Makros zur Kombination von deskriptiven und analytischen Ergebnissen aus klinisch-/epidemiologischen Studien

Heribert Ramroth
Universitätsklinikum Heidelberg,
Institut für Public Health
Im Neuenheimer Feld 324
69120 Heidelberg
Heribert.Ramroth@uni-heidelberg.de

Zusammenfassung

Hintergrund: Die übersichtliche Darstellung von Studienergebnissen ist sowohl für Publikationen als auch bei der Präsentation in der alltäglichen Praxis für Berichte und Besprechungen sinnvoll und wünschenswert. Prozeduren der analytischen Statistik beinhalten jedoch nicht die Häufigkeitsverteilungen der einbezogenen Variablen. Ein einfaches SAS Programm zur sinnvollen Kombination von Häufigkeiten und Risikokennwerten wurde auf der KSFE 2008 vorgestellt. Ziel des Programms war zu demonstrieren, dass es unter Anwendung von SAS ODS für die Darstellung der Ergebnistabellen ohne zusätzliche manuelle Eingabe bzw. Format-Nachbearbeitung möglich ist, beschreibende und analytische Ergebnisteile übersichtlich zu kombinieren. Dies soll nun unter Verwendung von SAS Makros verallgemeinert dargestellt werden.
Für die deskriptive Auswertung wird aufgrund der einfachen Syntax PROC FREQ verwendet. Hierbei werden absolute und relative Häufigkeiten in ansprechenden Formaten ausgegeben, so dass in der Ausgabetabelle keine manuelle Nachbearbeitung mehr nötig ist.
Der analytische Schwerpunkt liegt auf einem Makro unter Anwendung von PROC LOGISTIC, stellvertretend für alle Prozeduren, die ein CLASS Statement verwenden. Im Gegensatz zu Prozeduren, die ohne CLASS Statement arbeiten, muss hier nicht zwischen binären und kategorialen Variablen mit mehr als 2 Kategorien unterschieden werden [1].
Auch die Ergebnisse der analytischen Prozedur werden in einer Ausgabetabelle gespeichert, in der die Schätzparameter zusammen mit den zugehörigen 95%-Konfidenzintervallen in wiederum optisch ansprechendem Format ausgegeben werden. Dies macht, wie im deskriptiven Teil, eine manuelle Nachbearbeitung der numerischen Ergebnisse überflüssig.
Die hier verwendete Prozedur LOGISTIC lässt sich leicht durch andere analytische Prozeduren ersetzen, die ein CLASS Statement verwenden, so dass sich der vorgestellte Ansatz auf andere Studienformen mit unterschiedlichen Auswertungsprozeduren übertragen lässt.
Ein zweites Makro mit Anwendung der Prozedur PHREG unter Verwendung von Dummy-Variablen, somit ohne Verwendung eines CLASS Statements, wird angedeutet.
Aufgrund der Änderungen in den Variablennamen der zur Aufbereitung der Ergebnisse verwendeten ODS OUTPUT Tabellen von SAS 9.1 auf SAS 9.2, wird das hier vorgestellte Makro in der aktuellen Version SAS 9.2 realisiert. Da die Schätzparameter abhängig von der analytischen Prozedur außerdem sowohl in unterschiedlich benannten, als auch unterschiedlich konstruierten ODS OUTPUT Tabellen ausgegeben werden, ist das hier gezeigte Vorgehen entweder für jede verwendete analytische Prozedur in leichter Weise anzupassen, oder das Makro entsprechend aufwendiger zu erweitern.
Anhand der Daten einer häufigkeitsgematchten Fall-Kontroll-Studie wird das Vorgehen für kategoriale (inkl. binäre) und stetige Variablen demonstriert. Makro und Beispieldatensatz lassen sich per Kopieren/Einfügen in SAS testen.

Schlussfolgerung: Beschreibende und analytische Prozeduren sind austauschbar, sodass sich ein allgemeiner Ansatz für andere Studienformen ableiten lässt. Die Konstruktion eines Makros realisiert die Anwendung der Methodik auf ähnliche Datensituationen. Da die analytischen Ergebnisse bei kategorialen Variablen immer zusammen mit den dem Modell zugrundeliegenden Häufigkeiten interpretiert werden müssen, wäre die Integration der hier aufgeführten Methodik in die SAS Syntax eine sinnvolle Alternative für die Zukunft.

