QUE00640
QUE00650
QUE00660
QUE00670
QUE00680
QUE00690
QUE00700
QUE00710
QUE00720
QUE00730
QUE00740
QUE00750
QUE00760
QUE00770
QUE00780
QUE00790
JEQUE00800

```

```

A FALL IN DATEI EINGABEDATEN))
IF (ITX .EQ. 1) WRITE (6,79) MV
79 FORMAT (1H ,I3,3X,30TEXTKARTEN FUER VARIABLENNAMEN)
101 CONTINUE
IF (ITREN .LE. 0) GO TO 102
IF (IGR .LE. 0) IGR = 1
JJ = IGR*ITREN + IGR
WRITE (6,92) JJ
92 FORMAT (1H ,I3,3X,30HGRUPPENTEXT- UND GRUPPENKARTEN)
102 CONTINUE
IF (KLASS .EQ. 1) WRITE (6,94) IE
94 FORMAT (1H ,I3,3X,15HKLASSENKARTE(N))
IF (IVAR .GT. 0) WRITE (6,96) IE
96 FORMAT (1H ,I3,3X,22HVARIANZANALYSEKARTE(N))
IF (IVER .EQ. 1) WRITE (6,97) IE
97 FORMAT (1H ,I3,3X,19HVERTEILUNGSKARTE(N))
IF (IT .EQ. 5) WRITE (6,99) NP
99 FORMAT (1H ,6X,16HDATENKARTEN FUER,I6,10H PROBANDEN)
WRITE (6,95) ALPHA
95 FORMAT (1H0,5IHSIGNIFIKANZNIVEAU FUER SCHIEFE UND EXZESS BZW FUER
A,15HSCHIEFE-TESTS -,F6.3/)
IF (IGR .LE. 1 .OR. ITREN .GT. 0) GO TO 502
WRITE (6,7)
7 FORMAT (1H0,64HITREN (SP 19-20) MUSS GROESSER NULL SEIN. LAUF WIRD
A ABGEBROCHEN.)
IF (IREW .EQ. 1) REWIND IT
STOP
502 CONTINUE
IF (IGR .GE. 2) GO TO 501
IGR = 1
IDIM = 1
501 CONTINUE
IDIM = MAX0(1,IGR)
MDIM = MV
IF (IDIM .LE. IDMAX) GO TO 59
WRITE (6,32)
32 FORMAT (1H ,42HZAHL DER GRUPPEN ZU GROSS. PROGRAMMABBRUCH)
GO TO 600
59 IF (MDIM .LE. MDMAX) GO TO 62
WRITE (6,34)

```

```

QUE00810
QUE00820
QUE00830
QUE00840
QUE00850
QUE00860
QUE00870
QUE00880
QUE00890
QUE00900
QUE00910
QUE00920
QUE00930
QUE00940
QUE00950
QUE00960
QUE00970
QUE00980
QUE00990
QUE01000
QUE01010
QUE01020
QUE01030
QUE01040
QUE01050
QUE01060
QUE01070
QUE01080
QUE01090
QUE01100
QUE01110
QUE01120
QUE01130
QUE01140
QUE01150
QUE01160
QUE01170
QUE01180
QUE01190
QUE01200

```

```

34 FORMAT (1H ,44HZAHZL DER VARIABLEN ZU GROSS. PROGRAMMABBRUCH)
GO TO 600
62 CONTINUE
IF (IQ .EQ. 1) GO TO 50
DO 65 J = 1,MV
DO 65 K = 1,80
65 TEXT(J,K) = BLEER
IF0 = 80*MAX0(1,IF0)
READ (5,3) (FM(J),J=1,IF0)
WRITE (6,4) (FM(J),J=1,IF0)
3 FORMAT (80A1)
4 FORMAT (1H0,30HEINGELESENES FORMAT FUER DATEN/(1H ,80A1))
CALL AFORM (FM,FMA,IF0)
IF (KDAT .LE. 0) GO TO 705
JFO = 80*MAX0(1,JFO)
READ (5,3) (XM(J),J=1,JFO)
WRITE (6,701) (XM(J),J=1,JFO)
701 FORMAT (/1H0,42HEINGELESENES FORMAT FUER DIE AUSGANGSDATEN/
A(1H ,80A1))
705 CONTINUE
IF (MISD .NE. 2) GO TO 1050
REWIND 9
READ (5,FMA) (BMIS(J),J=1,MV)
WRITE (9,FMA) (BMIS(J),J=1,MV)
REWIND 9
READ (9,FM) (AMIS(J),J=1,MV)
DO 1049 J = 1,MV
IF (AMIS(J) .NE. 0.) GO TO 1049
IF (BMIS(J) .EQ. BLEER) AMIS(J) = FBLANK
1049 CONTINUE
REWIND 9
1050 CONTINUE
5 IF (ITX .EQ. 1) READ (5,5) ((TEXT(I,J),J=1,80),I=1,MV)
FORMAT (80A1)
IF (ITX .NE. 2) GO TO 1210
REWIND 15
DO 1201 J = 1,80
IF (KDAT .LE. 0) GO TO 1204
READ (15,XM) (TEXT(I,J),I=1,MV)
GO TO 1201
QUE01210
QUE01220
QUE01230
QUE01240
QUE01250
QUE01260
QUE01270
QUE01280
QUE01290
QUE01300
QUE01310
QUE01320
QUE01330
QUE01340
QUE01350
QUE01360
QUE01370
QUE01380
QUE01390
QUE01400
QUE01410
QUE01420
QUE01430
QUE01440
QUE01450
QUE01460
QUE01470
QUE01480
QUE01490
QUE01500
QUE01510
QUE01520
QUE01530
QUE01540
QUE01541
QUE01542
QUE01543
QUE01544
QUE01545
QUE01546

```

```

1204 READ (15,FMA) (TEXT(I,J),I=1,MV)
1201 CONTINUE
1210 CONTINUE
50 CONTINUE
IF (ITREN .LE. 0) GO TO 55
REWIND 9
DO 1052 J = 1,IGR
READ (5,45) TX(J,1),TX(J,2),(TY(J,JJ),JJ=1,17)
45 FORMAT (A8,A7,16A4,A1)
DO 1052 K = 1,ITREN
READ (5,6) NO(K),GU(J,K),GO(J,K),G5,G6,BTEXT(J,K)
6 FORMAT (I4,1X,2F5.0,2A5,1X,A8)
NR(J,K) = NO(K)
WRITE (9,49) G5,G6,G5,G6
49 FORMAT (4A5)
1052 CONTINUE
REWIND 9
DO 1053 J = 1,IGR
DO 1053 K = 1,ITREN
READ (9,54) G5,G6,G7,G8
54 FORMAT (2A5,2F5.0)
IF (G5 .EQ. ALEER ) G7 = FBLANK
IF (G6 .EQ. ALEER ) G8 = FBLANK
G3(J,K) = G7
G4(J,K) = G8
1053 CONTINUE
REWIND 9
WRITE (6,21)
21 FORMAT ( 1H1,43HDIE GRUPPEN WERDEN FOLGENDERMASSEN GEBILDET/1H0,
A6HGRUPPE,2X,15HKURZBEZEICHNUNG/1H ,23(1H-))
DO 1062 J = 1,IGR
WRITE (6,64) J,TX(J,1),TX(J,2),(TY(J,JJ),JJ=1,17)
64 FORMAT (1H ,I3,5X,A8,A7,2X,16A4,A1)
1062 CONTINUE
WRITE (6,900)
900 FORMAT (1H0,6HGRUPPE,3X,3HVAR,14X,3HVON,4X,3HBIS,2X,14HZUSAETZL.CO
ADES/1H ,51(1H-))
DO 1061 J = 1,IGR
WRITE (6,901)
901 FORMAT (1H )

```

QUE01547  
 QUE01548  
 QUE01549  
 QUE01550  
 QUE01560  
 QUE01570  
 QUE01580  
 QUE01590  
 QUE01600  
 QUE01610  
 QUE01620  
 QUE01630  
 QUE01640  
 QUE01650  
 QUE01660  
 QUE01670  
 QUE01680  
 QUE01690  
 QUE01700  
 QUE01710  
 QUE01720  
 QUE01730  
 QUE01740  
 QUE01750  
 QUE01760  
 QUE01770  
 QUE01780  
 QUE01790  
 QUE01800  
 QUE01810  
 QUE01820  
 QUE01830  
 QUE01840  
 QUE01850  
 QUE01860  
 QUE01870  
 QUE01880  
 QUE01890  
 QUE01900  
 QUE01910

```

DO 1060 K = 1, ITREN
N = NR(J,K)
IF (J .EQ. 1) LT(K) = NR(J,K)
WRITE (6,67) J, NR(J,K), BTEXT(J,K), GU(J,K), GO(J,K),
A(TEXT(N,LL), LL=1,80)
67 FORMAT (1H , I5, 4X, I3, 2X, A8, 1X, F6.0, 1X, F6.0, 16X, 80A1)
IF (G3(J,K) .GT. FBLANK) WRITE (6,68) G3(J,K)
68 FORMAT (1H+, 38X, F6.0)
69 IF (G4(J,K) .GT. FBLANK) WRITE(6,69) G4(J,K)
1060 CONTINUE
IF (J .EQ. 1) GO TO 1061
DO 1064 K = 1, ITREN
IF (LT(K) .NE. NR(J,K)) GO TO 1065
1064 CONTINUE
GO TO 1061
1065 WRITE (6,63)
63 FORMAT (/1H0,100HDIE NUMMERN DER TRENNVARIALBEN STIMMEN IN DEN VER
ASCHIEDENEN GRUPPEN NICHT UEBEREIN. PROGRAMMABBRUCH.)
GO TO 600
1061 CONTINUE
55 CONTINUE
DO 665 J = 1, MV
IES(J) = 0
665 KVAR(J) = 0
IF (IA .GT. 0) WRITE (6,44)
44 FORMAT (/1H0,30X,1H1,7X,2H10,8X,2H20,8X,2H30,8X,2H40,8X,2H50,8X,
A2H60,8X,2H70,8X,2H80,8X,2H90,7X,3H100/1H ,30X,10(10H-----+))
IF (KLASS .NE. 1) GO TO 57
READ (5,3) (KL(J), J=1,MV)
WRITE (6,51) (KL(J), J=1,MV)
51 FORMAT (1H ,15HKLASSENKARTE(N),15X,100A1)
DO 56 J = 1,MV
IF (KL(J) .EQ. LEER) GO TO 53
DO 1153 JJ = 1,10
IF (KL(J) .NE. I19(JJ)) GO TO 1153
KL(J) = JJ
GO TO 56
1153 CONTINUE
53 KL(J) = 1

```

QUE01920

QUE01930

QUE01940

QUE01950

QUE01960

QUE01970

QUE01980

QUE01990

QUE02000

QUE02010

QUE02020

QUE02030

QUE02040

QUE02050

QUE02060

QUE02070

QUE02080

QUE02090

QUE02100

QUE02110

QUE02120

QUE02130

QUE02140

QUE02150

QUE02160

QUE02170

QUE02180

QUE02190

QUE02200

QUE02210

QUE02220

QUE02230

QUE02240

QUE02250

QUE02260

QUE02270

QUE02280

QUE02290

QUE02300

QUE02310

