Häufigkeitstabellen mit PROC SQL – Eine Alternative?

Murat Ipek
PAREXEL International
Spandauer Damm 130
14050 Berlin
Murat.Ipek@parexel.com

Zusammenfassung

Häufigkeitstabellen dienen der übersichtlichen Darstellung von Daten. Der Leser möchte neben den absoluten Werten eines Merkmals – wie beispielsweise die Anzahl der männlichen Teilnehmer an einer Befragung – auch deren Verteilung in der erhobenen Stichprobe erfahren – wie beispielsweise 23 der 59 Befragten (≈39%) sind männlich. Bei einem eindeutigen Merkmal wie dem Geschlecht ist die Berechnung unkompliziert und kann i.d.R. mit einem einzigen Prozedurenschritt in SAS berechnet werden. Betrachtet man Merkmale, die wiederholt erhoben/beobachtet werden, wie beispielsweise das Auftreten von Kopfschmerzen in einer Interventionsstudie, so möchte man neben der absoluten und relativen Anzahl der Ereignisse auch wissen, bei wie vielen Personen mindestens einmal Kopfschmerzen beobachtet wurden. Eine Darstellung dieser Art ist im Vorfeld verbunden mit einigen Datenmanipulationen, um die Daten in die gewünschte Struktur zu bringen. Neben Summary-Prozeduren bietet SAS mit PROC SQL eine alternative Möglichkeit, Häufigkeitstabellen effizient und ohne Datenmanipulation zu erstellen.

Schlüsselwörter: Häufigkeitstabellen, PROC SQL, Zusammenführung von Dateien, Join

1 Motivation und Ziel

Ziel dieses ist es, dem Leser neben den Standardprozeduren in SAS®, wie PROC FREQ, PROC UNIVARIATE oder PROC TABULATE, eine weitere Möglichkeit aufzuzeigen, Häufigkeiten zu berechnen. Es ist keinesfalls als Ersatz für die etablierten Prozeduren zu verstehen. Allerdings bietet SAS® mit PROC SQL eine Prozedur an, mit der umfangreiche und vor allem komplexe Operationen durchgeführt werden können.

Dieser Beitrag beschränkt sich auf die Berechnung von Häufigkeiten mit PROC SQL, enthält allerdings in Abschnitt 3 eine kurze Einführung über das Zusammenfügen von Tabellen, Werten oder Listen. Für weitergehende Information und insbesondere das Einlesen in die Eigenschaften von PROC SQL sei auf die SAS®-Hilfe in [1] verwiesen.

1.1 Ausgangslage

Gegeben sind Beobachtungen von unerwünschten Ereignissen, die während einer Interventionsstudie erhoben wurden. Es wurden neben der Start- und Endzeit auch die be-

treffende Körperfunktion sowie die Erkrankung festgehalten wie in der folgenden Tabelle dargestellt.

Tabelle 1: Ausgangslage: Datensatz mit beobachteten unerwünschten Ereignissen

Patient	No.	Körperfunktion	Erkrankung	Startzeit punkt	Endzeit punkt
KSFE18-002	1	Nervensystem-Erkrankung	Kopfschmerzen	2014-02-23	2014-02-23
KSFE18-004	1	Nervensystem-Erkrankung	Schwindelanfall	2014-02-01	2014-02-01
KSFE18-004	2	Nervensystem-Erkrankung	Kopfschmerzen	2014-02-09	2014-02-09
KSFE18-004	3	Nervensystem-Erkrankung	Kopfschmerzen	2014-02-15	2014-02-15
KSFE18-007	1	Allgemeine-Erkrankung	Muedigkeit	2014-02-03	2014-02-05
KSFE18-007	2	Herz-Kreislauf-Erkrankung	Herzrasen	2014-02-04	2014-02-04
KSFE18-008	1	Magen-Darm-Erkrankung	Bauchschmerzen	2014-02-04	2014-02-04
KSFE18-009	1	Magen-Darm-Erkrankung	Bauchschmerzen	2014-02-04	2014-02-04
KSFE18-009	2	Allgemeine-Erkrankung	Influenza	2014-02-07	2014-02-20
KSFE18-011	1	Nervensystem-Erkrankung	Kopfschmerzen	2014-02-03	2014-02-04
KSFE18-011	2	Infektions-Erkrankung	Nasopharyngitis	2014-02-04	2014-02-17
KSFE18-011	3	Nervensystem-Erkrankung	Kopfschmerzen	2014-02-17	2014-02-18
KSFE18-011	4	Nervensystem-Erkrankung	Kopfschmerzen	2014-02-17	2014-02-18