Schlüsselwörter: ODS OUTPUT, kategoriale Variablen, stetige Variablen, PROC FREQ, PROC LOGISTIC, CLASS Anweisung, SAS 9.2, FREQ-Prozedur, LOGISTIC-Prozedur

1 Einleitung

Am Ende der Auswertung von klinischen und epidemiologischen Studien steht die übersichtliche Darstellung und Publikation der Ergebnisse. Die Übertragung der Ergebnisse aus den beschreibenden bzw. analytischen SAS-Prozeduren in eine publikationsfähige Tabelle erfolgt dabei oft manuell bzw. unter manueller Nachbearbeitung des Ergebnisformats im verwendeten Textverarbeitungsprogramm. Ziel dieses Beitrages ist zu zeigen, dass sich der auf der KSFE 2008 vorgestellte Ansatz [1], unter Verwendung von SAS ODS in der endgültig präsentierten Ergebnistabelle beschreibende und analytische Ergebnisteile übersichtlich innerhalb eines SAS-Programms ohne zusätzliche manuelle Bearbeitung zu kombinieren, auf einfache Art im Makro realisieren lässt. Da die meisten der analytischen Prozeduren seit SAS 9.2 CLASS Statements verwenden, ist die 2008 vorgestellte Idee noch leichter für diese Prozeduren nachvollziehbar und realisierbar.

2 Methodik

Die Methode lässt sich in drei Schritten innerhalb eines SAS-Programms strukturieren:
(i) Die Beschreibung des Studienkollektivs mittels kategorialer Variablen unter Verwendung von PROC FREQ. Eine sinnvolle Kombination von absoluten Werten mit den entsprechenden Prozentangaben in Klammern zur Ergebnisdarstellung wird mit einer einfachen Nachbearbeitung der von PROC FREQ via SAS ODS erzeugten Ergebnisdatei durchgeführt. Die gegebenenfalls für die kategorialen Variablen verwendeten Formate werden zusätzlich in diese Ausgabetabelle geschrieben.
(ii) Die zugehörige Risikoschätzung erfolgt im zweiten Schritt in einer analytischen Prozedur, hier mit PROC LOGISTIC unter Verwendung des CLASS statements. Für die endgültige publikationsfähige Darstellung der Risikokennwerte und den dazugehörigen 95%-Konfidenzintervalle werden zwei von PROC LOGISTIC via SAS ODS erzeugte Ergebnisdateien in einfacher Weise nachbearbeitet.
(iii) Im dritten Schritt werden die beiden aufbereiteten beschreibenden und analytischen Ergebnisteile kombiniert.

Zur Illustration dienen die Daten einer häufigkeitsgematchten Fall-Kontroll-Studie. Unterschieden wird beim vorgestellten Ansatz nun bei den kategorialen Variablen nicht mehr zwischen binär und kategorial mit mehr als 2 Kategorien [1].

Beschreibende und analytische Prozeduren sind austauschbar, sodass sich ein allgemeiner Ansatz für andere Studienformen ableiten lässt. Die Konstruktion eines Makros lässt sich aufgrund des einfachen und insbesondere kurzen Programmcodes für eine bestimmte Auswertungsprozedur sehr leicht realisieren. Aufgrund der Änderungen in den Variablennamen der zur Aufbereitung der Ergebnisse verwendeten ODS OUTPUT Tabellen von SAS 9.1 auf SAS 9.2, wird das hier vorgestellte Makro in der aktuellen Version SAS 9.2 realisiert. Da die Schätzparameter abhängig von der analytischen Prozedur außerdem sowohl in unterschiedlich benannten, als auch unterschiedlich konstruierten ODS OUTPUT Tabellen ausgegeben werden, ist das hier gezeigte Vorgehen entweder für jede verwendete analytische Prozedur wiederum in wenigen Schritten anzupassen, oder das Makro entsprechend aufwendiger zu erweitern.

