

Schritt für Schritt-Einführung in die Prozedur Tabulate

Carina Ortseifen
Universitätsrechenzentrum Heidelberg
Im Neuenheimer Feld 293
69120 Heidelberg
Carina.Ortseifen@urz.uni-heidelberg.de

Zusammenfassung

Mit der Prozedur Tabulate kann man nicht nur ansprechende Tabellen – im Vergleich zur Prozedur Print - erzeugen, sondern neben den Spaltensummen weitere statistische Kenngrößen wie Mittelwert, Summe, Minimum und Maximum ermitteln. Vor dem Einsatz dieser Prozedur schreckt der Anfänger dann aber häufig wegen der ungewöhnlichen Syntax zurück. Die Leser dieses Tutoriums werden in die Verwendung der Prozedur Schritt für Schritt eingeführt und lernen die wesentlichen Anweisungen und Optionen kennen, um ein- bis mehrdimensionale Tabellen mitsamt ihrer statistischen Kennwerte ansprechend darzustellen.

Schlüsselworte: Prozedur Tabulate, Tutorium, Einführung, Table, Var, Class, Condense, Format, Box, Statistik, ODS HTML, ODS RTF, MS Excel.

1. Einleitung

Die Prozedur Tabulate kann statistische Kennwerte berechnen und diese in Tabellenform darstellen. Während die Berechnung der Kennwerte auch von anderen SAS-Prozeduren durchgeführt werden kann, z.B. Means und Univariate, ist die tabellarische Darstellung auch mit dem ODS (Output Delivery System) nicht überflüssig geworden, wie später gezeigt werden wird.

Ein Grund, warum die Prozedur Tabulate trotz ihrer zahlreichen Fähigkeiten nicht überall eingesetzt wird, liegt sicherlich in ihrer schwer zugänglichen Syntax verborgen. Selbst wenn man zahlreiche SAS-Prozeduren kennt, lässt sich deren Logik nicht direkt auf die Prozedur Tabulate übertragen.

C. Ortseifen

Diese Schritt für Schritt-Einführung, basierend auf dem SUGI27-Tutorial von Lauren Haworth [1], schafft da - hoffentlich - Abhilfe. Nach Vorstellung der den nachfolgenden Programmschritten zugrunde liegenden Beispieletabelle Prdsale.sas7bdat werden zunächst eindimensionale Tabellen erzeugt, die aus einer Zeile bzw. Spalte bestehen. Darauf aufbauend zweidimensionale und danach dreidimensionale Tabellen. Anschließend werden die Tabellen verschönt, u.a. durch Reduktion redundanter Beschriftungen und Exportmöglichkeiten als HTML-File bzw. Excel-Tabelle vorgestellt.

1.1 Die SAS-Tabelle Prdsale.sas7bdat

Alle im weiteren Verlauf vorgestellten Programmschritte greifen auf die SAS-Tabelle Prdsale.sas7bdat zu, die von SAS Institute mit der SAS Software ausgeliefert wird.

Die Tabelle Prdsale.sas7bdat enthält Verkaufszahlen (Variablen Actual und Predict) von Möbeln (Prodtype, Product) für verschiedene Produktgruppen (Division) für die drei Länder (Country) Deutschland, USA und Kanada, unterteilt in zwei Regionen (Region) Ost und West und für die Jahre 1993 und 1994 (Month, Quart, Year).