Das SAS-System bietet verschiedene Prozeduren an, um Häufigkeiten von Ausprägungen zu berechnen, genannt sei hier beispielsweise die Prozedur PROC FREQ. Leider gibt es Einschränkungen, die die Benutzung von PROC FREQ für unsere Belange nicht nur erschweren sondern teilweise unmöglich machen, ohne vorherige Datenmanipulationen vorzunehmen.

Eine der Schwierigkeiten beruht auf dem Sachverhalt, dass ein Studienteilnehmer mehr als ein Ereignis berichten kann, dieser aber in der späteren Darstellung unterschiedlich gezählt werden muss (sei beispielsweise Teilnehmer 011 genannt mit 4 beobachteten Ereignissen). In der erwarteten Übersichtsdarstellung werden identische Beobachtungen der Studienteilnehmer in der kleinsten Kategorie (hier Erkrankung) stets nur einmal gezählt, wogegen alle beobachteten Ereignisse in einer zweiten Darstellung berücksichtigt werden.

Hinzu kommt, dass für die Ermittlung der relativen Häufigkeit der Studienteilnehmer unterschiedliche Annahmen für die Population (i.d.R. mit N angegeben) getroffen wer-

den können. Meist wird definiert, ob alle Studienteilnehmer einer klinischen Studie zur Ermittlung der relativen Häufigkeit herangezogen werden, also auch die, die kein Ereignis beobachtet haben, oder man sich auf die beschränkt, die ein Ereignis beobachtet haben. In unserem Fall haben alle Studienteilnehmer, es sind insgesamt 6 an der Zahl (N=6), mindestens ein Ereignis beobachtet.

Die folgende Tabelle zeigt die Übersichtstabelle der Ereignisse, die wir im Folgenden erstellen werden. Die Tabelle ist gegliedert nach Körperfunktion und Erkrankung. Ebenfalls abgebildet sind die Häufigkeiten für die Körperfunktion, angegeben mit „Gesamt". In der Ergebnisspalte n (%) können wir die Anzahl der Studienteilnehmer in der entsprechenden Kategorie ablesen. Die Spalte e (%) gibt die Anzahl der Ereignisse wieder.

Tabelle 2: Ziel: Häufigkeitstabelle mit Angabe der #Studienteilnehmer und #Ereignisse

		Ergebnis	
Körperfunktion	**Erkrankung**	**n (%)**	**e (%)**
Beobachtete Ereignisse Gesamt		6 (100.00)	13 (100.00)
Allgemeine- Erkrankung	Gesamt	2 (33.33)	2 (15.38)
	Influenza	1 (16.67)	1 (7.69)
	Muedigkeit	1 (16.67)	1 (7.69)
Herz-Kreislauf- Erkrankung	Gesamt	1 (16.67)	1 (7.69)
	Herzrasen	1 (16.67)	1 (7.69)
Infektions- Erkrankung	Gesamt	1 (16.67)	1 (7.69)
	Nasopharyngitis	1 (16.67)	1 (7.69)
Magen-Darm- Erkrankung	Gesamt	2 (33.33)	2 (15.38)
	Bauchschmerzen	2 (33.33)	2 (15.38)
Nervensystem- Erkrankung	Gesamt	3 (50.00)	7 (53.85)
	Kopfschmerzen	3 (50.00)	6 (46.15)
	Schwindelanfall	1 (16.67)	1 (7.69)

2 Häufigkeiten berechnen mit PROC SQL

In diesem Abschnitt wird Schritt für Schritt die Tabelle 2 mit der Prozedur SQL produziert.

Im ersten Schritt ist es Notwendig die Population zu ermitteln, d.h. in Erfahrung zu bringen wie viele Studienteilnehmer insgesamt zur Auswertung herangezogen werden. Die Ereignisse geben wir mit einem großen E an, wobei E alle beobachteten Ereignisse ohne Ausnahme wiedergibt.