2.1 Voraussetzungen

Konkrete Voraussetzungen an die Daten erfordert das vorgestellte Verfahren natürlich in soweit, dass kategoriale Variablen vorhanden sein müssen. Stetige Variablen werden nur ins analytische Modell aufgenommen, jedoch nicht ausgezählt. Eine Gruppierung von stetigen Variablen per Format wäre als Erweiterung denkbar, ist jedoch derzeit nicht realisiert. Der Grund dafür liegt darin, dass bei der Anwendung der analytischen Prozedur der Einfachheit halber die CLASS Option DESCENDING eingesetzt wird. Im Falle von Formaten würde hier alphabetisch sortiert, was möglicherweise nicht in der Absicht des Anwenders liegt. Daher wird bei PROC FREQ und PROC LOGISTIC die Formatanweisung FORMAT _ALL_ verwendet.
Zur besseren Vergleichbarkeit mit den vorherigen Ergebnissen der KSFE 2008 [1] wird der Vorgang jedoch mit den gleichen drei Variablen (eine binäre, eine kategoriale und eine stetige Variable) vorgestellt. Die Kodierung spielt in keinem Falle eine Rolle. Die binäre Variable wird aus Gründen der Programmvereinfachung mit ins CLASS Modell geschrieben, was für die Ergebnisse jedoch irrelevant ist.

2.2 Beispiel

Das Anwendungsbeispiel verwendet Daten einer deutschen Fall-Kontrollstudie zum Kehlkopfkarzinom [2]. Die verwendeten Variablen dienen dabei der Beschreibung der Methodik und geben nicht die beste Modellanpassung wieder.

Variable	Name	Label	Kodierung	Format
Binär	JsR	Jahre seit Rauchende	0, 1	0-1=Raucher, 2+=Exraucher
Kategorial	pck4kat	Packungsjahre des Rauchens (PJ)	1, 2, 3, 4	1=0 2=0 < PJ < 20 3=20 <= PJ < 40 4=40 <= PJ < ..
Stetig	EthTag	Ethanol pro Tag	-	g Ethanol / 25

Der Aufruf eines Makros unter Verwendung von Dummy-Variablen anstelle eines CLASS Statements wird nur angedeutet, um die zusätzlichen Arbeitsschritte für diesen Analysefall zu illustrieren. Details, die sich im Makro-Fall nicht von der schrittweisen Darstellung ohne Makro unterscheiden, da sie sich nur auf die Verbindungstabelle zwischen Häufigkeitstabelle und Ergebnistabelle des analytischen Teils beziehen, können in der letztjährigen Publikation nachgelesen werden [1].

Zur Auswertung dieser häufigkeitsgematchten Fall-Kontroll-Studie mittels bedingter logistischer Regression wird die Prozedur PROC LOGISTIC verwendet. Details sind z.B. nachzulesen bei Hosmer & Lemeshow (1989) [3] bzw. in der SAS-Hilfe (SAS-Example 65.4: Conditional Logistic Regression for m:n Matching [4]).

Im Beispiel werden folgende weitere Variablen benötigt:

Fallkontrollstatus, *caco*, binär (0=Kontrollen, 1=Fälle)
Stratifizierungsvariable, *SexAgeGrp*, kategorial (Altersgruppen 1 bis n).

3 Deskriptive Statistik

Die einfachste Prozedur zur Angabe von absoluten und relativen Häufigkeiten ist sicherlich PROC FREQ, deren im OUTPUT Fenster dargestellten Ergebnisse sich in dieser Form nicht für die direkte Übernahme in einer Publikation eignen (Tabelle 1). Dies gilt in gleicher Weise, wenn statt PROC FREQ - PROC TABULATE verwendet würde.

Tabelle 1: Ausgabe von PROC FREQ der Variablen JsR im OUTPUT–Fenster

Table of JsR by caco

JsR (Jahre seit RauchEnde)	*Caco*		
Frequency *Col Pct*	*0*	*1*	*Total*
0	383 49.80	176 68.48	559
1	386 50.20	81 31.52	467
Total	769	257	1026

Daher wird per ODS OUTPUT eine Ausgabedatei für PROC FREQ erzeugt und das Ergebnis in Tabelle 2 angezeigt. Überflüssige Prozentangaben sind im folgenden Prozedurschritt weggelassen, da in einer Fall-Kontroll-Studie die Prozentangaben der Exposition in der jeweiligen Gruppe von Fällen bzw. Kontrollen von Interesse sind.

```
ods output CrossTabFreqs=ctf;
proc freq data=smoke;
tables (JsR pck4kat) * caco /nopercent norow;
format _all_;
run;
```