```

56 CONTINUE
57 CONTINUE
  IF (IVAR .LE. 0) GO TO 72
  READ (5,8) (KVAR(J),J=1,MV)
  IF (KVAR(1) .NE. 9) GO TO 60
  DO 61 J = 1,MV
    61 KVAR(J) = 1
60 CONTINUE
  IF (KVAR(1) .NE. 8) GO TO 678
  DO 73 J = 1,MV
    73 KVAR(J) = 2
678 WRITE (6,52) (KVAR(J),J=1,MV)
52 FORMAT (1H,22HVARIANZANALYSEKARTE(N),8X,100I1)
  IF (IGR .LE. 10) GO TO 650
  DO 651 J = 1,MV
    651 CONTINUE
    IF (KVAR(J) .EQ. 2) KVAR(J) = 1
650 CONTINUE
  IF (IGR .GT. 1) GO TO 72
  DO 676 J = 1,MV
    676 KVAR(J) = 0
  IVAR = 0
72 CONTINUE
  IF (IVER .LE. 0) GO TO 675
  READ (5,8) (IES(J),J=1,MV)
  DO 673 J = 1,MV
    673 KES(J) = IES(J)
  IF (IES(1) .NE. 9) GO TO 677
  DO 674 J = 1,MV
    674 IES(J) = 1
    KES(J) = 1
  IF (IES(J) = 1)
677 WRITE (6,74) (IES(J),J=1,MV)
74 FORMAT (1H,19HVERTEILUNGSKARTE(N),11X,100I1)
675 DO 680 J = 1,MV
    680 CONTINUE
    IF (KVAR(J) .EQ. 2) IES(J) = 1
    8 FORMAT (80I1)
  IF (IQ .EQ. 1) GO TO 105
  DO 100 J = 1,MV
    SMALL(J) = 10000.

```

```

QUE02320
QUE02330
QUE02340
QUE02350
QUE02360
QUE02370
QUE02380
QUE02390
QUE02400
QUE02410
QUE02420
QUE02430
QUE02440
QUE02450
QUE02460
QUE02470
QUE02480
QUE02490
QUE02500
QUE02510
QUE02520
QUE02530
QUE02540
QUE02550
QUE02560
QUE02570
QUE02580
QUE02590
QUE02600
QUE02610
QUE02620
QUE02630
QUE02640
QUE02650
QUE02660
QUE02670
QUE02680
QUE02690
QUE02700
QUE02710

```

```

100 BIG(J) = -10000.
    REWIND 8
    IF (KDAT .EQ. 1) GO TO 710
    CALL WAND (FMA,IT,IFO,NP,MV)
    GO TO 720
710 REWIND 9
    DO 715 I = 1,NP
    READ (IT,XM) (X(J),J=1,MV)
    WRITE (9,FMA) (X(J),J=1,MV)
    WRITE (9,FMA) (X(J),J=1,MV)
715 CONTINUE
    REWIND 9
720 CONTINUE
    DO 110 I = 1,NP
    READ (9,FM) (X(J),J=1,MV)
    READ (9,FMA) (Y(J),J=1,MV)
    IF (MISD .EQ. 3) GO TO 120
    DO 1110 J = 1,MV
    IF (X(J) .NE. 0.) GO TO 1110
    IF (Y(J) .NE. BLEER) GO TO 1111
    X(J) = FBLANK
    GO TO 1110
1111 IF (Y(J) .EQ. CLEER .OR. Y(J) .EQ. DLEER) X(J) = FBLANK
    IF (X(J) .EQ. 0. .AND. SIGN(1.,X(J)) .LT. 0.) X(J) = FBLANK
1110 CONTINUE
    IF (MISD .NE. 2) GO TO 117
    DO 116 J = 1,MV
    IF (AMIS(J) .LE. FBLANK) GO TO 116
    IF (X(J) .LE. FBLANK) GO TO 116
1116 IF (X(J) .EQ. AMIS(J)) X(J) = FBLANK
116 CONTINUE
117 CONTINUE
    IF (MISD .NE. 1) GO TO 120
    DO 119 J = 1,MV
    IF (X(J) .EQ. 0.) X(J) = FBLANK
119 CONTINUE
120 CONTINUE
    DO 115 J = 1,MV
    IF (X(J) .LE. FBLANK) GO TO 112
    IF (X(J) .LT. SMALL(J)) SMALL(J) = X(J)
114 IF (X(J)

```

```

QUE02720
QUE02730
QUE02740
QUE02750
QUE02760
QUE02770
QUE02780
QUE02790
QUE02800
QUE02810
QUE02820
QUE02830
QUE02840
QUE02850
QUE02860
QUE02870
QUE02880
QUE02890
QUE02900
QUE02910
QUE02920
QUE02930
QUE02940
QUE02950
QUE02960
QUE02970
QUE02980
QUE02990
QUE03000
QUE03010
QUE03020
QUE03030
QUE03040
QUE03050
QUE03060
QUE03070
QUE03080
QUE03090
QUE03100
QUE03110

```

```

IF (X<J) .GT. BIG(J)) BIG(J) = X(J)
GO TO 115
112 IV(J) = IV(J) + 1
115 CONTINUE
WRITE (8) (X(J),J=1,MV)
110 CONTINUE

REWIND 9
WRITE (6,10)
10 FORMAT (1H1,8HVARIABLE,5X,11HMINIM. WERT,5X,11HMAXIM. WERT,
A5X,12HMISSING DATA)
IF (MISD .EQ. 2) WRITE (6,40)
40 FORMAT (1H+,60X,14HMISS.DATA=CODE)
WRITE (6,41)
41 FORMAT (1H ,74(1H-)/)
DO 118 J = 1,MV
WRITE (6,9) J,SMALL(J),BIG(J),IV(J),<TEXT<(J,L),L=1,57)
IF (MISD .EQ. 2) WRITE (6,42) AMIS(J)
42 FORMAT (1H+,60X,F8.0)
118 CONTINUE

9 FORMAT (1H ,15,9X,F8.2,7X,F8.2,8X,I5,25X,57A1)
WRITE (6,36)
36 FORMAT (5(/),1H0,30HERLAEUTERUNGEN ZU DEN TABELLEN/1H ,30(1H-)/
A1H0,23HZEILE A = HAEUFIGKEITEN/
B1H ,52HZEILE B = PROZENTWERTE BEZOGEN AUF ZEILENHAEUFIGKEIT/
C1H ,53HZEILE C = PROZENTWERTE BEZOGEN AUF SPALTENHAEUFIGKEIT/
D1H ,52HZEILE D = PROZENTWERTE BEZOGEN AUF GESAMTHAEUFIGKEIT/
E1H ,50HZEILE E = ERWARTETE HAEUFIGKEITEN FUER CHI-QUADRAT//
DO 141 JJ = 1,MV
141 GVAR(JJ) = 1.-SMALL(JJ)
105 CONTINUE
REWIND 8
IF (KLASS .NE. 1) GO TO 146
DO 145 JJ = 1,MV
145 KS(JJ) = (BIG(JJ) - SMALL(JJ) + 1.)/FLOAT(KL(JJ)) + 0.999
146 CONTINUE
DO 131 K = 1,NDIM
131 IH(K) = 0
IDIM = IGR

```

```

QUE03120
QUE03130
QUE03140
QUE03150
QUE03160
QUE03170
QUE03180
QUE03190
QUE03200
QUE03210
QUE03220
QUE03230
QUE03240
QUE03250
QUE03260
QUE03270
QUE03280
QUE03290
QUE03300
QUE03310
QUE03320
QUE03330
QUE03340
QUE03350
QUE03360
QUE03370
QUE03380
QUE03390
QUE03400
QUE03410
QUE03420
QUE03430
QUE03440
QUE03450
QUE03460
QUE03470
QUE03480
QUE03490
QUE03500
QUE03510

```

C

```

JDIM = NDIM/(IDIM*MDIM)
J = MIN0(200,JDIM)
WRITE (6,37) IDIM,MDIM,J
37 FORMAT (I10,3HBEI,I3,12H GRUPPEN UND,I5,55H VARIABLEN DARF DIE SPA
ANNWEITE DER VARIABLEN HOECHSTENS,I5,6H SEIN./)
LDIM = IDIM*JDIM
DO 130 JJ = 1,MV
DO 130 K = 1,IGR
SU(K,JJ) = 0.
SU2(K,JJ) = 0.
X3(K,JJ) = 0.
X4(K,JJ) = 0.
130 CONTINUE
NPB = 0
DO 153 J = 1,IGR
LT(J) = 0
153 DO 151 I = 1,NP
READ (8) (X(J),J=1,MV)
DO 148 J = 1,IGR
IF (ITREN .LE. 0) GO TO 161
DO 147 K = 1,ITREN
MM = NO(K)
IF (X(MM) .LE. FBLANK) GO TO 151
IF (X(MM) .GE. GU(J,K) .AND. X(MM) .LE. GO(J,K)) GO TO 147
IF (G3(J,K) .LE. FBLANK) GO TO 148
IF (X(MM) .EQ. G3(J,K)) GO TO 147
IF (G4(J,K) .LE. FBLANK) GO TO 148
IF (X(MM) .EQ. G4(J,K)) GO TO 147
GO TO 148
147 CONTINUE
161 LT(J) = LT(J) + 1
NPB = NPB + 1
DO 150 JJ = 1,MV
JJI = (JJ-1)*LDIM
KR = BIG(JJ) - SMALL(JJ) + 1.
IF (KLASS .EQ. 1) KR = KS(JJ)
IF (KR .GT. JDIM) GO TO 150
IF (KR .GT. JDMAX) GO TO 150
IF (X(JJ) .LE. FBLANK) GO TO 150
N = J

```

```

QUE03520
QUE03530
QUE03540
QUE03550
QUE03560
QUE03570
QUE03580
QUE03590
QUE03600
QUE03610
QUE03620
QUE03630
QUE03640
QUE03650
QUE03660
QUE03670
QUE03680
QUE03690
QUE03700
QUE03710
QUE03720
QUE03730
QUE03740
QUE03750
QUE03760
QUE03770
QUE03780
QUE03790
QUE03800
QUE03810
QUE03820
QUE03830
QUE03840
QUE03850
QUE03860
QUE03870
QUE03880
QUE03890
QUE03900
QUE03910

```

```

SU(N, JJ) = SU(N, JJ) + X(JJ)
SU2(N, JJ) = SU2(N, JJ) + X(JJ)**2
X3(N, JJ) = X3(N, JJ) + X(JJ)**3
X4(N, JJ) = X4(N, JJ) + X(JJ)**4
IF (KLASS .EQ. 0) GO TO 140
IF (KL(JJ) .EQ. 1) GO TO 140
KK = K5(JJ)
ZKL = KL(JJ)
ZU(1) = SMALL(JJ)
IF (KK .LT. 2) GO TO 143
DO 144 LL = 2, KK
ZU(LL) = ZU(LL-1) + ZKL
144 CONTINUE
DO 142 LL = 1, KK
Z0 = ZU(LL) + ZKL - 0.001
IF (X(JJ) .LT. ZU(LL)) .OR. X(JJ) .GT. Z0) GO TO 142
IR = N + (LL-1) * IDIM + JJI
IH(IR) = IH(IR) + 1
GO TO 150
142 CONTINUE
GO TO 150
140 M = X(JJ) + GVAR(JJ)
IR = N + (M-1) * IDIM + JJI
IH(IR) = IH(IR) + 1
150 CONTINUE
GO TO 151
148 CONTINUE
151 WRITE (6, 71)
71 FORMAT (//)
DO 162 J = 1, IGR
162 WRITE (6, 70) J, LT(J)
70 FORMAT (1H, 6HGRUPPE, I3, 2X, 17HZAHL DER FAELLE =, I5)
23 FORMAT (1H1)
IF (NPB .GT. 0) GO TO 341
WRITE (6, 47)
47 FORMAT (1H0, 122HMIT DEN GRUPPENBESTIMMENDEN VARIABLEN UND DEREN COQUE04290
ADES LASSEN SICH KEINE GRUPPEN MIT BESETZUNGSZAHLEN GROESSER NULL FQUE04300
BINDEN./1H , 22HLAUF WIRD ABGEBROCHEN.) QUE04310

```

QUE03920  
 QUE03930  
 QUE03940  
 QUE03950  
 QUE03960  
 QUE03970  
 QUE03980  
 QUE03990  
 QUE04000  
 QUE04010  
 QUE04020  
 QUE04030  
 QUE04040  
 QUE04050  
 QUE04060  
 QUE04070  
 QUE04080  
 QUE04090  
 QUE04100  
 QUE04110  
 QUE04120  
 QUE04130  
 QUE04140  
 QUE04150  
 QUE04160  
 QUE04170  
 QUE04180  
 QUE04190  
 QUE04200  
 QUE04210  
 QUE04220  
 QUE04230  
 QUE04240  
 QUE04250  
 QUE04260  
 QUE04270  
 QUE04280

COQUE04290  
 FQUE04300  
 QUE04310