Mit Hilfe des folgenden SAS-Codes können wir N und E basierend auf dem Datensatz _10calc ermitteln:

```
PROC SQL;
  CREATE TABLE _50calc_NE AS
    SELECT COUNT(DISTINCT usubjid) AS bigN LABEL="N"
          ,COUNT(*)                AS bigE LABEL="E"
      FROM      _10calc
  ;
QUIT;
```

Wir errechnen, dass N=6 Studienteilnehmer insgesamt E=13 Ereignisse beobachtet haben.

Bedient haben wir uns der Zählfunktion COUNT(), die mit der zusätzlichen Option DISTINCT identische Einträge nur einmal zählt, ähnlich der Option NODUPKEY, die in der Prozedur PROC SORT Verwendung findet. Die Syntax der COUNT()-Funktion, gegeben wie folgt, COUNT(<DISTINCT> varname|*), erlaubt die Benennung einer Variable. In diesem Fall werden nur die Beobachtungen gezählt werden, die nicht MISSING sind.

2.1 Beobachtete Ereignisse Gesamt

Zur Berechnung der „Beobachtete Ereignisse Gesamt“, die eine Übersichtsangabe ist, können wir genauso vorgehen wie im vorherigen Abschnitt dargestellt. Unterschiede kann es lediglich bei der Definition der Population geben. Es kann sein, dass der Datensatz zur Ermittlung von N und E verschieden ist. Das stellt kein Problem dar, da in SQL das Zusammenfügen von Tabellen sehr schön geregelt ist. Es können Werte oder Listen aus unterschiedlichen Tabellen benutzt werden, ohne dass diese in die resultierende Tabelle aufgenommen werden müssen. Daher können N und E auch unabhängig voneinander berechnet und im Anschluss zusammengeführt werden. Näheres über das Zusammenfügen von Tabellen in Abschnitt 3 später.

2.2 Häufigkeiten gruppiert nach einer oder mehreren Variablen

In unserem Beispiel müssen wir die Häufigkeiten für Gruppen, einmal getrennt für den Gruppenoverall „Körperfunktion“ und einmal für die Untergruppe „Erkrankung“ in Abhängigkeit von „Körperfunktion“ berechnen. Dabei werden wir neben der Anzahl der Studienteilnehmer (n) auch die Anzahl der Ereignisse (e) in der jeweiligen Kategorie berechnen. Wie bereits erwähnt werden die Studienteilnehmer eindeutig ausgegeben, d.h. falls ein Studienteilnehmer (wie unserem Beispiel der Teilnehmer 011) mehr als ein Ereignis beobachtet hat, wird dieser nur einmal gezählt wobei alle Ereignisse für e berücksichtigt werden.

PROC SQL bietet mit der Option GROUP BY die Möglichkeit Aggregatfunktionen auf Gruppen oder Tabellen, falls keine Gruppiervariablen angegeben wurden, anzuwenden. Aggregatfunktionen sind sogenannte Summary-Funktionen, zu der auch die COUNT()-

Funktion gehört. Sie haben die Eigenschaft, alle Werte einer Observation oder einer Spalte auf eine einzige Zahl oder „Aggregat“ zu reduzieren.

Als Beispiel sei genannt, dass die Summe einer Spalte (ein einzelner Wert) aus der Addition aller Einzelwerte dieser Spalte berechnet wird (Aggregat).

Genannt sei auch, dass, im Gegensatz zu vielen anderen Prozeduren, die Daten in PROC SQL bei der Anwendung von Gruppierungen entsprechend der Gruppiervariablen nicht sortiert sein müssen, da PROC SQL diese automatisch handhabt.

Mit Hilfe der Gruppierung berechnen wir nun die Gesamtzahl für die Gruppen „Körperfunktion“ durch Hinzunahme des folgenden SAS®-Codes:

```
PROC SQL;
  CREATE TABLE _65calc_resume AS
    SELECT aebodsys                AS koerperkt
          ,"Gesamt"                AS erkrankung
          ,COUNT(DISTINCT usubjid) AS n
          ,COUNT(*)                AS e
      FROM _10calc
        GROUP BY aebodsys
  ;
QUIT;
```

Gemäß der vorgegebenen Syntax GROUP BY variable1 <, variable2, … ,variableN> berechnen wir mit der COUNT()-Funktion die Anzahl der beobachteten Studienteilnehmer (n) und die Anzahl der Ereignisse (e) gruppiert nach den Ausprägungen der Variablen AEBODSYS (=Körperfunktion). Damit unsere Ausgabe zusätzliche Informationen für den späteren Leser oder Reviewer beinhaltet, erstellen wir eine weitere Variable ERKRANKUNG die die Zeichenkette „Gesamt“ beinhaltet.