	Actual Sales	Predicted Sales	Country	Region	Division	Product type	Product	Quarter	Year	Month
1	\$925.00	\$950.00	CANADA	EAST	EDUCATION	FURNITURE	SOFA	1	1993	Jan
2	\$999.00	\$297.00	CANADA	EAST	EDUCATION	FURNITURE	SOFA	1	1993	Feb
3	\$608.00	\$946.00	CANADA	EAST	EDUCATION	FURNITURE	SOFA	1	1993	Mar
4	\$642.00	\$533.00	CANADA	EAST	EDUCATION	FURNITURE	SOFA	2	1993	Apr
5	\$656.00	\$646.00	CANADA	EAST	EDUCATION	FURNITURE	SOFA	2	1993	May
6	\$949.00	\$486.00	CANADA	EAST	EDUCATION	FURNITURE	SOFA	2	1993	Jun
7	\$612.00	\$717.00	CANADA	EAST	EDUCATION	FURNITURE	SOFA	3	1993	Jul
8	\$114.00	\$564.00	CANADA	EAST	EDUCATION	FURNITURE	SOFA	3	1993	Aug
9	\$695.00	\$230.00	CANADA	EAST	EDUCATION	FURNITURE	SOFA	3	1993	Sep
10	\$657.00	\$494.00	CANADA	EAST	EDUCATION	FURNITURE	SOFA	4	1993	Oct
11	\$608.00	\$903.00	CANADA	EAST	EDUCATION	FURNITURE	SOFA	4	1993	Nov
12	\$353.00	\$266.00	CANADA	EAST	EDUCATION	FURNITURE	SOFA	4	1993	Dec
13	\$107.00	\$190.00	CANADA	EAST	EDUCATION	FURNITURE	SOFA	1	1994	Jan
14	\$354.00	\$139.00	CANADA	EAST	EDUCATION	FURNITURE	SOFA	1	1994	Feb
15	\$101.00	\$217.00	CANADA	EAST	EDUCATION	FURNITURE	SOFA	1	1994	Mar
16	\$553.00	\$560.00	CANADA	EAST	EDUCATION	FURNITURE	SOFA	2	1994	Apr
17	\$877.00	\$148.00	CANADA	EAST	EDUCATION	FURNITURE	SOFA	2	1994	May
18	\$431.00	\$762.00	CANADA	EAST	EDUCATION	FURNITURE	SOFA	2	1994	Jun
19	\$511.00	\$457.00	CANADA	EAST	EDUCATION	FURNITURE	SOFA	3	1994	Jul
20	\$157.00	\$532.00	CANADA	EAST	EDUCATION	FURNITURE	SOFA	3	1994	Aug
21	\$520.00	\$629.00	CANADA	EAST	EDUCATION	FURNITURE	SOFA	3	1994	Sep

2. Eindimensionale Tabellen

Der Proc Tabulate Schritt benötigt drei Anweisungen:

```
Proc Tabulate Data=d.prdsale;  
  Var actual;  
  Table actual;  
Run;
```

Die drei Anweisungen lauten: Proc Tabulate, Var und Table. Anstelle der Anweisung Var kann auch die Anweisung Class erscheinen, worauf aber erst in Abschnitt 2.3 eingegangen wird.

Die Anweisung

```
Proc Tabulate Data=d.prdsale
```

benennt über die Option Data= die zu verwendende SAS-Tabelle.

Die Anweisung

```
Var actual;
```

definiert die Analysevariable, die für die Berechnung der Statistik verwendet wird. Man könnte auch sagen, dass mit dieser Anweisung der Variable actual ihre Rolle als Analysevariable zugewiesen wird.

Die Anweisung

```
Table actual;
```

benennt die Variablen, die tabellarisch dargestellt werden, hier nur eine Variable, actual.

Würde eine der beiden Anweisungen Var oder Table fehlen, erschiene sofort eine der folgenden Fehlermeldungen:

C. Ortseifen

```
ERROR: No table statement has been specified.  
ERROR: The type of name (actual) is unknown.
```

Führt man den Programmschritt

```
Proc Tabulate Data=d.prdsale;  
  Var actual;  
  Table actual;  
Run;
```

nun aus, erscheint eine einspaltige Tabelle:

Actual Sales
Sum
730337.00

Die Kopfzeile enthält die Überschrift, Actual Sales, dann folgt die Statistik, Sum, und anschließend die Summe der aktuellen Verkaufszahlen: 730337.00.

2.1 Ändern der Statistik

In dem ersten Beispiel wird – ob sinnvoll oder nicht sei dahingestellt – die Summe der aktuellen Verkaufszahlen berechnet. Summenbildung ist somit der Standard (oder Default), wenn keine Kenngröße benannt wird.