```

IF (IT .EQ. 5) GO TO 500
IF (IREW .EQ. 1) REWIND IT
GO TO 500
341 CONTINUE
C
DO 400 JJ = 1,MV
KB = 0
IF (SMALL(JJ) .NE. 10000.) GO TO 98
WRITE (6,46) JJ,(TEXT(JJ,L),L=1,80)
46 FORMAT (/IHO,11HIN VARIABLE,I3,22H SIND NUR MISSING DATA,4X,80A1)
GO TO 400
98 CONTINUE
KGR = IGR
JJ1 = (JJ-1)*LDIM
JGR = BIG(JJ) - SMALL(JJ) + 1.
IF (JGR.LE. 1) KVAR(JJ) = 0
IF (JGR.LE. 1) IES(JJ) = 0
IF (JGR.GT. 15 .AND. KL(JJ) .EQ. 1) KB = 1
IF (KLASS .EQ. 1) JGR = KS(JJ)
IF (JGR .LE. JDIM) GO TO 179
173 WRITE (6,39) JJ,(TEXT(JJ,L),L=1,80)
39 FORMAT (IHO,23HSPANNWEITE BEI VARIABLE,I5,9H ZU GROSS,3X,80A1)
GO TO 400
179 IF (JGR .LE. JDMAX) GO TO 174
GO TO 173
174 CONTINUE
IF (JGR .LE. 1) KL(JJ) = 1
IF (KLASS .NE. 1) GO TO 171
IF (KL(JJ) .EQ. 1) GO TO 171
ZKL = KL(JJ)
JGR = KS(JJ)
171 CONTINUE
B = 0.
NGES = 0
MINFE = 0
SUM = 0.
S2 = 0.
EXZ = 0.
SCH = 0.
DO 125 K = 3,17

```

```

QUE04320
QUE04330
QUE04340
QUE04350
QUE04360
QUE04370
QUE04380
QUE04390
QUE04400
QUE04410
QUE04420
QUE04430
QUE04440
QUE04450
QUE04460
QUE04470
QUE04480
QUE04490
QUE04500
QUE04510
QUE04520
QUE04530
QUE04540
QUE04550
QUE04560
QUE04570
QUE04580
QUE04590
QUE04600
QUE04610
QUE04620
QUE04630
QUE04640
QUE04650
QUE04660
QUE04670
QUE04680
QUE04690
QUE04700
QUE04710

```

```

125 FM2(K) = ALEER
    FM1(K) = ALEER
    DO 175 J = 1, IGR
    SUM = SUM + SU(J, JJ)
    S2 = S2 + SU2(J, JJ)
    EXZ = EXZ + X4(J, JJ)
    SCH = SCH + X3(J, JJ)
175 NN(J) = 0
    ICH = 1
    CHI = 0.
    DO 178 K = 1, JGR
    K1 = (K-1) * IDIM
    ISP(K) = 0
    DO 178 J = 1, IGR
    IR = J + K1 + JJI
178 ISP(K) = ISP(K) + IH(IR)
    KSP = JGR
    DO 272 K = 1, JGR
    IF (ISP(K) .EQ. 0) KSP = KSP - 1
272 CONTINUE
    IF (KSP .LE. 1) KVAR(JJ) = 0
    IF (KSP .LE. 1) IES(JJ) = 0
    IF (IGR .GT. 10 .OR. JGR .GT. 10) ICH = 0
    IF (JGR .LT. 2) ICH = 0
    IF (IGR .LE. 1) ICH = 0
    IF (ICH .EQ. 0) GOTO 177
    DO 176 J = 1, IGR
    DO 176 K = 1, JGR
176 FE(J, K) = 0.
177 CONTINUE
    DO 180 J = 1, IGR
    DO 181 K = 1, JGR
    IR = J + (K-1)*IDIM + JJI
181 NN(J) = NN(J) + IH(IR)
    IF (NN(J) .EQ. 0) GO TO 185
    SN = NN(J)
    AM(J) = SU(J, JJ)/SN
    IF (SN .EQ. 1.) GO TO 186
    S(J) = SQRT((SN*SU2(J, JJ)-SU(J, JJ)**2)/(SN*(SN-1.)))
186 NGES = NGES + NN(J)

```

```

QUE04720
QUE04730
QUE04740
QUE04750
QUE04760
QUE04770
QUE04780
QUE04790
QUE04800
QUE04810
QUE04820
QUE04830
QUE04840
QUE04850
QUE04860
QUE04870
QUE04880
QUE04890
QUE04900
QUE04910
QUE04920
QUE04930
QUE04940
QUE04950
QUE04960
QUE04970
QUE04980
QUE04990
QUE05000
QUE05010
QUE05020
QUE05030
QUE05040
QUE05050
QUE05060
QUE05070
QUE05080
QUE05090
QUE05100
QUE05110

```

```

IF (SN .EQ. 1.) S(J) = 0.
B = B + (SU(J,JJ)**2)/SN
GO TO 180
185 AM(J) = 0.
S(J) = 0.
KGR = KGR - 1
180 CONTINUE
IF (KGR .LE. 1) KVAR(JJ) = 0
IF (KGR .EQ. 2 .AND. KVAR(JJ) .EQ. 2) KVAR(JJ) = 1
IF (NGES .GT. 0) GO TO 187
WRITE (6,48) JJ, (TEXT(JJ,L),L=1,80)
48 FORMAT (/1H0,8HVARIABLE,15,3X,80A1/1H ,54HBEI VORLIEGENDER GRUPPEN
ABILDUNG GESAMTHAEUFIGKEIT NULL)
GO TO 400
187 CONTINUE
AGES = NGES
IF (ICH .EQ. 0) GO TO 183
DO 182 J = 1,IGR
DO 182 K = 1,JGR
FE(J,K) = FLOAT(NN(J)*ISP(K))/AGES
IR = J + (K-1)*IDIM + JJI
AH = IH(IR)
IF (FE(J,K) .EQ. 0.) GO TO 182
CHI = CHI + (AH-FE(J,K))*2/FE(J,K)
IF (FE(J,K) .LT. 5.) MINFE = MINFE + 1
182 CONTINUE
183 CONTINUE
IF (IVAR .LE. 0) GO TO 195
IF (KVAR(JJ) .LE. 0) GO TO 195
IF (KGR .LE. 1) ICH = 0
IF (KGR .LE. 1) GO TO 195
DO 184 J = 1,ITREN
IF (JJ .EQ. NO(J)) GO TO 195
184 CONTINUE
IF (NGES .EQ. 0) GO TO 199
IF (KGR .GE. NGES) GO TO 195
A = S2
C = (SUM*SUM)/AGES
AGR = KGR
QIN = (A-B)/(AGES-AGR)

```

QUE05120  
 QUE05130  
 QUE05140  
 QUE05150  
 QUE05160  
 QUE05170  
 QUE05180  
 QUE05190  
 QUE05200  
 QUE05210  
 QUE05220  
 QUE05230  
 QUE05240  
 QUE05250  
 QUE05260  
 QUE05270  
 QUE05280  
 QUE05290  
 QUE05300  
 QUE05310  
 QUE05320  
 QUE05330  
 QUE05340  
 QUE05350  
 QUE05360  
 QUE05370  
 QUE05380  
 QUE05390  
 QUE05400  
 QUE05410  
 QUE05420  
 QUE05430  
 QUE05440  
 QUE05450  
 QUE05460  
 QUE05470  
 QUE05480  
 QUE05490  
 QUE05500  
 QUE05510

```

IF (QIN .LE. 0.) GO TO 195
QZW = (B-C)/(AGR-1.)
F = QZW/QIN
OMEGA = (B-C-(AGR-1.)*QIN)/(A-C+QIN)
IDF1 = KGR-1
IDF2 = NGES-KGR
DF1 = IDF1
DF2 = IDF2
ALFA = PRBF(DF1,DF2,F) * 100.
CH = BART(IGR,AGES,KRED)
IF (CH .LT. 0.) CH = -CH
IF (KRED .LE. 1) GO TO 195
WACH = PRBF(DF1,1000.,CH/DF1) * 100.
CHI50 = .98*DF1 - .59 + DF1/100.
IF (IDF1 .EQ. 1) CHI50 = .455
IF (CH .GE. DF1 .OR. CH .LE. CHI50) GO TO 196
WACH = 100. - WACH
196 CONTINUE
IF (WACH .GT. 5.) IERG = JAA
IF (WACH .LE. 5.) IERG = NEIN
195 CONTINUE
IF (NGES .EQ. 0) GO TO 199
XQU = SUM/AGES
IF (AGES .EQ. 1.) GO TO 198
STA = SQRT((AGES*S2-SUM*SUM)/(AGES*(AGES-1.)))
198 IF (AGES .EQ. 1.) STA = 0.
GO TO 200
199 STA = 0.
XQU = 0.
200 CONTINUE
JJGR = JGR
IF (KB .EQ. 0) GO TO 267
ICH = 0
KU(1) = SMALL(JJ)
MG(1) = ISP(1)
JS = 1
IZ = SMALL(JJ) + 1.
DO 268 K = 2,JGR
IF (ISP(K) .LE. 0) GO TO 268
JS = JS + 1

```

```

QUE05520
QUE05530
QUE05530
QUE05540
QUE05550
QUE05550
QUE05560
QUE05570
QUE05580
QUE05590
QUE05600
QUE05610
QUE05620
QUE05630
QUE05640
QUE05650
QUE05650
QUE05660
QUE05670
QUE05680
QUE05690
QUE05700
QUE05710
QUE05720
QUE05730
QUE05740
QUE05750
QUE05750
QUE05770
QUE05780
QUE05790
QUE05800
QUE05810
QUE05820
QUE05830
QUE05840
QUE05850
QUE05860
QUE05870
QUE05880
QUE05890
QUE05900
QUE05910

```

```

KU(JS) = IZ
MG(JS) = ISP(K)
268 IZ = IZ + 1
      JGR = JS
      LGR = JGR
267 CONTINUE
      IA = 1
      IAA = 1
      IE = 15
      IE = MIN0(IE,JGR)
      IP = IE - IA + 1
      DO 255 K = 1,IP
      FM2(K+2) = STR
255 FM1(K+2) = FZ
      JDF2 = KGR*KSP/5+1
      IP = 0
285 WRITE (6,11) JJ,(TEXT(JJ,L),L=1,80)
      IP = IP+1
11  FORMAT (3(/),9H VARIABLE,I5,3X,80A1/1H0,26X,17HMITTEL- STANDARD-,
      A20X,13SHAUEFIGKEITEN/1H ,6HGRUPPE,12X,24HANZAHL WERT ABWEICH.)
      IF (KLASS .NE. 1) GO TO 270
      IF (KL(JJ) .EQ. 1) GO TO 270
      IF (IP .GT. 1) GO TO 264
      ZU(1) = (2.*SMALL(JJ)+ZKL-1.)/2.
      DO 265 K = 2,JGR
265 ZU(K) = ZU(K-1) + ZKL
264 WRITE (6,12) (ZU(K),K=IA,IE)
12  FORMAT (1H+,42X,15F6.1)
      WRITE (6,29)
29  FORMAT (1H ,42(1H-))
      WRITE (6,FM1)
      WRITE (6,30)
30  FORMAT (1H )
      GO TO 275
270 IF (IP .GT. 1) GO TO 266
      IF (KB .EQ. 1) GO TO 266
      KU(1) = SMALL(JJ)
      DO 271 K = 2,JGR
271 KU(K) = KU(K-1) + 1
266 WRITE (6,13) (KU(K),K=IA,IE)

```

```

QUE05920
QUE05930
QUE05940
QUE05950
QUE05960
QUE05970
QUE05980
QUE05990
QUE06000
QUE06010
QUE06020
QUE06030
QUE06040
QUE06050
QUE06060
QUE06070
QUE06080
QUE06090
QUE06100
QUE06110
QUE06120
QUE06130
QUE06140
QUE06150
QUE06160
QUE06170
QUE06180
QUE06190
QUE06200
QUE06210
QUE06220
QUE06230
QUE06240
QUE06250
QUE06260
QUE06270
QUE06280
QUE06290
QUE06300
QUE06310

```