Mittels *ods output* werden Häufigkeiten und weitere Informationen aus PROC FREQ an die selbst benannte Ausgabedatei *ctf* weitergegeben. Die Informationen dieser Datei *ctf* sind in Tabelle 2 wiedergegeben. Automatisch werden in der Dabei ctf zusätzlich zu den Häufigkeiten und Prozentangaben die Variablen *Table*, *_TYPE_* und *missing* erzeugt. Aus Platzgründen ist in Tabelle 2 der vollständige Ergebnisteil nur für die Variable *JsR* dargestellt. Die Informationen bzgl. der Variablen *pck4kat* sind jedoch in weiteren Zeilen dieser Tabelle enthalten (in diesem konkreten Beispiel 15 Zeilen).
Die Variable *Table* enthält die Information der Variablen, von welcher die Häufigkeiten in der Zeile folgen. Die Textvariable *_TYPE_* zeigt an, ob es sich hier um innere Werte der Kreuztabelle handelt (_TYPE_ ="11"), bzw. die Randverteilungen bzgl. der Variablen JsR (_TYPE_ ="10") bzw. caco (_TYPE_ ="01"). Die Summe der eingegangenen Beobachtungen ist mit _TYPE_ ="00" gekennzeichnet. Dies lässt sich leicht anhand der beiden Tabellen 1 und 2 nachvollziehen. Die Variable (=Spalte) JsR enthält die Werte der Kodierung der Originalvariablen. Für jede zusätzliche Variable im TABLES Statement der Variablenliste von PROC FREQ ist eine weitere Spalte in der Tabelle ctf mit dem entsprechenden Wertebereich vorhanden. Bei Auftreten von fehlenden Werten ist diese Information in der Spalte *Missing* abgelegt.

```
proc print data=ctf;        /* Zur Ausgabe von Tabelle 2 */
```

```
run;
```

Tabelle 2: Von PROC FREQ erzeugte Ergebnistabelle ctf (Auszug)

Obs	Table	caco	_TYPE_	Frequency	ColPercent	JsR	pck4kat	Missing
1	Table JsR * caco	0	11	383	49.8049	0	.	.
2	Table JsR * caco	1	11	176	68.4825	0	.	.
3	Table JsR * caco	.	10	559	.	0	.	.
4	Table JsR * caco	0	11	386	50.1951	1	.	.
5	Table JsR * caco	1	11	81	31.5175	1	.	.
6	Table JsR * caco	.	10	467	.	1	.	.
7	Table JsR * caco	0	01	769	.	.	.	.
8	Table JsR * caco	1	01	257	.	.	.	.
9	Table JsR * caco	.	00	1026	.	.	.	0
10	Table pck4kat * caco	0	11	203	26.3979	.	1	.
11	Table pck4kat * caco	1	11	9	3.5019	.	1	.
12	Table pck4kat * caco	.	10	212	.	.	1	.
..	..	..	..	..	..	..	..	..
24								

Mit einem einfachen Datenschritt lassen sich nun 3 Aufgaben durchführen:

- Absolute und relative Häufigkeiten optisch ansprechend in einer neuen Textvariablen *NPct* kombinieren.
- Den Namen der original in PROC FREQ ausgewerteten Variablen mit der Funktion SCAN aus der Variablen *Table* extrahieren (Spalte: *OrigVar*).
- Die Werte der Variablen *JsR* und *pck4kat* mittels eines Arrays in eine einzige Wertespalte mit Namen *value* überführen.

Schritt 1:
Verwendung von PUT zur Formatzuweisung bei Erzeugung der Variablen *NPct* bewirkt eine Dezimalstelle für jede Prozentangabe, auch im Falle von ganzen Zahlen. Häufigkeiten und formatierte Prozentangaben werden nun noch mittels Klammern und führendem Leerzeichen kombiniert.