Um eine andere Kenngröße der Analysevariablen zu ermitteln, muss die Anweisung Table erweitert werden um den Tabulate-Operator * und den Namen der Statistik, die berechnet werden soll:

```
Proc Tabulate Data=d.prdsale;  
  Var actual;  
  Table actual*Mean;  
Run;
```

Das Ergebnis hat dann folgende Form:

Actual Sales
Mean
507.18

Die Summenstatistik wird durch den Mittelwert (Mean) ersetzt und als errechneten Wert erhält man die mittlere aktuelle Verkaufszahl, 507,18.

2.2 Hinzufügen einer weiteren Statistik

Möchte man zusätzlich zum Mittelwert noch die Anzahl der Beobachtungen ausgeben, ergänzt man die Anweisung Table. Dazu stehen zwei Möglichkeiten zur Verfügung.

```
* Variante A ;
Proc Tabulate Data=d.prdsale;
  Var actual;
  Table actual*N actual*Mean;
Run;
```

Fügt man die zweite Anforderung actual*N ein, erhält man eine Tabelle mit einer weiteren Spalte mit drei Zeilen:

Actual Sales	Actual Sales
N	Mean
1440.00	507.18

Alternativ dazu kann man die angeforderten Statistiken in einer Klammer an den Tabulate-Operator * anfügen:

C. Ortseifen

```
* Variante B;  
Proc Tabulate Data=d.prdsale;  
  Var actual;  
  Table actual*(N Mean);  
Run;
```

Damit erhält man die etwas ansprechendere Form: Unter der gemeinsamen Überschrift Actual Sales erscheinen zwei Spalten à zwei Zeilen für die Kennwerte und die ermittelten Zahlen.

Actual Sales	
N	Mean
1440.00	507.18

2.3 Hinzufügen einer Klassifizierungsvariablen

Wurden bislang die Kennwerte für alle Beobachtungen ermittelt, soll im folgenden eine Klassifizierungsvariable eingefügt werden, damit die Kennwerte für einzelne Gruppen berechnet werden.

Dazu verwendet man analog zur Prozedur Means die Anweisung Class. Diese Anweisung weist die Klassifizierungsvariable aus. In der Anweisung Table legt man dann mit Hilfe des Tabulate-Operators * zusätzlich fest, an welcher Stelle die Untergruppen dargestellt werden.

Im folgenden Beispiel werden die mittleren Verkaufszahlen getrennt für die drei Länder ermittelt. Dazu wird die Variable country als Class-Variable definiert.

```
Proc Tabulate Data=d.prdsale;  
  Class country;  
  Var actual;  
  Table actual*Mean*country;  
Run;
```

Actual Sales		
Mean		
Country		
CANADA	GERMANY	U.S.A.
514.56	512.50	494.48

Die Aufteilung der Länder erfolgte nach der Zeile Mean.
Wie sieht wohl das Ergebnis von folgendem Beispielprogramm aus?

```
Proc Tabulate Data=d.prdsale;
  Class country;
  Var actual;
  Table country*actual*Mean;
Run;
```

3. Zweidimensionale Tabellen

Alle bisher erzeugten Tabellen wiesen nur eine Dimension auf. Interessanter wird das ganze natürlich, wenn eine zweite Dimension ins Spiel kommt, d.h. die Tabelle aus mehreren Spalten und Zeilen besteht, wie das der SAS-Anwender etwa von der Prozedur Freq kennt.

Die zweite Dimension wird im Prozedur Tabulate-Schritt definiert, indem die Variable, die in der zweiten Dimension eingetragen werden soll, in der Anweisung Table durch Komma von der Variablen der ersten Dimension abgetrennt wird. Die zweite Variable, im folgenden Beispiel prodtype, muss dazu in einer Class-Anweisung als Klassifizierungsvariable definiert werden.

```
Proc Tabulate Data=d.prdsale;
  Class prodtype;
  Var actual;
  Table prodtype, actual*(N Mean);
Run;
```

C. Ortseifen

	Actual Sales	
	N	Mean
Product type		
FURNITURE	576.00	504.56
OFFICE	864.00	508.93

Getrennt für die Produkttypen Furniture und Office werden die Beobachtungszahlen und die mittleren Verkaufszahlen ausgegeben. Im Unterschied zum vorherigen Beispiel aus Abschnitt 2.3 werden die Produkttypen als zweite Dimension eingeführt, so dass eine zweidimensionale Tabelle mit den Produkttypen als Zeilen und den Kenngrößen der Verkaufszahlen als Spalten erscheint.