```

13  FORMAT (1H+,42X,15I6)
    WRITE (6,29)
    WRITE (6,FM1)
    WRITE (6,30)
275  CONTINUE
    IF (IGR .LE. 1) GO TO 291
    IF (NGES .EQ. 0) GO TO 291
    IF (ICH .LE. 0) GO TO 432
    DO 430 J = 1,IGR
430  MAXZ(J) = 0
    DO 431 K = 1,JGR
431  MAXS(K) = 0
    MAXZS = 0
    MAXSS = 0
432  CONTINUE
C
    IF (KB .EQ. 1) JGR = JJGR
    DO 290 J = 1,IGR
    IF (ICH .LE. 0) GO TO 435
    IF (NN(J) .GT. MAXSS) MAXSS = NN(J)
435  CONTINUE
    K1 = J
    IF (NN(J) .EQ. 0 .AND. IGR .GT. 10) GO TO 290
    IS = 0
    IF (KB .EQ. 0) GO TO 420
    JS = 0
    BN = NN(J)
    DO 405 K = IAA,JGR
    IF (K .GT. 1) GO TO 415
    IF (ISP(1) .GT. 0) GO TO 415
    MA(1) = 0
    AB(1) = 0.
    AC(1) = 0.
    AD(1) = 0.
    JS = 1
    GO TO 404
415  CONTINUE
    IF (ISP(K) .LE. 0) GO TO 404
    JS = JS + 1
    IR = J + (K-1) * IDIM + JJI
    QUE06320
    QUE06330
    QUE06340
    QUE06350
    QUE06360
    QUE06370
    QUE06380
    QUE06382
    QUE06383
    QUE06384
    QUE06385
    QUE06386
    QUE06387
    QUE06388
    QUE06389
    QUE06390
    QUE06400
    QUE06410
    QUE06412
    QUE06413
    QUE06414
    QUE06420
    QUE06430
    QUE06440
    QUE06450
    QUE06460
    QUE06470
    QUE06480
    QUE06490
    QUE06500
    QUE06510
    QUE06520
    QUE06530
    QUE06540
    QUE06550
    QUE06560
    QUE06570
    QUE06580
    QUE06590
    QUE06600

```

```

MA(J5) = IH(IR)
BK = ISP(K)
BH = IH(IR)
IF (IREL .NE. 1) GO TO 407
IF (NN(J) .EQ. 0) GO TO 406
AB(JS) = BH*100./3N
GO TO 407
406 AB(JS) = 0
407 IF (JREL .NE. 1) GO TO 410
AC(JS) = BH*100./BK
410 IF (KREL .NE. 1) GO TO 404
AD(JS) = BH*100./AGES
404 KK = K
IF (JS .EQ. 15) GO TO 420
405 CONTINUE
420 CONTINUE
IF (KB .EQ. 1) GO TO 293
DO 292 K = IA,IE
IS = IS + 1
IR = J + (K-1)*IDIM + JJI
JH(IS) = IH(IR)
IF (ICH .LE. 0) GO TO 292
IF (JH(IS) .GT. MAXS(IS)) MAXS(IS) = JH(IS)
IF (JH(IS) .GT. MAXZ(J)) MAXZ(J) = JH(IS)
292 CONTINUE
WRITE (6,14) K1,NN(J),AM(J),S(J),(JH(K),K=1,IS)
14 FORMAT (1H0,I5,10X,4H A ,I5,1X,F7.2,1X,F8.3,1X,I5I6)
GO TO 295
293 WRITE (6,14) K1,NN(J),AM(J),S(J),(MA(K),K=1,J5)
295 WRITE (6,24) TX(J,1),TX(J,2)
24 FORMAT (1H ,A8,A7)
DO 296 K = 1,ITREN
IF (JJ .EQ. NO(K)) GO TO 290
296 CONTINUE
IF (IREL .NE. 1) GO TO 310
IF (KB .EQ. 1) GO TO 277
IS = 0
DO 280 K = IA,IE
IS = IS + 1
IF (NN(J) .EQ. 0) GO TO 279

```

```

QUE06610
QUE06620
QUE06630
QUE06640
QUE06650
QUE06660
QUE06670
QUE06680
QUE06690
QUE06700
QUE06710
QUE06720
QUE06730
QUE06740
QUE06750
QUE06760
QUE06770
QUE06780
QUE06790
QUE06800
QUE06810
QUE06812
QUE06813
QUE06814
QUE06815
QUE06820
QUE06830
QUE06840
QUE06850
QUE06860
QUE06870
QUE06880
QUE06890
QUE06900
QUE06910
QUE06920
QUE06930
QUE06940
QUE06950
QUE06960

```

```

IR = J + (K-1)*IDIM + JJI
AH = IH(IR)
H(1S) = (AH*100.)/FLOAT(NN(J))
GO TO 280
279 H(1S) = 0.
280 CONTINUE
15 FORMAT (1H+,15X,4H B ,5H100.0,18X,15F6.1)
WRITE (6,15) (H(K),K=1,1S)
GO TO 310
277 WRITE (6,15) (AB(K),K=1,JS)
310 IF (JREL.NE. 1) GO TO 320
IF (KB .EQ. 1) GO TO 306
IS = 0
DO 305 K = IA,IE
IS = IS+1
IF (ISP(K) .EQ. 0) GO TO 304
IR = J + (K-1)*IDIM + JJI
AH = IH(IR)
H(1S) = (AH*100.)/FLOAT(ISP(K))
GO TO 305
304 H(1S) = 0.
305 CONTINUE
306 HH = FLOAT(NN(J)*100)/AGES
IF (IREL .EQ. 1) WRITE (6,30)
IF (KB .EQ. 0) WRITE (6,22) HH,(H(K),K=1,1S)
IF (KB .EQ. 1) WRITE (6,22) HH,(AC(K),K=1,JS)
22 FORMAT (1H+,15X,4H C ,F5.1,18X,15F6.1)
320 IF (KREL.NE. 1) GO TO 330
IF (NGES .EQ. 0) GO TO 330
IF (KB .EQ. 1) GO TO 314
IS = 0
DO 315 K = IA ,IE
IS = IS + 1
IR = J + (K-1)*IDIM + JJI
AH = IH(IR)
315 H(1S) = (AH*100.)/AGES
314 HH = FLOAT(NN(J)*100)/AGES
IF (IREL .EQ. 1) WRITE (6,30)
IF (KB .EQ. 0) WRITE (6,35) HH,(H(K),K=1,1S)
IF (KB .EQ. 1) WRITE (6,35) HH,(AD(K),K=1,JS)

```

```

QUE06970
QUE06980
QUE06990
QUE07000
QUE07010
QUE07020
QUE07030
QUE07040
QUE07050
QUE07060
QUE07070
QUE07080
QUE07090
QUE07100
QUE07110
QUE07120
QUE07130
QUE07140
QUE07150
QUE07160
QUE07170
QUE07180
QUE07190
QUE07200
QUE07210
QUE07220
QUE07230
QUE07240
QUE07250
QUE07260
QUE07270
QUE07280
QUE07290
QUE07300
QUE07310
QUE07320
QUE07330
QUE07340
QUE07350
QUE07360

```

```

35 FORMAT (1H+,15X,4H D ,F5.1,18X,15F6.1)
330 IF (ICH .EQ. 0) GO TO 290
IF (MINFE .GT. JDF2) GO TO 290
IS = 0
DO 325 K = IA,IE
IS = IS+1
325 H(IS) = FE(J,K)
IF (IREL.EQ.1.OR.JREL.EQ.1.OR.KREL.EQ.1) WRITE (6,30)
WRITE (6,25) (H(K),K=1,IS)
25 FORMAT (1H+,15X,4H E ,23X,15F6.1)
290 CONTINUE
IF (ICH .LE. 0) GO TO 449
DO 445 K = 1,JGR
IF (ISP(K) .GT. MAXZS) MAXZS = ISP(K)
445 CONTINUE
449 CONTINUE
WRITE (6,26)
26 FORMAT (1H ,42(1H-))
WRITE (6,FM1)
291 CONTINUE
IF (KB .EQ. 1) GO TO 298
WRITE (6,16) NGES,XQU,STA,(ISP(K),K=IA,IE)
GO TO 299
298 WRITE( 6,16) NGES,XQU,STA,(MG(K),K=IA,IE)
299 CONTINUE
16 FORMAT (1H0,19X,I5,1X,F7.2,1X,F8.3,1X,15I6)
IF (IREL .NE. 1) GO TO 287
IS = 0
DO 286 K = IA,IE
IS = IS+1
IF (NGES .EQ. 0) GO TO 289
IF (KB .EQ. 1) GO TO 288
H(IS) = FLOAT(ISP(K)*100)/AGES
GO TO 286
289 H(IS) = 0.
GO TO 286
288 H(IS) = FLOAT(MG(K)*100)/AGES
286 CONTINUE
IF (KB .EQ. 1) JGR = LGR
WRITE (6,30)

```

```

QUE07370
QUE07380
QUE07390
QUE07400
QUE07410
QUE07420
QUE07430
QUE07440
QUE07450
QUE07460
QUE07470
QUE07472
QUE07473
QUE07474
QUE07475
QUE07476
QUE07480
QUE07490
QUE07500
QUE07510
QUE07520
QUE07530
QUE07540
QUE07550
QUE07560
QUE07570
QUE07580
QUE07590
QUE07600
QUE07610
QUE07620
QUE07630
QUE07640
QUE07650
QUE07660
QUE07670
QUE07680
QUE07690
QUE07700
QUE07710

```

```

WRITE (6,15) (H(K),K=1,IS)
287 WRITE (6,33)
33 FORMAT (1H ,42(1H=))
IF (KB .EQ. 1) JGR = LGR
WRITE (6,FM2)
IF (IE .EQ. JGR) GO TO 300
IA = IA + 15
IAA = KK + 1
IE = IE + 15
IE = MIN0(IE, JGR)
JA = IE-IA+1
IF (JA .EQ. 15) GO TO 285
JE = 15-JA
DO 294 K = 1,JE
FM1(18-K) = ALEER
294 FM2(18-K) = ALEER
GO TO 285
300 CONTINUE
IF (IGR .LE. 1) GO TO 381
DO 301 K = 1,ITREN
IF (JJ .EQ. NO(K)) GO TO 380
301 CONTINUE
IF (KVAR(JJ) .LE. 0) GO TO 370
IF (QIN .LE. 0.) KVAR(JJ) = 0
IF (KGR .LE. 1 .OR. QIN .LE. 0.) GO TO 370
KERG = JAA
IF (ALFA .LE. 5.) KERG = NEIN
WRITE (6,17) QZW,QIN,IDF1,IDF2,F,ALFA,KERG
17 FORMAT (1H ,33HVARIANZANALYSE. MAQ(ZWISCHEN) =,F10.2,5X,16HMAQ(I
ANNEHALB) =,F10.2,5X,13HDF(ZAEHLER) =,15,5X,12HDF(NENNER) =,15/
BIH ,3HF =,F10.3,5X,35HIRRTUMSWAHRSCHEINLICHKEIT (ALPHA) =,F10.2,
C8H PROZENT,31X,24HGLEICHHEIT MITTELWERTE =,1X,A4)
IF (ALFA .LE. 5.) GO TO 302
KVAR(JJ) = 1
IES(JJ) = KES(JJ)
302 CONTINUE
WRITE (6,1001) OMEGA
1001 FORMAT (1H ,46HPRAKTISCHE SIGNIFIKANZ. OMEGA-QUADRAT (HAYS) =,
AF7.4)
IF (IGR .NE. 2) GO TO 307

```