Genauso verhält es sich, wenn wir die Häufigkeiten gruppiert nach „Körperfunktion“ und „Erkrankung“ darstellen möchten. Der folgende SAS®-Code zeigt die Berechnung:

```
PROC SQL;
  CREATE TABLE _60calc_neByKoerperfktErkrankung AS
    SELECT aebodsys                AS koerperkt
          ,aedecod                 AS erkrankung
          ,COUNT(DISTINCT usubjid) AS n
          ,COUNT(*)                AS e
      FROM _10calc
        GROUP BY aebodsys, aedecod
  ;
QUIT;
```

Halten wir nun in der folgenden Abbildung fest, was wir bis hierher berechnet haben:

VIEWTABLE: Work._65calc_resume

	Körperfunktion	Erkrankung	N	E	n	e
1	ALLGEMEINE-ERKRANKUNG	Gesamt	6	13	2	2
2	ALLGEMEINE-ERKRANKUNG	INFLUENZA	6	13	1	1
3	ALLGEMEINE-ERKRANKUNG	MUEDIGKEIT	6	13	1	1
4	HERZ-KREISLAUF-ERKRANKUNG	Gesamt	6	13	1	1
5	HERZ-KREISLAUF-ERKRANKUNG	HERZRASEN	6	13	1	1
6	INFEKTIONSERKRANKUNG	Gesamt	6	13	1	1
7	INFEKTIONSERKRANKUNG	NASOPHARYNGITIS	6	13	1	1
8	MAGEN-DARM-ERKRANKUNG	Gesamt	6	13	2	2
9	MAGEN-DARM-ERKRANKUNG	BAUCHSCHMERZEN	6	13	2	2
10	NERVENSYSTEM-ERKRANKUNG	Gesamt	6	13	3	7
11	NERVENSYSTEM-ERKRANKUNG	KOPFSCHMERZEN	6	13	3	6
12	NERVENSYSTEM-ERKRANKUNG	SCHWINDELANFALL	6	13	1	1

Abbildung 1: Ergebnisse der bisherigen Berechnungen

Bekannt sind die

- Population N und die Gesamtanzahl aller Ereignisse E.
- beobachtete Anzahl der Studienteilnehmer (n) und der Ereignisse (e) Gesamt für die Kategorie „Körperfunktion“.
- beobachtete Anzahl der Studienteilnehmer (n) und der Ereignisse (e) gruppiert nach „Körperfunktion“ und „Erkrankung“.

Die einzelnen (Teil-) Ergebnisse sind hier bereits zusammengefügt. Die Prozedur SQL bietet für die Zusammenfassung von Tabellen verschiedene Möglichkeiten an, die in Abschnitt 3 kurz erwähnt werden.

2.3 Relative Häufigkeiten berechnen

Nahezu alle Funktionen zur Stringmanipulation sind in der Prozedur PROC SQL anwendbar. Hierzu sei für weitergehende Informationen auf die SAS®-Hilfe [1] verwiesen.

Der folgende SAS®-Code zeigt die Berechnung der relativen Häufigkeiten, die mit n/N*100 für die Studienteilnehmer sowie e/E*100 für die Ereignisse definiert sind:

```
PROC SQL;
  CREATE TABLE _70calc_percentages AS
    SELECT *
          ,PUT(STRIP(PUT(n/bigN*100,12.2)),$CHAR6. -R)  AS npercent
          ,PUT(STRIP(PUT(e/bigE*100,12.2)),$CHAR6. -R)  AS epercent
          ,CAT(n," (",CALCULATED npercent,")")          AS np
          ,CAT(e," (",CALCULATED epercent,")")          AS ep
      FROM _65calc_resume
  ;
QUIT;
```

Mit der PUT()-Funktion findet die Umwandlung von String-Variablen in numerische Variablen statt wobei mit der STRIP()-Funktion und der Kombination mit dem Format $CHAR6 und der Option –R die Ausrichtung nach rechts erfolgt. Die relativen Häufigkeiten sind hier mit zwei Nachkommastellen dargestellt.