Wird die Klassifizierungsvariable `prodtype` wie hier links von der Analysevariable `actual` aufgeführt, erscheint sie in den Zeilen. Stünde sie dagegen rechts von der Analysevariable,

```
Table actual*(N Mean), prodtype;
```

würde sie in den Spalten aufgetragen.

Anmerkung

Eine zweidimensionale Tabelle muss mindestens eine Klassifizierungsvariable enthalten. Oder, anders formuliert, die Analysevariablen dürfen nur in einer Dimension auftauchen. Andernfalls erscheint eine Fehlermeldung, wie das folgende Beispiel mit zwei Analysevariablen zeigt.

Während der Prozedurschritt

```
Proc Tabulate Data=d.prdsale;  
  Class prodtype;  
  Var actual predict;  
  Table prodtype, (actual predict)*(N Mean);  
Run;
```


folgendes Ergebnis erzeugt:

	Actual Sales		Predicted Sales	
	N	Mean	N	Mean
Product type				
FURNITURE	576.00	504.56	576.00	483.19
OFFICE	864.00	508.93	864.00	495.34

erhält man mit

```
Proc Tabulate Data=d.prdsale;
  Class prodtype;
  Var actual predict;
  Table actual*prodtype, (predict)*(N Mean);
Run;
```

folgende Fehlermeldungen:

```
ERROR: There are multiple analysis variables associated with
a single table cell in the following nesting : ACTUAL *
PRODTYPE * PREDICT * N.
```

```
ERROR: There are multiple analysis variables associated with
a single table cell in the following nesting : ACTUAL *
PRODTYPE * PREDICT * Mean.
```

Alle bislang vorgestellten Darstellungsmöglichkeiten können auch mit der Prozedur Means erzeugt werden. Dafür lohnt der Aufwand des Erlernens der Prozedur Tabulate wahrlich nicht – mögen Sie jetzt vielleicht denken. Doch die nun folgende Klassifizierung in zwei Dimensionen geht einen Schritt weiter:

3.1 Klassifizierung in zwei Dimensionen

Der nachfolgende Prozedur Tabulate-Schritt definiert zwei Klassifizierungsvariablen und setzt diese beiden in verschiedene Dimensionen mit Hilfe der Anweisung Table ein:

```
Proc Tabulate Data=d.prdsale;  
  Class country prodtype;  
  Var actual;  
  Table prodtype,actual*country*Mean;  
Run;
```

Als Ergebnis erhalten wir die mittleren Verkaufszahlen getrennt nach Möbeltypen und getrennt nach Ländern:

	Actual Sales		
	Country		
	CANADA	GERMANY	U.S.A.
	Mean	Mean	Mean
Product type			
FURNITURE	509.71	527.05	476.91
OFFICE	517.80	502.79	506.19

Wird nun noch eine dritte Klassifizierungsvariable hinzugefügt, hat man wieder zwei mögliche Darstellungsformen: Entweder werden die Kennzahlen für beide Klassifizierungsvariablen unabhängig voneinander berechnet.

```
Proc Tabulate Data=d.prdsale;
  Class country prodtype year;
  Var actual;
  Table prodtype year,actual*country*Mean;
Run;
```

	Actual Sales		
	Country		
	CANADA	GERMANY	U.S.A.
	Mean	Mean	Mean
Product type			
FURNITURE	509.71	527.05	476.91
OFFICE	517.80	502.79	506.19
Year			
1993	504.25	530.85	504.39
1994	524.88	494.14	484.57

Im Prinzip wurden hier zwei Tabellen untereinander gehängt.