```

QUE07720
QUE07730
QUE07740
QUE07750
QUE07760
QUE07770
QUE07780
QUE07790
QUE07800
QUE07810
QUE07820
QUE07830
QUE07840
QUE07850
QUE07860
QUE07870
QUE07880
QUE07890
QUE07900
QUE07910
QUE07920
QUE07930
QUE07940
QUE07950
QUE07960
QUE07970
QUE07980
QUE07990
QUE08000
QUE08010
QUE08020
QUE08030
QUE08040
QUE08050
QUE08060
QUE08070
QUE08080
QUE08090
QUE08100
QUE08110

```

```

T = SQRT(ABS(F))
IF (F .LE. 0.) GO TO 307
IF (NGES .LE. 2) GO TO 307
NN1 = NN(1) - 1
NN2 = NN(2) - 1
NN3 = NN1*NN2
IF (NN3 .LE. 0) GO TO 307
NN1 = NN(1)
NN2 = NN(2)
CALL TTEST(NN1, NN2, F, Z, P, R2)
KM = NGES-2
WRITE (6,43) R2, T, KM, P, Z
43 FORMAT (1H, 24X, 22HR-QUADR. (PUNKTBIS.) =, F7.4, 6X, 4H(T =, F7.3,
A5X, 4HDF =, I5, 1H)/1H, 24X, 22HU (NACH LEVY, 1967) =, F5.1, 6H PROZ.,
B2X, 15HZ-WERT FUER U =, F6.2)
307 CONTINUE
IF (KRED .LE. 1) WRITE (6,28)
28 FORMAT (1H, 76HBARTLETT-TEST AUF GLEICHHEIT DER VARIANZEN KONNTE NI
ACHT DURCHGEFUERT WERDEN/)
IF (KRED .LE. 1) GO TO 370
KRED = KRED - 1
WRITE (6,27) CH, KRED, WACH, IERG
27 FORMAT (1H, 49HBARTLETT-TEST AUF GLEICHHEIT DER VARIANZEN. CHI =,
AF7.2, 3X, 4HDF =, I4, 3X, 15HIRRT.WAHRSCH. =, F6.2, 8H PROZENT, 5X,
B22HGLEICHHEIT VARIANZEN =, IX, A4/)
370 IF (ICH .EQ. 0) GO TO 381
AXS = MAXSS
AXZ = MAXZS
ALA = 0.
DO 480 J = 1, IGR
ALA = ALA + FLOAT(MAXZ(J))
BLA = 0.
DO 481 K = 1, JGR
BLA = BLA + FLOAT(MAXS(K))
481 IF (MAXZS .GE. NGES .OR. MAXSS .GE. NGES) GO TO 484
ALAM = (ALA-AXZ)/(AGES-AXZ)
BLAM = (BLA-AXS)/(AGES-AXS)
ABLAM = (ALA+BLA-AXZ-AXS)/(2.*AGES-AXZ-AXS)
GO TO 485
484 ALAM = 0.

```

QUE08120

QUE08130

QUE08140

QUE08150

QUE08160

QUE08170

QUE08180

QUE08190

QUE08200

QUE08210

QUE08220

QUE08230

QUE08240

QUE08250

QUE08260

QUE08270

QUE08280

QUE08290

QUE08300

QUE08310

QUE08320

QUE08330

QUE08340

QUE08350

QUE08360

QUE08370

QUE08372

QUE08373

QUE08374

QUE08375

QUE08376

QUE08377

QUE08378

QUE08379

QUE08380

QUE08381

QUE08382

QUE08383

QUE08384

QUE08385

```

BLAM = 0.
ABLAM = 0.
485 CONTINUE
IDF1 = (KGR-1)*(KSP-1)
IDF2 = KGR*KSP/5+1
DF1 = IDF1
IF (IDF1 .LE. 0) GO TO 381
DF2 = CHI/DF1
ALFA = PRBF(DF1,1000.0,DF2) * 100.
IF (MINFE .GT. IDF2) GO TO 379
CHI50 = .98*DF1 - .59 + DF1/100.
IF (IDF1 .EQ. 1) CHI50 = .455
IF (DF2 .GE. 1. .OR. CHI .LE. CHI50) GO TO 371
ALFA = 100. -ALFA
371 CONTINUE
KERG = JAA
IF (ALFA .LE. 5.) KERG = NEIN
WRITE (6,19) CHI, IDF1, ALFA, KERG
19 FORMAT (1H, 13HCHI-QUADRAT =, F10.3, 5X, 4HDF =, I3, 3X, 35HIRRTUMSWAHRSA,
ACHENLICHKEIT (ALPHA) =, F10.2, 8H PROZENT, 5X, 31HGLEICHE VERTEILUNG
BIN GRUPPEN =, I1, A4)
329 IF (MINFE .GT. 0) WRITE (6,18) MINFE
18 FORMAT (1H, 14, 38H ERWARTETE HAEUFIGKEITEN KLEINER ALS 5)
379 CONTINUE
CONT = SQRT((CHI+(CHI+AGES)))
CM = MIN0(KGR, KSP)
DM = CM - 1.
CM = SQRT((CM-1.)/CM)
CM = CONT/CM
PHI2 = SQRT((CHI/(AGES*DM)))
IF (MINFE .GT. IDF2) WRITE (6,31) CHI
31 FORMAT (1H, 13HCHI-QUADRAT =, F10.3, 5X, 25HSIGNIFIKANZPRUEFUNG NICHT,
A, 61HDURCHGEFUEHRT, DA MEHR ALS 20 PROZ. DER ERW.HAEUF. KLEINER 5.)
WRITE (6,38) CONT, CM, PHI2
38 FORMAT (1H, 25HKONTINGENZKOEFFIZIENT C =, F6.3, 6X, 36HKORRIGIERTER K,
AONTINGENZKOEFFIZIENT =, F6.3, 5X, 12HV (CRAMER) =, F6.3)
IF (ALAM .LT. 0.) ALAM = 0.
IF (BLAM .LT. 0.) BLAM = 0.
IF (ABLAM .LT. 0.) ABLAM = 0.
WRITE (6,66) ALAM, BLAM, ABLAM

```

```

QUE08386
QUE08387
QUE08388
QUE08389
QUE08390
QUE08400
QUE08410
QUE08420
QUE08430
QUE08440
QUE08450
QUE08460
QUE08470
QUE08480
QUE08490
QUE08500
QUE08510
QUE08520
QUE08530
QUE08540
QUE08550
QUE08560
QUE08570
QUE08575
QUE08580
QUE08590
QUE08591
QUE08600
QUE08610
QUE08611
QUE08615
QUE08616
QUE08617
QUE08620
QUE08630
QUE08640
QUE08645
QUE08646
QUE08647
QUE08650

```

```

66 FORMAT (1H ,44HINDEX OF PREDICTIVE ASSOCIATION. LAMBDA(A) =,F6.3,
A5X,11HLAMBDA(B) =,F6.3,5X,12HLAMBDA(AB) =,F6.3)
381 IF (IES(JJ) .LE. 0) GO TO 380
IF (IGR .LE. 1) GO TO 620
DO 610 J = 1,IGR
PB = NN(J)
CALL SCHEX(SU(J,JJ),SU2(J,JJ),X3(J,JJ),X4(J,JJ),AM(J),PB,SC(J),
AEX(J),SSS(J),SEE(J),TS(J),TEX(J),ALS(J),AEE(J),TESCH(J),TEEX(J))
610 CONTINUE
J = IGR + 1
620 CONTINUE
IF (IGR .LE. 1) J = 1
PB = NGES
CALL SCHEX(SUM,S2,SCH,EXZ,XQU,PB,SC(J),EX(J),SSS(J),SEE(J),TS(J),
ATEX(J),ALS(J),AEE(J),TESCH(J),TEEX(J))
IF (KVAR(JJ) .LT. 2) GO TO 399
IF (IGR .GT. 10) GO TO 399
MGR = IGR
CALL SCHEFF (QIN,MGR,AM,NGES,SCHEF)
CALL FKRT (NGES,MGR,FKR,IAL)
FKR = FLOAT(MGR-1)*FKR
399 CONTINUE
KVJ = KVAR(JJ)
KG = IGR-1
WRITE (6,81)
81 FORMAT (1H0,6HGRUPPE,11X,41HSCHIEFE/ ALPHA / I EXZESS/ ALPHA
A/ I)
IF (KVJ .EQ. 2) WRITE (6,82) FKR
82 FORMAT (1H+,58X,46H SCHEFFE-TEST PAARWEISE MITTELW.-VERGLEICHE ,
A16HKRIT Fx(IGR-1) =,F8.2)
WRITE (6,83)
83 FORMAT (1H ,16X,2(2X,19HT-WERT ERGEBNIS I))
IF (KVJ .EQ. 2) WRITE (6,84) (J,J=1,KG)
84 FORMAT (1H+,59X,9I8)
IF (IGR .GE. 2) WRITE (6,85)
85 FORMAT (1H ,36(1H-),1H+,20(1H-),1H+)
IF (KVJ .EQ. 2) WRITE (6,86)
86 FORMAT (1H+,58X,73(1H-))
IF (IGR .GE. 2) WRITE (6,87)

```

QUE08660

QUE08670

QUE08680

QUE08690

QUE08700

QUE08710

QUE08720

QUE08730

QUE08740

QUE08750

QUE08760

QUE08770

QUE08780

QUE08790

QUE08800

QUE08810

QUE08820

QUE08830

QUE08840

QUE08850

QUE08860

QUE08870

QUE08880

QUE08890

QUE08900

QUE08910

QUE08920

QUE08930

QUE08940

QUE08950

QUE08960

QUE08970

QUE08980

QUE08990

QUE09000

QUE09010

QUE09020

QUE09030

QUE09040

QUE09050

```

FE(1,1) = BLEER
FE(1,2) = BLEER
87 FORMAT (1H ,36X,1HI,20X,1HI)
   IF (IGR .LE. 1) GO TO 625
   DO 640 J = 1,IGR
   IF (NN(J) .LE. 0) GO TO 640
   WRITE (6,88) J,SC(J),ALS(J),EX(J),AEE(J)
88 FORMAT (1H ,14,13X,F7.2,3X,F7.3,2X,1HI,2X,F7.2,2X,F7.3,2X,1HI)
   J1 = J - 1
   IF (KVJ .NE. 2) GO TO 630
   IF (J .EQ. 1) GO TO 630
   DO 629 K = 1,J1
   FE(K,1) = BLEER
   FE(K,2) = BLEER
   IF (SCHEF(J,K) .LT. FKR) GO TO 629
   FE(K,1) = SIG1
   FE(K,2) = SIG2
629 CONTINUE
   WRITE (6,89) (SCHEF(J,K),K=1,J1)
89 FORMAT (1H+,59X,9F8.2)
630 WRITE (6,90) TX(J,1),TX(J,2),TS(J),TESCH(J),TEX(J),TEEX(J)
90 FORMAT (1H ,A8,A7,2X,F7.2,4X,A6,2X,1HI,2X,F7.2,3X,A6,2X,1HI)
   IF (KVJ .EQ. 2) WRITE (6,91) (FE(K,1),FE(K,2),K=1,J1)
91 FORMAT (1H+,59X,9(2X,A4,A2))
   WRITE (6,87)
640 CONTINUE
625 CONTINUE
   WRITE (6,85)
   IF (KVJ .EQ. 2) WRITE (6,86)
   J = IGR + 1
   IF (IGR .LE. 1) J = 1
   WRITE (6,93) SC(J),ALS(J),EX(J),AEE(J),TS(J),TESCH(J),TEX(J),
   ATEEX(J)
93 FORMAT (1H ,17X,F7.2,3X,F7.3,2X,1HI,2X,F7.2,2X,F7.3,2X,1HI/
   A1H ,17X,F7.2,4X,A6,2X,1HI,2X,F7.2,3X,A6,2X,1HI)
   WRITE (6,85)
380 CONTINUE
   IF (IGR .GE. 2) WRITE (6,20)
20 FORMAT (4(/))
   IF (IGR .LE. 1) WRITE (6,71)

```