Schritt 2 + 3:
Die Variablen *OrigVar* und *Value* (Schritt 3) werden später bei der Zusammenführung der deskriptiven und der analytischen Variablen benötigt.

```
data ctf1;
```

```
      set ctf;
      /* Schritt 1: */
      NPct=Frequency||" ("||PUT(ColPercent,4.1)||")";

      /* Schritt 2: */
      OrigVar=scan(Table, 2);

      /* Schritt 3: */
      array ArrVars JsR pck4kat;
      do over ArrVars;
         if ArrVars ne . then Value=put(ArrVars,1.);
      end;    *Ende des do-over-Array-Schrittes;
run;
```

Tabelle 3 zeigt die Ausgabe PROC PRINT nur der inneren Werten der Kreuztabelle JSR*caco (d.h. _TYPE_="11") und den neu konstruierten Variablen *NPct*, *OrigVar* und *Value*. Zur besseren Übersicht sind insbesondere die Spalten *Frequency* und *ColPercent* nicht ausgegeben.

Tabelle 3: Ergebnis der aufbereiteten Tabelle ctf1.

Obs	Table	caco	NPCT	JsR	pck4kat	OrigVar	value
1	Table JsR * caco	0	383 (49.8)	0	.	JsR	0
2	Table JsR * caco	1	176 (68.5)	0	.	JsR	0
4	Table JsR * caco	0	386 (50.2)	1	.	JsR	1
5	Table JsR * caco	1	81 (31.5)	1	.	JsR	1
10	Table pck4kat * caco	0	203 (26.4)	.	1	pck4kat	1
11	Table pck4kat * caco	1	9 (3.5)	.	1	pck4kat	1
13	Table pck4kat * caco	0	297 (38.6)	.	2	pck4kat	2
14	Table pck4kat * caco	1	33 (12.8)	.	2	pck4kat	2
16	Table pck4kat * caco	0	147 (19.1)	.	3	pck4kat	3
17	Table pck4kat * caco	1	88 (34.2)	.	3	pck4kat	3
19	Table pck4kat * caco	0	122 (15.9)	.	4	pck4kat	4
20	Table pck4kat * caco	1	127 (49.4)	.	4	pck4kat	4

Das Ziel einer Darstellung der Häufigkeiten von Fällen und Kontrollen in 2 Spalten erfordert die Anwendung von PROC TRANSPOSE auf die inneren Werte von Tabelle ctf1. Transponiert wird mit den eindeutig identifizierenden Variablen *OrigVar* und *Value*. Mittels p*refix=caco* und *id caco* werden die neuen beiden Spalten *caco0* und

caco1 benannt, die später mit den LABELn „Kontrollen“ und „Fälle“ bezeichnet werden könnten. Die Ausgabe erfolgt in Datei ctf2 (siehe Tabelle 4, linke Seite).

```
proc transpose data=ctf1 out=ctf2 prefix=caco;
     var NPct;
     id caco;
     by OrigVar Value;
     where _TYPE_="11";
run;
```

Tabelle 4: Von PROC TRANSPOSE erzeugte Ergebnisdatei ctf2.

Obs	OrigVar	Value	caco0	caco1	which	Choose	FmtValue
1	JsR	0	383 (49.8)	176 (68.5)	1	fJsR.	Quit<2Jahre
2	JsR	1	386 (50.2)	81 (31.5)	1	fJsR.	Quit>=2Jahre
3	pck4kat	1	203 (26.4)	9 (3.5)	2	fpck4kat.	Nieraucher
4	pck4kat	2	297 (38.6)	33 (12.8)	2	fpck4kat.	0<PJ.<20
5	pck4kat	3	147 (19.1)	88 (34.2)	2	fpck4kat.	20<=PJ.<40
6	pck4kat	4	122 (15.9)	127 (49.4)	2	fpck4kat.	40<=PJ.

In Tabelle 4 sind die Kategorien (*Value*) aufgrund von fehlenden Formatangaben nur schwer lesbar. Es gibt sicher verschiedene Möglichkeiten vorhandene Formate in diese Tabelle zu schreiben. Die hier vorgestellte Lösung verwendet die Funktionen „whichc“, „choosec“ und „putn“. Der Sinn des Datenschrittes besteht allein darin, den Variablen *JsR* und *pck4kat* ihre jeweiligen Formate (hier: *fJsR*. und *fpck4kat*. genannt) zuzuordnen. Das Ergebnis ist auf der rechten Seite in Tabelle 4 dargestellt.

```
data ctf2;
  set ctf2;
    which = whichc(OrigVar, JsR pck4kat );
    choose = choosec(which, fJsR. fpck4kat.);
    fmtValue = putn(input(value,best.),choose);
  run;
```

Somit ist Teil 1 der gestellten Aufgabe, die Kombination von absoluten und relativen Häufigkeiten und die Ausgabe in einer weiter verarbeitbaren Datei, erfüllt.