CALCULATED erlaubt die in derselben SELECT-Abfrage erstellten „neuen“ Variablen augenblicklich zu verwenden, um weitere Operationen durchzuführen.

Das Ergebnis der String-Manipulation ist wie folgt abgebildet:

VIEWTABLE: Work._70calc_percentages

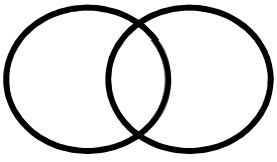
	Körperfunktion	Erkrankung	N	E	n	e	np	ep
1	ALLGEMEINE-ERKRANKUNG	Gesamt	6	13	2	2	2 (33.33)	2 (15.38)
2	ALLGEMEINE-ERKRANKUNG	INFLUENZA	6	13	1	1	1 (16.67)	1 (7.69)
3	ALLGEMEINE-ERKRANKUNG	MUEDIGKEIT	6	13	1	1	1 (16.67)	1 (7.69)
4	HERZ-KREISLAUF-ERKRANKUNG	Gesamt	6	13	1	1	1 (16.67)	1 (7.69)
5	HERZ-KREISLAUF-ERKRANKUNG	HERZRASEN	6	13	1	1	1 (16.67)	1 (7.69)
6	INFEKTIONSERKRANKUNG	Gesamt	6	13	1	1	1 (16.67)	1 (7.69)
7	INFEKTIONSERKRANKUNG	NASOPHARYNGITIS	6	13	1	1	1 (16.67)	1 (7.69)
8	MAGEN-DARM-ERKRANKUNG	Gesamt	6	13	2	2	2 (33.33)	2 (15.38)
9	MAGEN-DARM-ERKRANKUNG	BAUCHSCHMERZEN	6	13	2	2	2 (33.33)	2 (15.38)
10	NERVENSYSTEM-ERKRANKUNG	Gesamt	6	13	3	7	3 (50.00)	7 (53.85)
11	NERVENSYSTEM-ERKRANKUNG	KOPFSCHMERZEN	6	13	3	6	3 (50.00)	6 (46.15)
12	NERVENSYSTEM-ERKRANKUNG	SCHWINDELANFALL	6	13	1	1	1 (16.67)	1 (7.69)

Abbildung 2: Berechnung der relativen Häufigkeiten

3 Zusammenführen in PROC SQL

Sicherlich fragen Sie sich, wieso die Prozedur PROC SQL benutzt werden soll, wo es doch andere Prozeduren in SAS® gibt um dieselben Resultate zu erhalten. Eine Besonderheit der Prozedur SQL ist die Möglichkeit, Tabellen auf verschiedenste Arten zusammen zu führen und diese mit einfachen bis komplexen Bedingungen zu verknüpfen. Diese sollen im Folgenden anschaulich Anhand der Möglichkeiten der Zusammenführung von Tabellen, Werten oder Listen gezeigt werden.

Verallgemeinert gibt es zwei Wege der Zusammenführung. Zum einen die horizontale Zusammenführung ähnlich dem MERGE-Statement im DATA-Step, um zwei oder mehr Tabellen horizontal miteinander zu verbinden, um neue Eigenschaften (Spalten) zu erhalten. Zum anderen die Möglichkeit Tabellen vertikal zusammen zu führen ähnlich dem SET-Statement im DATA-Step, um zwei oder mehr Tabellen vertikal zu verbinden, um mehr Beobachtungen (Zeilen) zu erhalten.



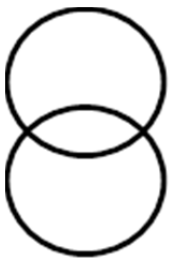
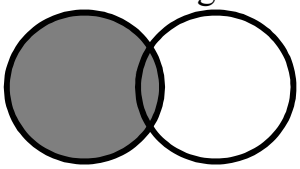


Abbildung 3 und 4: Schematische Darstellung der horizontalen und vertikalen Zusammenführung

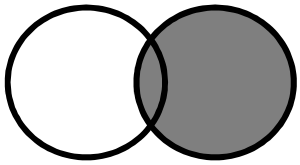
Neben der Zusammenführung bietet die Prozedur SQL auch die Möglichkeit Unterabfragen, sogenannte Subqueries, zu erstellen. Eine Subquery ist eine Abfrage die in einer anderen Abfrage eingeschlossen ist. Sie kann einen Wert oder eine Werteliste zurückgeben. Mit dieser Eigenschaft bietet die SQL Prozedur endlich viele Möglichkeiten an, die täglichen Herausforderungen zu meistern.