```

QUE09060
QUE09070
QUE09080
QUE09090
QUE09100
QUE09110
QUE09120
QUE09130
QUE09140
QUE09150
QUE09160
QUE09170
QUE09180
QUE09190
QUE09200
QUE09210
QUE09220
QUE09230
QUE09240
QUE09250
QUE09260
QUE09270
QUE09280
QUE09290
QUE09300
QUE09310
QUE09320
QUE09330
QUE09340
QUE09350
QUE09360
QUE09370
QUE09380
QUE09390
QUE09400
QUE09410
QUE09420
QUE09430
QUE09440
QUE09441

```

```

400 CONTINUE
GO TO 500
600 STOP
END
FUNCTION PRBF(DA,DB,FR)
PRBF = 1.
IF (DA*DB*FR .EQ. 0.) RETURN
IF (FR .LT. 1.) GO TO 5
A = DA
B = DB
F = FR
GO TO 10
5 A = DB
B = DA
F = 1./FR
10 AA = 2./(9.*A)
BB = 2./(9.*B)
Z = ABS(((1.-BB)*F**0.333333-1.+AA) / SQRT(BB*F**0.666667+AA))
IF (B .LT. 4.) Z = Z * (1.+0.08*Z**4/B**3)
PRBF = 0.5/(1.+Z*(0.196854+Z*(0.115194+Z*(0.000344+Z*0.019527))))
A**4
IF (FR .LT. 1.) PRBF = 1.-PRBF
RETURN
END
FUNCTION BART(K,GES,NK)
COMMON S(20),NN(20)
AK = K
S2 = 0.
C = 0.
SU = 0.
DO 10 I = 1,K
V = S(I) * S(I)
IF (V .LE. 0.) .AND. NN(I) .LE. 0) GO TO 5
IF (V .LE. 0.) GO TO 30
VP = NN(I) - 1
IF (NN(I) .LE. 1) GO TO 5
S2 = S2 + V * VP
C = C + (1./VP)
SU = SU + (VP * ALOG10(V))
GO TO 10

```

QUE09450  
 QUE09460  
 QUE09470  
 QUE09480  
 QUE09490  
 QUE09500  
 QUE09510  
 QUE09520  
 QUE09530  
 QUE09540  
 QUE09550  
 QUE09560  
 QUE09570  
 QUE09580  
 QUE09590  
 QUE09600  
 QUE09610  
 QUE09620  
 QUE09630  
 QUE09640  
 QUE09650  
 QUE09660  
 QUE09670  
 QUE09680  
 QUE09690  
 QUE09700  
 QUE09710  
 QUE09720  
 QUE09730  
 QUE09740  
 QUE09750  
 QUE09760  
 QUE09770  
 QUE09780  
 QUE09790  
 QUE09800  
 QUE09810  
 QUE09820  
 QUE09830  
 QUE09840

```

5 AK = AK - 1
10 CONTINUE
DF = GES - AK
NK = AK
IF (NK .LE. 1) RETURN
S2 = S2/DF
C = 1.0 + (C-(1.0/DF))/(3.0*(AK-1.0))
BART = (2.3026*(DF*ALOG10(S2)-SU))/C
RETURN
30 NK = 0
RETURN
END
SUBROUTINE WAND (FM,IT,IFO,NP,K1)
DIMENSION FM(400),X(200)
DATA STR/IH//
KART = 1
DO 100 J = 1,IFO
IF (FM(J) .NE. STR) GO TO 100
KART = KART + 1
100 CONTINUE
DO 190 I = 1,NP
READ (IT,FM) (X(J),J=1,K1)
WRITE (9,FM) (X(J),J=1,K1)
WRITE (9,FM) (X(J),J=1,K1)
190 CONTINUE
REWIND 9
RETURN
END
SUBROUTINE AFORM (FM,FMA,IFO)
DIMENSION FM(400),FMA(400)
DATA PUNKT,FFF,AAA,BLANK/IH.,IHF,IHA,IH /
DO 20 J = 1,IFO
FMA(J) = FM(J)
IF (FM(J) .NE. FFF) GO TO 10
FMA(J) = AAA
GO TO 20
10 IF (FM(J) .EQ. PUNKT) FMA(J) = BLANK
IF (J .EQ. 1) GO TO 20
IF (FM(J-1) .EQ. PUNKT) FMA(J) = BLANK
20 CONTINUE

```

QUE09850  
 QUE09860  
 QUE09870  
 QUE09880  
 QUE09890  
 QUE09900  
 QUE09910  
 QUE09920  
 QUE09930  
 QUE09940  
 QUE09950  
 QUE09960  
 QUE09970  
 QUE09980  
 QUE09990  
 QUE10000  
 QUE10010  
 QUE10020  
 QUE10030  
 QUE10040  
 QUE10050  
 QUE10060  
 QUE10070  
 QUE10080  
 QUE10090  
 QUE10100  
 QUE10110  
 QUE10120  
 QUE10130  
 QUE10140  
 QUE10150  
 QUE10160  
 QUE10170  
 QUE10180  
 QUE10190  
 QUE10200  
 QUE10210  
 QUE10220  
 QUE10230  
 QUE10240

```

WRITE (6,3) (FMA(J),J=1,IFO)
3 FORMAT (IH0,59HAUS DEM F-FORMAT ERMITTELTE DAS PROGRAMM FOLGENDES
AA-FORMAT/(IH ,80A1))
RETURN
END
SUBROUTINE SCHEX (XX1,XX2,XX3,XX4,XQUER,PROB,SCHIEF,EXZESS,SS,SE,
ATS,TE,AS,AE,IXSC,TEXE)
COMMON/ALF/ALPHA
REAL*8 BL,BNO,IXSC,TEXE
DATA BL,BNO/IH ,6HNORMAL/
IXSC = BL
TEXE = BL
IF (PROB .LE. 1.) GO TO 50
VAR = (PROB**XX2-XX1**XX1)/(PROB*(PROB-1.))
IF (VAR .LE. 0.) GO TO 50
ABW = SQRT(VAR)
SCHIEF = (XX3 - 3.*XQUER**XX2 + 3.*XQUER**2**XX1 - PROB**XQUER**3)
A/(PROB*ABW**3)
EXZESS = ((XX4 - 4.*XQUER**XX3 + 6.*XQUER**2**XX2 - 4.*XQUER**3*
AXX1 + PROB**XQUER**4)/(PROB**VAR**2)) - 3.
SS = SQRT(6./PROB)
SE = 2.*SS
TS = SCHIEF/SS
TE = EXZESS/SE
DF = PROB-1.
AS = PRBF(1.,DF,TS**2)
AE = PRBF(1.,DF,TE**2)
IF (AS .GT. ALPHA) IXSC = BNO
IF (AE .GT. ALPHA) TEXE = BNO
RETURN
50 SCHIEF = 0.
EXZESS = 0.
SS = 0.
SE = 0.
TS = 0.
TE = 0.
AS = 0.
AE = 0.
RETURN
END

```

```

QUEI0250
QUEI0260
QUEI0270
QUEI0280
QUEI0290
QUEI0300
QUEI0310
QUEI0320
QUEI0330
QUEI0340
QUEI0350
QUEI0360
QUEI0370
QUEI0380
QUEI0390
QUEI0400
QUEI0410
QUEI0420
QUEI0430
QUEI0440
QUEI0450
QUEI0460
QUEI0470
QUEI0480
QUEI0490
QUEI0500
QUEI0510
QUEI0520
QUEI0530
QUEI0540
QUEI0550
QUEI0560
QUEI0570
QUEI0580
QUEI0590
QUEI0600
QUEI0610
QUEI0620
QUEI0630
QUEI0640

```

```

SUBROUTINE SCHEFF(QIN, IGR, AM, NG, SCHEF)
COMMON S(20), NN(20)
DIMENSION SCHEF(10,10), AM(20)
COMMON/ALF/ALPHA
DO 15 J = 1, IGR
DO 15 K = 1, IGR
M = 0
15 SCHEF(J,K) = 0.
DO 5 J = 1, IGR
IF (NN(J) .LE. 0) M = M + 1
5 CONTINUE
KG = IGR-1
DO 10 L = 1, KG
KA = L + 1
DO 10 K = KA, IGR
IF (NN(L) .LE. 0) GO TO 10
IF (NN(K) .LE. 0) GO TO 10
SCHEF(K,L) = (AM(L)-AM(K))*2
A = NN(L)
B = NN(K)
SV = QIN*(1./A + 1./B)
SCHEF(K,L) = SCHEF(K,L)/SV
SCHEF(L,K) = SCHEF(K,L)
10 CONTINUE
IGR = IGR - M
RETURN
END
SUBROUTINE FKTRIT(NG, IGR, FW, IAL)
DIMENSION F5(9), F1(9), F01(9)
DIMENSION A5(9), A1(9), A01(9)
COMMON/ALF/ALPHA
DATA A5/0., 0.025, 0.005, 0.000, 0.000, 0.000, 0.000, 0.000, -.005/
DATA A1/0., -.015, -.015, -.015, -.015, -.015, -.015, -.015, -.015/
DATA A01/0., -.45, -.310, -.230, -.190, -.170, -.150, -.130, -.120/
DATA F5/3.84, 3.00, 2.60, 2.37, 2.21, 2.10, 2.01, 1.94, 1.88/
DATA F1/6.83, 4.61, 3.78, 3.32, 3.02, 2.80, 2.64, 2.51, 2.41/
DATA F01/10.83, 6.91, 5.42, 4.62, 4.10, 3.74, 3.47, 3.27, 3.10/
KG = IGR-1
IF (KG .LE. 0) GO TO 100
FF = F5(KG)

```

```

QUE10650
QUE10660
QUE10670
QUE10680
QUE10690
QUE10700
QUE10710
QUE10720
QUE10730
QUE10740
QUE10750
QUE10760
QUE10770
QUE10780
QUE10790
QUE10800
QUE10810
QUE10820
QUE10830
QUE10840
QUE10850
QUE10860
QUE10870
QUE10880
QUE10890
QUE10900
QUE10910
QUE10920
QUE10930
QUE10940
QUE10950
QUE10960
QUE10970
QUE10980
QUE10990
QUE11000
QUE11010
QUE11020
QUE11030
QUE11040

```

```

IF (KG .EQ. 2) FF = FF-0.05
IF (IAL .EQ. 1) FF = F1(KG)
IF (IAL .EQ. 2) FF = F01(KG)
D2 = NG - IGR
IF (D2 .LE. 0.) GO TO 100
IF (D2 .LE. 1500.) GO TO 15
FW = FF
IF (IAL .EQ. 0 .AND. KG .EQ. 2) FW = FW + 0.05
RETURN
15 CONTINUE
D1 = KG
N = 0
P = PRBF(D1,D2,FF)
IF (P .EQ. ALPHA) GO TO 90
20 FF = FF + 0.01
N = N + 1
P = PRBF(D1,D2,FF)
IF (P .LE. ALPHA) GO TO 90
IF (N .GT. 1000) GO TO 200
GO TO 20
90 FW = FF
IF (IAL .EQ. 0) FW = FF + A5(KG)
IF (IAL .EQ. 1) FW = FF + A1(KG)
IF (IAL .EQ. 2) FW = FF + A01(KG)
RETURN
100 FW = 0.
RETURN
200 FW = -99.99
RETURN
END
SUBROUTINE TTEST(N1,N2,T,Z,P,R2)
P1 = N1
P2 = N2
G = N1 + N2
FG = G-2.
TT = T
A = SQRT(ABS(T))
R2 = TT/(TT+FG)
Z = A/2.*SQRT(G/(P1*P2))
P = PRBF(1.0,1000.0,Z**2)

```

```

QUE11050
QUE11060
QUE11070
QUE11080
QUE11090
QUE11100
QUE11110
QUE11120
QUE11130
QUE11140
QUE11150
QUE11160
QUE11170
QUE11180
QUE11190
QUE11200
QUE11210
QUE11220
QUE11230
QUE11240
QUE11250
QUE11260
QUE11270
QUE11280
QUE11290
QUE11300
QUE11310
QUE11320
QUE11330
QUE11340
QUE11350
QUE11360
QUE11370
QUE11380
QUE11390
QUE11400
QUE11410
QUE11420
QUE11430
QUE11440

```