4 Analytische Statistik

Der zweite Teil der gestellten Aufgabe besteht nun in der Aufbereitung der analytischen Ergebnisse. Im Vergleich zur PROC PHREG [1] stehen bei PROC LOGISTIC die Na-

men der Variablen und p-Werte, bzw. Odds Ratios und Konfidenzparameter in 2 verschiedenen ODS OUTPUT Tabellen.

Als Modellvariablen für die PROC LOGISTIC werden nun die binäre Variable *JsR*, die kategoriale Variable *pck4kat* und die stetige Variable *EthTag* ins Modell aufgenommen. Damit die Nichtraucher (*pck4kat*=1) auch als Referenz ausgewiesen sind, wird hier die Option DESCENDING des CLASS Statements und die Formatanweisung FORMAT_all_ verwendet, da bei vorhandenen Formaten in den Datensätzen die bzgl. des Formates "kleinste" alphabetische Kategorie als Referenzgruppe herangezogen wird.
Vorhandene Formate werden im Makro also "ausgeschaltet". Das auf der CD zusätzlich vorhandene Makro "2_NPct_Logist.sas" gibt aber zusätzlich die Möglichkeit, Formate im Ergebnis mit auszugeben.
Die Variable *SexAgeGr* enthält die Gruppierung in 5-Jahres-Alters-und-Geschlechtsgruppen entsprechend den Matching-Kriterien der Fall-Kontroll-Studie. Analog Kapitel 3 werden die Ergebnisteile von PROC LOGISTIC, welche die Namen, p-Werte und weitere Schätzer enthalten (*ParameterEstimates*) bzw. Odds Ratios und Konfidenzpatrameter (*OddsRatios*) mittels *ods output* an 2 selbst benannte Ausgabedateien mit Namen *ParmEst* und *ORs* weitergegeben.

```
ods output ParameterEstimates=ParmEst;
ods output OddsRatios=ORs;
proc logistic data = Smoke;
class JsR pck4kat; /* JSR eigentlich hier nicht benötigt */
model caco (event="1")= JsR pck4kat EthTag / risklimits PARAM=REF;
strata SexAgeGr;
run;
```

Die Ausgabe des von PROC LOGISTIC für das weitere Vorgehen wichtigen Inhaltes ist in den Tabellen 5 und 6 wiedergegeben.

Auf eine Interpretation der Ergebnisse wird hier bewusst verzichtet.

Tabelle 5: Ausgabe der von PROC LOGISTIC erzeugten Datei *ParmeterEstimates*

Variable	Class Val0	DF	Estimate	StdErr	Wald ChiSq	Prob ChiSq
JsR	1	1	-1.04362	0.19108	29.8314	<.0001
pck4kat	4	1	3.44666	0.38944	78.3283	<.0001
pck4kat	3	1	3.13383	0.39288	63.6265	<.0001
pck4kat	2	1	1.79274	0.41733	18.4535	<.0001
EthTag		1	0.00560	0.00133	17.6387	<.0001

Tabelle 6: Ausgabe der von PROC LOGISTIC erzeugten Datei *ORs* für *(OddsRatios)*

Variable	OddsRatioEst	LowerCL	UpperCL		OR	CI
JsR 1 vs 0	0.352	0.242	0.512		0.4	(0.2,0.5)
pck4kat 4 vs 1	31.395	14.634	67.353		6.0	(2.7,13.6)
pck4kat 3 vs 1	22.962	10.631	49.593	>>>	23.0	(10.6,49.6)
pck4kat 2 vs 1	6.006	2.651	13.608		31.4	(14.6,67.4)
EthTag	1.150	1.077	1.228		1.2	(1.1,1.2)