3.1 Die horizontale Zusammenführung

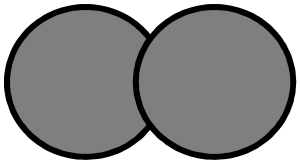
Die SQL Prozedur bietet vier Möglichkeiten der horizontalen Zusammenführung.



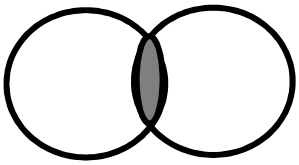
1. **LEFT JOIN**: Die linke Tabelle wird vollständig übernommen. Von der rechten Tabelle werden nur passende Zeilen übernommen.



2. **RIGHT JOIN**: Die rechte Tabelle wird vollständig übernommen. Von der linken Tabelle werden nur passende Zeilen übernommen.



3. **FULL JOIN**: Beide Tabellen werden vollständig übernommen. Zeilen die nicht zusammenpassen erzeugen in der jeweils anderen Tabelle Leerzellen (MISSINGs).



4. **INNER JOIN**: Es werden alle Zeilen übernommen, deren Verbindungswerte in beiden Tabellen identisch sind.

Abbildung 5-8: Schematische Darstellung der verschiedenen horizontalen Zusammenführungen

In Abschnitt 1.1 wurde erwähnt, dass verschiedene Annahmen zur Population und somit für N getroffen werden können. Mit der Prozedur PROC SQL können verschiedene Tabellen durch die horizontale Zusammenführung, optional verknüpft mit einer Unterabfrage, Werte- und/oder Listen zurückgegeben werden, die auf unterschiedlichen Datensätzen beruhen.

Mit Hilfe des folgenden SAS®-Codes werden unterschiedliche Datensätze benutzt, um die Population N und die Gesamtanzahl der Ereignisse E zu ermitteln. Für die Population N wird der in der Makrovariablen &sbjD. hinterlegte Datensatz verwendet, wogegen der in der Makrovariablen &evntD. hinterlegte Datensatz für die Berechnung der Ereignisse E verwendet wird.

Hinzu kommt, dass N und E in zwei separaten Subqueries berechnet und mit Hilfe der horizontalen Zusammenführung, in diesem Fall dem LEFT-Join, zusammengeführt werden. Somit bleibt das Grundgerüst unberührt ganz gleich welche Quelle für N oder E verwendet wird.

```
PROC SQL;
 CREATE TABLE _80joinAndUnion AS
  SELECT DISTINCT
    "Beobachtete Ereignisse Gesamt" AS koerperfkt
   ," "                                                                AS erk
   ,PUT(STRIP(PUT(COUNT(DISTINCT usubjid),BEST.)),$CHAR3. -R) AS n
   ,CALCULATED n / bN.bigN * 100                                       AS np
   ,PUT(STRIP(PUT(COUNT(*),BEST.)),$CHAR3. -R)                         AS e
   ,CALCULATED e / bE.bigE * 100                                       AS ep
  FROM indata
      LEFT JOIN (SELECT COUNT(DISTINCT usubjid) AS bigN
                   FROM &sbjD.
                ) AS bN
        ON 1
      LEFT JOIN (SELECT COUNT(*) AS bigE
                   FROM &evntD.
                ) AS bE
        ON 1
  ;
QUIT;
```

Jede horizontale Verknüpfung ist an Bedingungen geknüpft, die mit ON definiert werden. Im obigen Beispiel ist das Resultat der Subquery ein einzelner Wert wodurch die Bedingung für das horizontale Zusammenfügen mit 1 stets Wahr ist.

Wenn zwei oder mehrere Tabellen miteinander verbunden werden, wird die Zusammenführung „fast“ immer über die angegebenen ID-Variablen der Tabellen geregelt. Um fehlende Werte in den ID-Variablen bei einer horizontalen Zusammenführung zu vermeiden, sollten ID-Variablen immer vom Hauptdatensatz in den resultierenden Datensatz übernommen werden. Bei einem LEFT-JOIN von dem erstgenannten Datensatz und bei einem RIGHT-JOIN vom zuletzt genannten Datensatz. Bei einem FULL-JOIN empfiehlt es sich auf die Funktion COALESCE() zurück zu greifen, um sicherzustellen das einer der beiden Werte aus den ID-Variablen übernommen wird und fehlende Werte in der ID-Variable vermieden werden. Weitere Information zu den FULL JOIN und der COALESCE()-Funktion finden sie in der SAS Hilfe. Bei der INNER-JOIN dürfen die ID-Variablen nicht leer sein, daher ist die Auswahl der Variable in diesem Fall gleichgültig.