```
P = (1.-P/2.)*100.  
RETURN  
END  
SUBROUTINE DATV  
RETURN  
END
```

```
QUE11450  
QUE11460  
QUE11470  
QUE11480  
QUE11490  
QUE11500
```

## ANHANG B

### Programm TEXT

Mit diesem Programm lassen sich die alpha-numerischen Bezeichnungen von bis zu 1 000 Variablen auf ein File mit der logischen Nummer 15 schreiben. Enthält eine Datei mehr als 1 000 Variable, so ist die Dimensionierung der Matrix KK zu erhöhen, im Fall von 1 500 Variablen z.B. auf DIMENSION KK(1500,80). Zur Kontrolle, ob die Labels an der richtigen Stelle gespeichert werden bzw. ob das Eingabeformat korrekt ist, werden die Textkarten nach Abschluß der Übertragung auf den externen Speicher von File 15 gelesen und mit Angabe der Stelle, auf der sich die jeweiligen Variablen in den Datenkarten (bzw. Kartenimages) befinden, ausgedruckt. (Limitierung: sofern die Zahl der Datenkarten je Fall größer als 12 beträgt, bricht der Kontrollausdruck bei Karte 13 Spalte 40 ab).

#### Eingabe für Programm TEXT

##### 1. Parameterkarte

Spalten	1 - 4	MV	Zahl der Variablen (entspricht Zahl der Textkarten) Maximum: 1 000
	6	IFO	Zahl der Formatkarten (maximal 5) Default: IFO = 1

##### 2. Formatkarte(n)

Es folgen IFO Formatkarten: A-Format. Das Format dient zur Lokalisierung aller Variablen, deren Labels zu lesen und zu speichern sind (s. Kap. 7.11., alpha-numerische Bezeichnungen der Variablen)

##### 3. Textkarten

Für jede Variable ist eine Textkarte einzugeben, die bis zur Spalte 80 gelocht werden kann. Die Textkarten sind in der gleichen Reihenfolge einzulesen, wie die MV Variablen auf den Datenkarten bzw. der sie enthaltenden Datei erscheinen.

C PROGRAMM TEXT (ERSTELLUNG EINES TEXTFILES, DER AUF EINHEIT 15 GESCHRIBEN WIRD  
 DIMENSION KK(1000,80),FM(400),LL(80)

```

DATA KB,ST/IH ,IH//
NN = 0
READ (5,1) MV,IFO
1 FORMAT (I4,IX,I1)
IFO = MAX0(1,IFO)*80
READ (5,2) (FM(J),J=1,IFO)
2 FORMAT (80A1)
WRITE (6,3) MV,(FM(J),J=1,IFO)
3 FORMAT (IH1,39HPROGRAMM ZUR ERSTELLUNG EINES TEXTFILES/
AIH ,26HPROGRAMMAUTOR P. SCHNEIDER/
BIH,20HZAHLE DER VARIABLEN =,I5/
AIH ,20HEINGELESENES FORMAT ,5X,80A1/(IH ,25X,80A1))
KART = 1
DO 61 K = 1,IFO
IF (FM(K) .EQ. ST) KART = KART + 1
61 CONTINUE
WRITE (6,19) KART
19 FORMAT (IH0,36HFUER JEDEN PROBANDEN/FALL EXISTIEREN,I3,I2H DATENKA
ARTEN)
WRITE (6,4)
4 FORMAT (IH1,22HEINGELESENE TEXTKARTEN)
DO 100 I = 1,MV

```

```

READ (5,2,END=150) LL
NN = NN + 1
WRITE (6,5) LL
5 FORMAT (1H ,80A1)
DO 50 J = 1,80
50 KK(I,J) = LL(J)
100 CONTINUE
150 CONTINUE
IF (MV .NE. NN) WRITE (6,16)
16 FORMAT (/1H0,71HZ AHL DER TEXTKARTEN STIMMT NICHT UEBEREIN MIT ANGA
ABE AUF PARAMETERKARTE)
MV = NN
IE = MINO(MV,80)
DO 200 J = 1,80
WRITE (15,FM) (KK(I,J),I=1,MV)
200 CONTINUE
WRITE (6,15)
15 FORMAT (1H1,77HTEST, OB DIE KARTEN MIT DEM RICHTIGEN FORMAT AUF PL
AATTE 15 GESCHRIBEN WURDEN)
WRITE (6,17)
17 FORMAT (1H0,5HKARTE,2X,6HSPALTE,2X,4HTEXT/)
IEND = KART#80
IEND = MINO(IEND,1000)
REWIND 15
DO 249 I = 1,1000

```

```

      DO 249 J = 1,80
249  KK(I,J) = KR
      DO 250 J = 1,80
250  READ (15,2) (KK(I,J), I=1, IFND)
      IA = 1
      IE = 80
      KAR = 1
265  ISP = 0
      DO 270 I = IA,IE
      ISP = ISP + 1
      WRITE (6,21) KAR,ISP,(KK(I,J),J=1,80)
21  FORMAT (1H, I3,4X, I3,5X,80A1)
270  CONTINUE
      IF (IE .GE. IEND) GO TO 290
      IA = IE + 1
      IE = IE + 80
      KAR = KAR + 1
      GO TO 265
290  CONTINUE
      STOP
      END
```

## ANHANG C

### Missing Data Verarbeitung mittels T-Format

Die in QUER3 eingeschlagene Strategie der MD-Verarbeitung über zweifache Zwischenspeicherung der Daten jedes Falles (s. Kap. 7.4.) ist insbesondere bei umfangreichen Dateien sehr zeitaufwendig. Bei Anlagen, für die eine T-Format-Spezifikation möglich ist, läßt sich die Zwischenspeicherung umgehen und die Unterscheidung von Blanks und Nullen direkt an den Ausgangsdaten vornehmen.<sup>1</sup> Die T-Spezifikation arbeitet folgendermaßen: Man liest einen Record (hier: Karte bzw. Kartenimage) im F-Format, springt dann mit Hilfe der T-Anweisung zurück an den Anfang der Karte und liest die gleichen Daten nochmals im A-Format. Für das Format: (4X,20F1.0/4X,10F1.0) sieht die so erweiterte Formatanweisung folgendermaßen aus: (4X,20F1.0,T5,20A1/4X,10F1.0,T5,10A1) bzw. (4X,20F1.0,T1,4X,20A1/4X,10F1.0,T1,4X,10A1).

Sofern die im Anhang C aufgelisteten Programmstatements in das Quellprogramm eingefügt werden, wird das T-Format von QUER3 ermittelt. Dabei sind folgende Regeln für die Formatbildung (Format für Eingabedaten) zu beachten:

- Die Formatanweisung darf weder die Zeichenfolge „/(“ noch die Zeichenfolge „/“ (Ausnahme /) als Abschluß des Formats ist erlaubt) enthalten.

Beispiel: 8 Karten je Fall, auf der 4. Karte sind 10 einspaltige Variablen (Spalten 1 bis 10), auf den Karten 5 bis 7 jeweils 10 zweispaltige Variablen (Spalten 1 bis 20) zu lesen. Das folgende Format kann nicht in ein T-Format umgesetzt werden: (3(/),10F1.0/3(10F2.0/) )

(Unzulässige Zeichenfolgen sind unterstrichen). Statt dessen ist zu schreiben: (///10F1.0/10F2.0/10F2.0/10F2.0/). Wiederholungsfaktoren innerhalb einer Karte, wie z.B. (2(2X,F1.0),1X,3(F1.0,F2.0)/6X,F1.0) sind dagegen zulässig.

<sup>1</sup> Über die MD-Verarbeitung bei Maschinen, die Blanks mit - 0.verschlüsseln, s. Kap. 8)

Sofern ein nicht umwandelbares Format angetroffen wird, erscheint eine entsprechende Meldung im Programmausdruck. Die Missing Data Verarbeitung wird in diesem Fall in der ursprünglichen Weise (zweifache Zwischenspeicherung) vorgenommen. Ist die Umsetzung in ein T-Format vom Programm her durchführbar, so wird das entsprechende Format ausgedruckt, sowie die Zahl der Variablen auf den einzelnen Karten. Zu überlesende Karten werden dabei nicht mitgezählt. Im Beispiel erscheint demnach folgende Meldung:

ZU LESENDER RECORD	ZAHL DER VARIABLEN
1	10
2	10
3	10
4	10
TOTAL	40

Der Benutzer kann die MD-Verarbeitung über T-Format durch die Spezifikation des Parameters KDAT = 2 (Parameterkarte Spalte 17) unterdrücken. In diesem Fall erfolgt die Unterscheidung zwischen Null und Blanks über die in Kap. 7.4. beschriebene Strategie.<sup>1</sup>

Anmerkung:

Da die Daten nicht mehr zweifach zwischengespeichert werden müssen, ist eine Datenkomprimierung (s. Kap. 7.5.) überflüssig.

**Vorgehensweise bei der Programmänderung:**

- a) Die Karten bis zum Programmstatement 760 CONTINUE (Label QUE02875) sind im Hauptprogramm an die durch den Label bezeichneten Stellen zu setzen. Befindet sich bereits eine Karte mit diesem Label im Programm, so ist diese zu ersetzen. Beispiel:

IF (KDAT .LE. 0) GO TO 705 QUE01340 ist zu ersetzen durch

IF (KDAT .NE. 1) GO TO 705 QUE01340.

Im Anhang C aufgelistete Karten, deren Labels im Hauptprogramm nicht anzutreffen sind, werden eingefügt.

<sup>1</sup> Um einen Einblick in die Höhe der Zeitersparnis zu gewinnen wurde auf einer IBM 370/148 folgende Auswertung vorgenommen: 2500 Probanden, 4 Karten je Proband, 100 Variable, davon 1 gruppenbestimmende Variable, 2 Gruppen: Mit der T-Format-Routine ergab sich eine Verkürzung der CPU-Zeit um ca. 40 %.



DIMENSION SCHEF(10,10),KES(100),W(202),TM(900),IVEC(50)

QUE00100

COMMON/MISSD/IVEC,NV,X,Y,h,MM

QUE00225

MM = 2\*MV

QUE00385

IF (KDAT .NE. 1) GO TO 705

QUE01340

JB = 0

QUE01401

IF (KDAT .GT. 0) GO TO 709

QUE01402

CALL TFORM(FM,FMA,TM,IFC,JB,MV,IVEC,NV)

QUE01404

709 CONTINUE

QUE01406

IF (KDAT .NE. 1) GO TO 1204

QUE01544

IF (JB .EQ. 1) GO TO 720

QUE02745

IF (JB .EQ. 1) GO TO 740

QUE02855

GO TO 760

QUE02871

740 CONTINUE

QUE02872

READ (IT, TM) (W(J), J=1, MM)

QUE02873

CALL ZUGRD(MV)

QUE02874

760 CONTINUE

QUE02875

```

SURROUTINE TFORM(FM,FMA,TM,IFD,JA,MV,IVEC,NV)
DIMENSION TM(900),FM(400),FMA(400),X(200),IVEC(50)
DATA STR,AUF,T,ZI,AKC,ZU,BL,F/1H/,1H(,1H,1H),1H,1H,1HF/
N = 0
JA = 1
DO 200 K = 1,IFD
IF (FM(K) .EQ. BL) GO TO 200
N = N + 1
TM(N) = FM(K)
200 CONTINUE
KE = N-2
DO 210 K = 1,KE
IF (FM(K) .NE. STR) GO TO 210
IF (FM(K+1) .NE. ZU) GO TO 201
JA = 0
GO TO 220
201 CONTINUE
IF (FM(K+1) .NE. AUF) GO TO 210
JA = 0
GO TO 220
210 CONTINUE
GO TO 230
220 WRITE (6,1)
1 FORMAT (1H0,50HT-FORMAT KANN VOM PROGRAMM NICHT ERMITTELT WEERDEN,QUE11740
A60H MISSING DATA VERARBEITUNG ERFOLGT UEBER ZWEIFACHE ZWISCHEN-Y QUE11750

```