Oder die Kennwerte werden verschachtelt berechnet, wenn man sich z.B. dafür interessiert, wie sich die Verkaufszahlen der beiden Produkttypen mit den Jahren verändern.

```
Proc Tabulate Data=d.prdsale;
  Class country prodtype year;
  Var actual;
  Table prodtype*year,actual*country*Mean;
Run;
```

C. Ortseifen

		Actual Sales		
		Country		
		CANADA	GERMANY	U.S.A.
		Mean	Mean	Mean
Product type	Year			
FURNITURE	1993	505.88	565.06	480.05
	1994	513.54	489.04	473.77
OFFICE	1993	503.17	508.04	520.61
	1994	532.43	497.54	491.76

3.2 Spalten- und Zeilensummen

Komplexe Tabellen werden verständlicher, wenn neben den einzelnen Summen auch Zeilen- und Spaltensummen angegeben werden, wie der SAS-Anwender dies vermutlich von der Prozedur Freq kennt.

Im Prozedur Tabulate-Schritt verwendet man hierzu die interne Klassifizierungsvariable All.

Das folgende Beispiel legt mittels All eine weitere Spalte überschrieben mit All an, welche die Zeilensummen der Beobachtungszahlen enthält.

```
Proc Tabulate Data=d.prdsale;
  Class country prodtype;
  Var actual;
  Table prodtype,(country All)*actual*N;
Run;
```

	Country			All
	CANADA	GERMANY	U.S.A.	
	Actual Sales	Actual Sales	Actual Sales	Actual Sales
	N	N	N	N
Product type				
FURNITURE	192.00	192.00	192.00	576.00
OFFICE	288.00	288.00	288.00	864.00

Erscheint das Wörtchen All dagegen vor dem Komma, werden die Spaltensummen berechnet:

```
Proc Tabulate Data=d.prdsale;
  Class country prodtype;
  Var actual;
  Table prodtype All,country*actual*N; Run;
```

	Country		
	CANADA	GERMANY	U.S.A.
	Actual Sales	Actual Sales	Actual Sales
	N	N	N
Product type			
FURNITURE	192.00	192.00	192.00
OFFICE	288.00	288.00	288.00
All	480.00	480.00	480.00

C. Ortseifen

Und verwendet man All sowohl rechts als auch links vom Komma, werden Spalten- und Zeilensummen verwendet:

```
Proc Tabulate Data=d.prdsale;  
  Class country prodtype;  
  Var actual;  
  Table prodtype All,(country All)*actual*N;  
Run;
```

	Country			All	
	CANADA	GERMANY	U.S.A.		
	Actual Sales	Actual Sales	Actual Sales		Actual Sales
	N	N	N		N
Product type					
FURNITURE	192.00	192.00	192.00	576.00	
OFFICE	288.00	288.00	288.00	864.00	
All	480.00	480.00	480.00	1440.00	

3.3 Sonderfall: Einzelwerte darstellen

Möchte man mit der Prozedur Tabulate anstelle der summarischen Kenngrößen Einzelwerte darstellen, benötigt man eine Identifikationsvariable, die für jede Beobachtung einen eindeutigen Wert aufweist. (Falls solche eine Variable nicht vorliegt, kann man diese leicht mit Hilfe der internen Variable `_N_` des Datenschnitts anlegen:

```
Data d.prdsale;  
  Set d.prdsale;  
  id=_N_;  
Run;
```

Diese Identifikationsvariable, hier `id`, wird anschließend im Prozedur `Tabulate`-Schritt als Klassifizierungsvariable verwendet:

```
Proc Tabulate Data=d.prdsale;  
  Var actual;  
  Class id;  
  Table id,actual*Sum;  
Run;
```

und man erhält folgende Tabelle:

	Actual Sales
	Sum
Id	
1	925.00
2	999.00
3	608.00
...	
1438	526.00
1439	652.00
1440	573.00

Die Bezeichnung `Sum` ist natürlich etwas irreführend. Wie diese unterdrückt werden kann, wird im Abschnitt 5.1 beschrieben.

4. Dreidimensionale Tabellen

Der Übergang von zwei- zu dreidimensionalen Tabellen fällt nun nicht mehr schwer.

Da dreidimensionale Zahlenwürfel jedoch schlecht dargestellt bzw. zu Papier gebracht werden können, werden höherdimensionale Tabellen in Form von mehrseitigen zweidimensionalen Tabellen dargestellt.