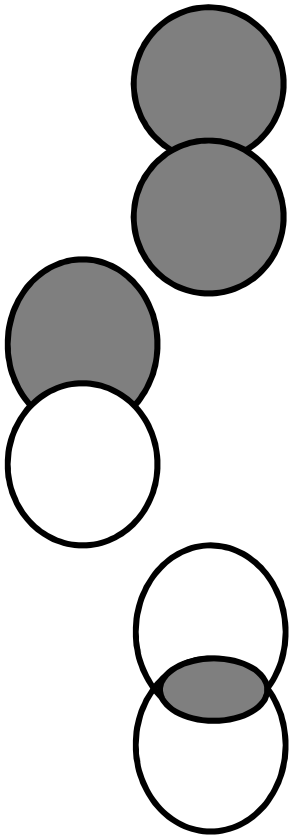
Für komplexere Zusammenführungen können mehrere Bedingungen durch eine UND-Bedingung (Schlüsselwort AND) innerhalb der ON-Bedingung abgefragt werden, dass sich wie eine WHERE-Bedingung verhält. Folgender SAS®-Code zeigt beispielhaft für mehrere Bedingungen das Zusammenführen von Tabellen, Werten oder Listen:

```
  FROM indata                  AS l
      LEFT JOIN sashelp.class  AS r
        ON l.usubjid  = r.name
       AND l.aebodsys = r.aebodsys
       AND l.age      > 50
```

Zusätzlich zur nominalen Skala mit Gleich / Ungleich kann die Bedingung ordinaler Natur sein und auf größer (und größer-gleich) beziehungsweise kleiner (und kleiner-gleich) abgefragt werden, wodurch sich auch komplexe Bedingungen programmieren lassen.

3.2 Vertikale Zusammenführung

Die SQL Prozedur bietet drei Möglichkeiten der vertikalen Zusammenführung an.



UNION: Alle Zeilen aus beiden Tabellen werden übernommen. Mit der Option OUTER werden auch die doppelten Zeilen übernommen. Mit der zusätzlichen Option CORR werden alle Variablen durch ihren Namen zugeordnet, anstatt ihrer Position.

EXCEPT: Alle Zeilen aus der oberen Tabelle, die nicht in der unteren Tabelle vorkommen werden übernommen. Mit der zusätzlichen Option CORR werden alle Variablen gelöscht, die in den jeweils anderen Datensätzen nicht vorkommen.

INTERSECT: Nur die Zeilen, die in beiden Tabellen identisch sind werden übernommen. Mit der zusätzlichen Option CORR werden alle Variablen gelöscht, die in den jeweils anderen Datensätzen nicht vorkommen.

Abbildung 9-11: Schematische Darstellung der verschiedenen vertikalen Zusammenführungen

Für unsere Zwecke bedienen wir uns dem UNION in Kombination mit den beiden Optionen OUTER und CORR. Mit Hilfe dessen werden die einzelnen Ergebnisse aus Abschnitt 2 vertikal zu einem einzigen resultierenden Datensatz zusammengefügt. Der folgende SAS-Code zeigt Schematisch, wie Datensätze mit Hilfe von PROC SQL vertikal zusammengefügt werden können.

```
PROC SQL;
 CREATE TABLE schemaUnion AS
  SELECT *
    FROM sashelp.class

 OUTER UNION CORR

  SELECT *
    FROM sashelp.class
QUIT;
```

Somit enthält der resultierende Datensatz schemaUnion die Einträge aus dem Datensatz sashelp.class doppelt.

4 Schematische Darstellung des Programmaufbaus

Die folgende Grafik zeigt Schematisch wie die einzelnen Schritte zur Berechnung der Häufigkeiten mit PROC SQL durchgeführt werden können.