```

QUE11510
QUE11520
QUE11530
QUE11540
QUE11550
QUE11560
QUE11570
QUE11580
QUE11590
QUE11600
QUE11610
QUE11620
QUE11630
QUE11640
QUE11650
QUE11660
QUE11670
QUE11680
QUE11690
QUE11700
QUE11710
QUE11720
QUE11730
QUE11740
QUE11750

```

B28HSPEICHERUNG JEDES PROBANDEN.)

RETURN

230 CONTINUE

KEND = IFO

DO 5 K = 2, IFO

IF (FM(K) .NE. BL) KEND = K

5 CONTINUE

DO 8 L = 1, 900

8 TM(L) = BL

DO 10 K = 1, IFO

IF (FM(K) .NE. AUF) GO TO 10

IANSF = K + 1

GO TO 11

10 CONTINUE

11 CONTINUE

NF = 0

N = 0

DO 100 K = 1, IFO

IF (FM(K) .EQ. F) NF = 1

IF (K .EQ. KEND) GO TO 31

IF (FM(K) .NE. STR) GO TO 50

IF (FM(K-1) .EQ. STP) GO TO 50

IF (FM(K-1) .EQ. AUF) GO TO 50

IF (NF .EQ. C) GO TO 50

GO TO 60

QUE11760  
 QUE11770  
 QUE11780  
 QUE11790  
 QUE11800  
 QUE11810  
 QUE11820  
 QUE11830  
 QUE11840  
 QUE11850  
 QUE11860  
 QUE11870  
 QUE11880  
 QUE11890  
 QUE11900  
 QUE11910  
 QUE11920  
 QUE11930  
 QUE11940  
 QUE11950  
 QUE11960  
 QUE11970  
 QUE11980  
 QUE11990  
 QUE12000

```

31 CONTINUE
  IF (FM(K-1).EQ. STR) GO TO 50
60 IEND = K
  N = N + 1
  TM(N) = AKO
  N = N + 1
  TM(N) = T
  N = N + 1
  TM(N) = Z1
  N = N + 1
  TM(N) = AKO
  DO 40 L = IANF, IEND
  N = N + 1
  TM(N) = FMA(L)
40 CONTINUE
  NF = 0
  GO TO 100
50 N = N + 1
  TM(N) = FM(K)
  IF (K.LE. 2) GO TO 100
  IF (FM(K-1).EQ. ZU) GO TO 80
  IF (FM(K-1).EQ. STR) GO TO 90
  GO TO 100
80 IF (FM(K-2).NE. STR) GO TO 100
  IANF = K + 1
  QUE12010
  QUE12020
  QUE12030
  QUE12040
  QUE12050
  QUE12060
  QUE12070
  QUE12080
  QUE12090
  QUE12100
  QUE12110
  QUE12120
  QUE12130
  QUE12140
  QUE12150
  QUE12160
  QUE12170
  QUE12180
  QUE12190
  QUE12200
  QUE12210
  QUE12220
  QUE12230
  QUE12240
  QUE12250

```

```

GO TO 100
90 IANF = K
100 CONTINUE
WRITE (6,2) (TM(J),J=1,N)
2 FORMAT (/1H0,50HFOLGENDES T-FORMAT WURDE ZUR UNTERSCHIEDUNG VON BL
A,35HANK UND NULL VOM PROGRAMM ERMITTELT/(1H ,132A1))
MM = 2*MV
DO 300 K = 1,MM
300 X(K) = BL
WRITE (9,FMA) (X(K),K=1,MV)
REWIND 9
READ (9,FM) (X(K),K=1,MM)
IA = I
M = 0
260 L = 0
IF (IA .GT. MM) GO TO 270
M = M + I
DO 240 K = IA,MM
IF (X(K) .NE. 0.) GO TO 250
L = L + I
240 CONTINUE
250 IVEC(M) = L
IA = IA + 2*L
GO TO 260
270 CONTINUE

```

QUE12260  
 QUE12270  
 QUE12280  
 QUE12290  
 QUE12300  
 QUE12310  
 QUE12320  
 QUE12330  
 QUE12340  
 QUE12350  
 QUE12360  
 QUE12370  
 QUE12380  
 QUE12390  
 QUE12400  
 QUE12410  
 QUE12420  
 QUE12430  
 QUE12440  
 QUE12450  
 QUE12460  
 QUE12470  
 QUE12480  
 QUE12490  
 QUE12500

```

NV = M
N = 0
WRITE (6,3)
3 FORMAT (/1H0,49HZAHZ DER VARIABLEN AUF DEN EINZELNEN RECORDS (CHN,
A33HE RECORDS, DIE UEBERLESEN WERDEN)/1H ,18HZU LESENDER RECCRD,5X,
A14HZAHZ VARIABLEN/1H ,37(1H-)/)
DO 290 J = 1,NV
WRITE (6,4) J,IVEC(J)
4 FORMAT (1H ,16,17X,15)
N = N + IVEC(J)
290 CONTINUE
WRITE (6,6) N
6 FORMAT (1H ,28(1H-)/1H ,5HTOTAL,18X,15/)
RETURN
END

```

```

SUBROUTINE ZUORD(MV)
COMMON/MISSD/IVEC,NV,X,Y,W,MM
DIMENSION IVEC(50),X(101),Y(101),W(202)
KE = 0
IE = 0
DO 100 K = 1,NV

```

```

QUE12510
QUE12520
QUE12530
QUE12540
QUE12550
QUE12560
QUE12570
QUE12580
QUE12590
QUE12600
QUE12610
QUE12620
QUE12630
QUE12640
QUE12650

```

```

QUE12660
QUE12670
QUE12680
QUE12690
QUE12700
QUE12710

```

```

IA = IE + 1
N = KE
IE = IVEC(K) + IE
DO 50 L = IA,IE
N = N + 1
50 X(N) = W(L)
IA = IE + 1
IE = IE + IVEC(K)
N = KE
DO 60 L = IA,IE
N = N + 1
60 Y(N) = W(L)
KE = N
100 CONTINUE
IF (KE .EQ. MV) RETURN
WRITE (6,1) KE
1 FORMAT (1H0,41HFEHLER IM EINGABEVEKTOR. PROGRAMMABBRUCH.)
STOP
END

```

```

QUE12720
QUE12730
QUE12740
QUE12750
QUE12760
QUE12770
QUE12780
QUE12790
QUE12800
QUE12810
QUE12820
QUE12830
QUE12840
QUE12850
QUE12860
QUE12870
QUE12880
QUE12890
QUE12900

```

## ANHANG D

### Reduzierung des Kernspeicherbedarfs

Insbesondere bei Maschinen, die eine Verkürzung der Wortlänge mit Hilfe des Statements INTEGER\*2 nicht zulassen, besteht die Gefahr der Überschreitung der Kernspeicherkapazität. Der Platzbedarf läßt sich jedoch durch die folgenden Programmeingriffe vermindern.

#### a) Reduzierung des Vektors IH

Dieser Vektor ist mit 40 000 dimensioniert (INTEGER\*2). Er umfaßt demzufolge 80 000 Bytes bzw. bei CDC-Maschinen 40 000 Worte à 60 Bits. Der Vektor läßt sich verkleinern, was allerdings Auswirkungen auf die zulässige Spannweite der Variablen (s. Kap. 4) mit sich bringt. Sofern z.B. nach 5 Substichproben gruppiert wird und 100 Variable auszuzählen sind, darf die maximale Spannweite bei einem Vektor IH (40 000) gleich  $40\,000/(100 \cdot 5) = 80$  betragen. Verkleinert man IH auf 8 000, so verringert sich die zulässige Spannweite auf  $8\,000/500 = 16$ . Variable mit größerer Spannweite werden in diesem Fall nicht ausgewertet. (Abhilfe: man führt beispielsweise 5 Rechenläufe mit je 20 Variable durch, dadurch erhöht sich die zulässige Spannweite auf  $8\,000/(20 \cdot 5) = 80$ ).

Anmerkung: Bei Veränderung der Dimensionierung von IH (Karte mit Label QUE00190) ist die Konstante NDIM (label QUE00310) auf den gleichen Wert zu reduzieren.

- b) Bedingt durch die Format-gesteuerte Textverarbeitung (s. Kap. 7.11) wird eine Matrix mit  $100 \cdot 80$  Worten (à 4 Bytes) benötigt. Mit Hilfe der EQUIVALENCE-Anweisung lassen sich die Matrizen SU, SU2, X3 und X4, die jeweils mit (20,100) dimensioniert sind, auf die Speicherplätze der Textmatrix legen. Folgende Strategie wird eingeschlagen:
- (1) Die Textmatrix wird von File 15 mit Format (80A1) gelesen und „gekippt“ (s. Kap. 7.11).
  - (2) Die Textmatrix wird auf File 9 mit dem Format (80A1) geschrieben.

- (3) Der Text wird von File 9 in die Matrix TLAB mit dem Format (20A4) eingelesen. Für TLAB benötigt man daher nur 100·20 Worte, was eine Einsparung von 6 000 Worten bedeutet.
- (4) Die Matrix TEXT ist jetzt frei und kann durch die Werte der Matrizen SU, SU2, X3 und X4 überschrieben werden.

**Änderung des Programms:**

Die im Anhang D aufgelisteten Programmkarten sind an die durch die Label gekennzeichneten Stellen im Hauptprogramm einzufügen. Befindet sich bereits eine Karte mit diesem Label im Programm, so ist diese zu ersetzen.

```

DIMENSION TLAB(100,20)
EQUIVALENCE (TEXT(1,1),SU(1,1)),(TEXT(1,21),SU2(1,1)),
A          (TEXT(1,41),X3(1,1)),(TEXT(1,61),X4(1,1))
DO 1265 J = 1,MV
DO 1265 K = 1,20
1265 TLAB(J,K) = BLEER

IF (ITX .EQ. 1) READ (5,5) ((TLAB(I,J),J=1,20),I=1,MV)
5 FORMAT (20A4)

REWIND 9
C ALTE KARTE 50 CONTINUE MIT LABEL QUE01550 ENTFERNEN
DO 1248 J = 1,MV
1248 WRITE (9,3) (TEXT(J,K),K=1,80)
REWIND 9
DO 1249 J = 1,MV
1249 READ (9,5) (TLAB(J,K),K=1,20)
REWIND 9
1210 CONTINUE
50 CONTINUE

```

QUE00151

QUE00226

QUE00227

QUE01271

QUE01272

QUE01273

QUE01530

QUE01540

QUE01549

QUE01550

QUE01551

QUE01552

QUE01553

QUE01554

QUE01555

QUE01556

QUE01557

```

A(TLAB(N,LL),LL=1,20)
67 FORMAT (1H ,I5,4X,I3,2X,A8,1X,F6.0,1X,F6.0,16X,20A4)
    QUE01960
    QUE01970

WRITE (6,9) J,SMALL(J),BIG(J),IV(J),(TLAB(J,L),L=1,14)
    QUE03230

9 FORMAT (1H ,I5,9X,F8.2,7X,F8.2,8X,I5,25X,14A4)
    QUE03330

WRITE (6,46) JJ,(TLAB(JJ,L),L=1,20)
46 FORMAT (/1H0,11HIN VARIABLE,I3,22H SIND NUR MISSING DATA,4X,20A4)
    QUE04400
    QUE04410

173 WRITE (6,39) JJ,(TLAB(JJ,L),L=1,20)
39 FORMAT (1H0,23HSPANNWEITE BEI VARIABLE,I5,9H ZU GROSS,3X,20A4)
    QUE04520
    QUE04530

WRITE (6,48) JJ,(TLAB(JJ,L),L=1,20)
48 FORMAT (/1H0,8HVARIABLE,I5,3X,20A4/1H ,54HSEI VORLIEGENDER GRUPPENQUE05230
    QUE05220
    QUE05230

295 WRITE (6,11) JJ,(TLAB(JJ,L),L=1,20)
11 FORMAT (3(/,9H VARIABLE,I5,3X,20A4/1H0,26X,17HMITTEL- STANDARD-, QUE06080
    QUE06100

```