Wie bereits im vorherigen Abschnitt deutlich wurde, entspricht die erste Dimension den Spalten, die zweite Dimension den Zeilen und die dritte Dimension folglich den Seiten, wobei die angegebene Zählung von rechts nach links erfolgt. (Auch dies ist analog zur Prozedur Freq.)

Soll nun eine dreidimensionale Tabelle erzeugt werden, legt man zunächst die zweidimensionale Tabelle fest und fügt in den Anweisungen Class und Table die dritte Dimension hinzu, bei Table links von der zweidimensionalen Tabelle mit Komma abgetrennt:

```
Proc Tabulate Data=d.prdsale;
  Class country prodtype year;
  Var actual;
  Table year, prodtype, (country All)*actual*Mean;
Run;
```

Als Ergebnis erhalten wir zwei Tabellen der mittleren Verkaufszahlen getrennt für Möbeltypen und Länder für die Jahre 1993 und 1994. Standardmäßig werden die beiden Tabellen auf getrennten Seiten ausgegeben. Um diesen Seitenvorschub zu unterbinden, kann man die Option Condense in der Anweisung Table ergänzen, so dass – sofern genügend Platz vorhanden ist – mehrere Tabellen auf einer Seite platziert werden.

```
Proc Tabulate Data=d.prdsale;
  Class country prodtype year;
  Var actual;
  Table year, prodtype, (country All)*actual*Mean
    / Condense;
Run;
```


5. Tabellenkosmetik

Nachdem die Erstellung von ein-, zwei- und dreidimensionalen Tabellen mit mehreren Kennwerten und Gruppierungen nach den ersten vier Kapiteln kein Problem mehr darstellen sollte, werden wir uns der Verschönerung der Tabellen widmen und hierzu insbesondere drei Punkte ansprechen: (1) Die Spalten- und Zeilenbeschriftungen, (2) der Tabellentitel (Box) und (3) die Zellenbreite (Formate).

5.1 Spalten- und Zeilenbeschriftungen

Die Beschriftung der Spalten und Zeilen erfolgt standardmäßig mit den Variablenamen bzw. – sofern definiert – den Labels oder den Namen der statistischen Kenngrößen, etwa Mean. Diesen Standard kann man überschreiben, indem man nach dem Variablenamen = 'Text' schreibt. Um einen Standardwert zu entfernen, schreibt man Variable=' ' (oder Variable="").

Im ersten Beispiel hierzu wird die Zeile Country unterdrückt (mit country=' ') und anstelle von All wird Total ausgegeben:

```
Proc Tabulate Data=d.prdsale;
  Class country prodtype;
  Var actual;
  Table prodtype,(country=' ' All='Total')*actual*Mean;
Run;
```

	CANADA	GERMANY	U.S.A.	Total
	Actual Sales	Actual Sales	Actual Sales	Actual Sales
	Mean	Mean	Mean	Mean
Product type				
FURNITURE	509.71	527.05	476.91	504.56
OFFICE	517.80	502.79	506.19	508.93

C. Ortseifen

Im zweiten Beispiel

```
Proc Tabulate Data=d.prdsale;  
  Class country prodtype;  
  Var actual;  
  Table prodtype,  
    (country=' ' All='Total')*actual=' '*Mean=' '  
Run;
```

werden zusätzlich die Zeilen Actual Sales und Mean unterdrückt, indem ein Leerzeichen zugewiesen wird.