Die Ausgabe der Odds Ratios und 95%-Konfidenzintervalle der Prozedur LOGISTIC wird mit einfachen Mitteln durch Aufbereitung der Datei *ORs* optisch verbessert und in der Ergebnisdatei *Ors1* ausgegeben. Dieser Schritt wird hier in den letzten beiden Spalten von Tabelle 6 dargestellt.

```
data ORs1;
set ORs;
OR=PUT(OddsRatioEst, 5.1);
CI=CATS("(",PUT(LowerCL,5.1))||", "||CATS(PUT(UpperCL,5.1),")");
run;
```

Mit der PUT-Funktion lässt sich ein Format mit einer Dezimalstelle erreichen (Der Unterschied gegenüber ROUND besteht darin, dass auch ganzzahlige Ergebnisse wie 6.0 auf 1 Dezimalstelle dargestellt werden, und nicht als „6“.). Die Funktion CATS fügt die Klammer und die untere Konfidenzgrenze ohne Leerstellen zusammen (analog die obere Konfidenzgrenze und die schließende Klammer). Dazwischen wird nun noch ein Komma mit Leerzeichen als optisches Trennzeichen eingefügt.

Der Unterschied zwischen dem obigen einfachen Programmcode zur Aufbereitung der Odds Ratios (OR) und den 95%-Konfidenzintervallen in der Datei *ORs1* und dem in Tabelle 7 dargestellten Endergebnis besteht darin, das für die schönere Formatierung in Tabelle 7 noch einige wenige extra Programmzeilen nötig sind. Diese tragen jedoch nur der Konvention Rechnung, Dezimalzahlen größer als 1 mit nur 1 Dezimalstelle darzustellen, Dezimalzahlen kleiner als 1 dagegen mit 2 Dezimalstellen. Die Zeile zur optischen Verbesserung der Odds Ratios (OR) wird in diesem Falle wie folgt ersetzt:

```
if OddsRatioEst <1 then OR=PUT(OddsRatioEst,5.2);
   else OR=PUT(OddsRatioEst,5.1);
```

Die Zeile zur optischen Verbesserung der Konfidenzintevalle (CI) wird wie folgt ersetzt (abkürzend sind hier LCL bzw. UCL anstelle von LowerCL bzw. UpperCL verwendet):

```
select;
when(LCL<1 and UCL<1)
          CI= CATS("(",PUT(LCL,5.2))||", "|| CATS(PUT(UCL,5.2),")");
when(LCL<1 and UCL>=1)
          CI= CATS("(",PUT(LCL,5.2))||", "|| CATS(PUT(UCL,5.1),")");
when(LCL>=1 and UCL>=1)
          CI= CATS("(",PUT(LCL,5.1))||", "|| CATS(PUT(UCL,5.1),")");
otherwise;
end;
```

Somit ist Teil 2 der gestellten Aufgabe, die Aufbereitung und Ausgabe der Risikokennwerte in einer weiter verarbeitbaren Datei, erfüllt.

5 Kombination der Ergebnisse deskriptiver und analytischer Statistik

In einem einzigen SQL Schritt lassen sich nun die beiden Ergebnisteile aus Kapitel 3 und 4 kombinieren. Ausgangspunkte sind die oben konstruierten Tabellen 4 mit den Häufigkeitsverteilungen Tabelle 5 (Risikokennwerte, Teil 1) und Tabelle 6 (Risikokennwerte, Teil 2: OR, 95%-CI). Da die Merge-Variablen in Tabelle 4 (*OrigVar, Value*) und 5 (*Variable, ClassVal0*) unterschiedliche Namen haben, bietet sich hier PROC SQL zur Kombination beider Tabellen an. Im gleichen SQL-Schritt werden außerdem Tabelle 5 und Tabelle 6 kombiniert (innere select-Anweisung in Klammern). Tabelle 7 enthält zusätzlich die stetige Variable EthTag aus dem analytischen Modell.

Als Hilfe bei evtl. nötigen Sortierungen lässt sich in Tabelle 4 zusätzlich eine Sortiervariable MyOrder konstruieren, die später eine Darstellung von genau der Reihenfolge der Beobachtungen zulässt, wie sie ursprünglich im TABLES Statement der PROC FREQ eingegeben wurden.

```
proc sql;
    create table ResultsSmoke as
    select MyOrder, f.OrigVar, f.Value, FmtValue, Effect,
          caco0, caco1, OR, CI format $12., ProbChisq
    from ctf3 as f full join
    (select p.Variable as Variable, o.Variable as oVar, p.ClassVa10,
          ProbChisq, o.Effect, o.or, o.ci
    from ParmEst as p left join ORs1 as o
    on p.Variable=o.Variable and p.ClassVa10=o.ClassVa10) as po
    on (f.OrigVar=po.variable and f.value=po.ClassVal0)
    order by MyOrder;
quit;
```

Zur optischen Verfeinerung lassen sich mit Labels die Variablen caco0 (=Kontrollen), caco1 (=Fälle) und ci (=95%CI) natürlich noch beschriften. Per Programm kann in der Referenzkategorie zusätzlich für OR der Wert 1 und ci="-„ eingetragen werden.