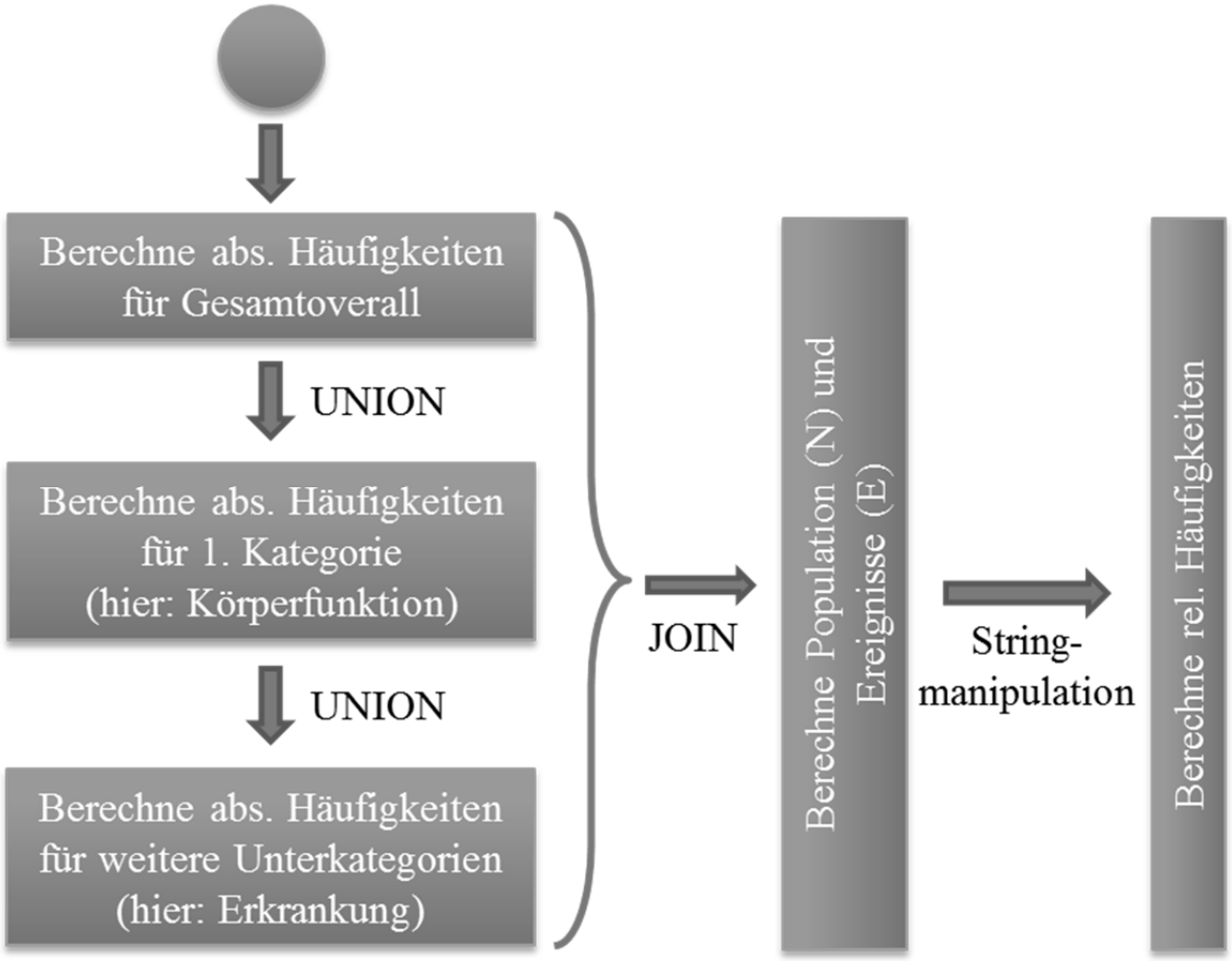


Abbildung 12: Schematische Darstellung über den Aufbau des SAS® Programms zum Erreichen des Ziels (Tabelle 2 im Abschnitt 1.1)

Es empfiehlt sich allerdings von vornherein Gedanken über die spätere Sortierung zu machen, damit die Ausgabe in sich Stimmig ist.

Literatur

[1] SAS Institute Inc., SAS® 9.3 SQL Procedure User's Guide, 2nd printing, August 2012, ISBN 978-1-60764-892-5, SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513