	CANADA	GERMANY	U.S.A.	Total
Product type				
FURNITURE	509.71	527.05	476.91	504.56
OFFICE	517.80	502.79	506.19	508.93

5.2 Titel - Option Box=

Nachdem der Name der Analysevariablen und die Kenngröße unterdrückt wurden, wird es erforderlich, dass die Tabelle eine Überschrift erhält. Natürlich kann man dazu die Anweisung Title verwenden. Betrachtet man allerdings die Tabelle genau, sieht man (vielleicht), dass sich auch die linke obere Zelle, die bislang noch leer ist, für eine Überschrift eignen würde. Diese Zelle wird nun mit der Option Box= gefüllt:

```
Proc Tabulate Data=d.prdsale;  
  Class country prodtype;  
  Var actual;  
  Table prodtype,  
    (country=' ' All='Total')*actual=' '*Mean=' '  
    / Box='Average Actual Sales';  
Run;
```

Average Actual Sales	CANADA	GERMANY	U.S.A	Total
Product type				
FURNITURE	509.71	527.05	476.91	504.56
OFFICE	517.80	502.79	506.19	508.93

5.3 Formate für die Zellenwerte

Die dritte der oben genannten Verschönerungsmöglichkeiten betrifft die Formatierung der einzelnen Zellen. Hierzu kann die Option `Format=` in der Anweisung `Proc Tabulate` verwendet werden.

```
Proc Tabulate Data=d.prdsale Format=Dollar10.;
  Class country prodtype;
  Var actual;
  Table prodtype,
    (country=' ' All='Total')*actual=' '*Mean=' '
    / Box='Average Actual Sales';
Run;
```

Average Actual Sales	CANADA	GERMANY	U.S.A.	Total
Product type				
FURNITURE	\$510	\$527	\$477	\$505
OFFICE	\$518	\$503	\$506	\$509

Mit dieser Option wird die Breite aller Spalten einheitlich festgelegt.

Einzelne Spalten und Zeilen können unabhängig vom mittels `Format=` festgelegten Ausgabeformat mit der Option `F=...`, die mit dem `Tabulate-Operator *` mit der betreffenden Variable verbunden werden muss, geändert werden.

C. Ortseifen

In folgendem Beispiel werden die Verkaufszahlen gruppiert nach Land mit jeweils 2 Nachkommastellen dargestellt, der Durchschnitt jedoch ohne die Dezimalstellen.

```
Proc Tabulate Data=d.prdsale Format=8.2;
  Class country prodtype;
  Var actual;
  Table prodtype,
    (country=' ' All='Total'*F=8.0)*actual=' '*Mean=' '
    / Box='Average Actual Sales';
Run;
```

Average Actual Sales	CANADA	GERMANY	U.S.A.	Total
Product type				
FURNITURE	509.71	527.05	476.91	505
OFFICE	517.80	502.79	506.19	509

Diese Form der Variablen-bezogenen Formatierung macht insbesondere dann Sinn, wenn verschiedene Kennwerte in den Zeilen dargestellt werden, wie etwa Anzahl der Beobachtungen (N) und Mittelwert wie im folgenden Beispiel:

```
Proc Tabulate Data=d.prdsale Format=8.2;
  Class country prodtype;
  Var actual;
  Table prodtype*actual*(N*F=3.0 Mean),
    (country=' ' All='Total'*F=8.0)
    / Box='Average Actual Sales';
Run;
```

Average Actual Sales			CANADA	GERMANY	U.S.A.	Total
Produ- ct type						
FURNI- TURE	Actual Sales	N	192	192	192	576
		Mean	509.71	527.05	476.91	505
OFFICE	Actual Sales	N	288	288	288	864
		Mean	517.80	502.79	506.19	509

Möchte man die Spalte Actual Sales mit

```
Table prodtype*actual=' '(N*F=3.0 Mean),
(country=' ' All='Total'*F=8.0)
```

unterdrücken, erhält man leider eine leere Spalte (vgl. hierzu auch SAS Note 002675):

Average Actual Sales			CANADA	GERMANY	U.S.A.	Total
Produ- ct type						
FURNI- TURE		N	192	192	192	576
		Mean	509.71	527.05	476.91	505
OFFICE		N	288	288	288	864
		Mean	517.80	502.79	506.19	509

6. Ausgabe als HTML-Dokument

Mit Hilfe des Output Delivery Systems (ODS) können die mit der Prozedur Tabulate erzeugten Tabellen leicht in ein HTML-Dokument überführt werden, das dann unabhängig vom SAS System mit einem beliebigen Browser (z.B: Internet Explorer, Netscape, Opera) betrachtet werden kann.