Tabelle 7: Ergebnisdatei, kombiniert aus den Tabellen 4, Tabelle 5 und Tabelle 6

Obs	Variable	Value	FmtValue	caco0	caco1	OR	CI
1	JsR	0	Ex-Raucher	383 (49.8)	383 (49.8)		
2	JsR	1	Raucher	386 (50.2)	81 (31.5)	0.35	(0.24, 0.51)
6	pck4kat	1	Nie-Raucher	203 (26.4)	9 (3.5)		
5	pck4kat	2	0 < PJ < 20	297 (38.6)	33 (12.8)	6.0	(2.7, 13.6)
4	pck4kat	3	20 <= PJ < 40	147 (19.1)	88 (34.2)	23.0	(10.6, 49.6)
3	pck4kat	4	40 <= PJ	122 (15.9)	127 (49.4)	31.4	(14.6, 67.4)
7	EthTag					1.2	(1.1, 1.2)

6 Schlussfolgerung

Das oben gezeigte Verfahren orientiert sich an den Ergebnisdateien der deskriptiven und der analytischen Prozeduren. Unter Verwendung des CLASS Statements ist auch die Kombination der beiden Ergebnisteile (deskriptiv und analytisch) einfacher als in der ohne CLASS durchgeführten Variante (siehe [1]). Bei Verwendung anderer analytischer Prozeduren ist es daher leicht möglich das Verfahren unter Beachtung der dort verwendeten Ausgabedateien und insbesondere der Benennungen der darin verwendeten Variablen anzupassen. Das Verfahren ist somit leicht auf andere Auswertungssituationen in klinischen und epidemiologischen Studien anwendbar. Da keine der Werte innerhalb der Tabelle im Textverarbeitungsprogramm nachbearbeitet wurden, lässt sich diese Tabelle mit einem einfachen Programmdurchlauf wiederholen, sollten sich Änderungen in den Daten oder im analytischen Modell ergeben.

7 Schlussbemerkung

Kleiner Exkurs für ein Makro ohne Class-Statement:

Eine kategoriale Variable (zum Beispiel pck4kat; kategorisiert in 4 Gruppen und beschrieben durch 4 Dummy-Variablen) könnte wie folgt im Makro-Aufruf erscheinen:

```
BinVars = pck4kat1* pck4kat2* pck4kat3* pck4kat4 JsR
```

(erforderlich sind unterschiedliche Trennzeichen für „zusammengehörende" Variablen)

Für Proc Freq würden dann die folgenden Variablen ausgezählt:

```
FreqVars= pck4kat1 pck4kat2 pck4kat3 pck4kat4 JsR
```

Für Proc Logistic wird die Referenzgruppe entfernt, und die stetigen Variablen ergänzt:

```
ModelVars= pck4kat2 pck4kat3 pck4kat4 JsR
```

Dass dies komplizierter in der Verarbeitung innerhalb des Makros ist, lässt sich im KSFE-Beitrag 2008 nachvollziehen [1].

Literatur

[1] Ramroth H (2008): Publikationsfertige Kombination von Häufigkeiten und Risiko-Kennwerten aus Ergebnissen von klinisch-epidemiologischen Studien; Proceedings der 12. KSFE; Aachen.

[2] Ramroth H, Dietz A, Becher H. Interaction effects and population-attributable risks for smoking and alcohol on laryngeal cancer and its subsites. Methods Inf Med 2004; 43 (5), 499-504

[3] Hosmer DWJ & Lemeshow S (1989) *Applied Logistic Regression*, New York: John Wiley & Sons, Inc.

[4] SAS-Example 65.4: Conditional Logistic Regression for m:n Matching.