Notwendig ist eine Anweisung zum Öffnen des Html-Kanals und eine Anweisung zum Schließen. Beim Öffnen muss zusätzlich der Name der Datei angegeben werden, die erzeugt werden soll. Mit folgendem Programm wird die Datei test.html angelegt, die obige Tabelle jetzt allerdings in Html-Darstellung enthält:

```
Ods Html File='c:\test.html'
```

```
Proc Tabulate Data=d.prdsale Format=Dollar12.;  
  Class country prodtype;  
  Var actual;  
  Table prodtype,(country=' ' All='Total')*actual=' '*Mean=' '  
    / Box='Average Actual Sales';  
Run;
```

```
Ods Html Close;
```

Average Actual Sales	CANADA	GERMANY	U.S.A.	Total
Product type				
FURNITURE	\$510	\$527	\$477	\$505
OFFICE	\$518	\$503	\$506	\$509

Zusätzlich kann über die Option Style= der Name eines Stils angegeben werden, um die Tabelle in geänderter Form darzustellen. Mögliche Stile sind: Minimal, D3d, Statdoc, Theme, Barrettsblue, wobei dies nur eine kleine Auswahl ist. Die komplette Liste findet man in der Online Dokumentation zur Prozedur Tabulate.

```
Ods Html File='d:\test.html' Style=D3d;
```

Und es sind noch zahlreiche weitere Optionen möglich, wie das abschließende Beispiel zeigen soll. Die genaue Erläuterung zur Syntax finden Sie in der unten aufgeführten weiterführenden Literatur bzw. in der Online Dokumentation zur Prozedur Tabulate.

```
Ods Html File='d:\test.html';
```

```
Proc Tabulate Data=d.prdsale Format=Dollar12.;
  Class country prodtype;
  Var actual;
  Table prodtype,(country=' ' All='Total')*actual=' '*Mean=' '
    / Box=[Label='Average Actual Sales'
      style=[preimage='m:\ksfe8-vortrag\paris.gif']];
```

```
Run;
```

```
Ods Html Close;
```

 Average Actual Sales	CANADA	GERMANY	U.S.A.	Total
Product type				
FURNITURE	\$510	\$527	\$477	\$505
OFFICE	\$518	\$503	\$506	\$509

6.1 Ausgabe als Excel-Tabelle

Der Html-Kanal kann auch dazu verwendet werden, um Excel-Tabellen direkt zu erzeugen. Dies ist aber nun weniger eine Besonderheit von SAS als die Nutzung der Fähigkeit von Excel, Html-Dateien direkt einzulesen.

C. Ortseifen

```
Ods Html File='c:\test.xls'  
  
Proc Tabulate Data=d.prdsale Format=Dollar12.;  
  Class country prodtype;  
  Var actual;  
  Table prodtype,(country=' ' All='Total')*actual=' '*Mean=' '  
    / Box='Average Actual Sales';  
Run;  
  
Ods Html Close;
```

Für dieses Programm wurde nur der Dateiname von test.html abgeändert in test.xls. Intern wird die identische Information abgelegt. Öffnet man diese Datei aber mittels Doppelklick aus dem Explorer wird unter Windows, sofern auf dem Rechner installiert, das Excel-Programm gestartet.

Literatur

- [1] Lauren Haworth: Anyone Can Learn PROC TABULATE. SUGI27, Beginning Tutorials.
<http://www2.sas.com/proceedings/sugi27/Proceed27.pdf>
- [2] Lauren Haworth: SAS Reporting 101: REPORT, TABULATE, ODS, and Microsoft Office. SUGI 28, Beginning Tutorials.
<http://www2.sas.com/proceedings/sugi28/071-28.pdf>
- [3] Lauren Haworth: ODS for PRINT, REPORT and TABULATE. Advanced Tutorials, SUGI 26.
<http://www2.sas.com/proceedings/sugi26/p003-26.pdf>
- [4] Online Doc von SAS (Help > Books and Training > SAS Online Doc)
- [5] SAS System Hilfe (Help > The SAS System oder Kommando help tabulate und Concepts, liefert beispielsweise die verfügbaren Statistiken mit zugehöriger Option.)