

ANALYZING MULTIPLE RESPONSE QUESTIONS: FACILITATING AN EXISTING METHOD AND PRESENTING AN ALTERNATIVE

Paul Pope, Texas Agricultural Extension Service, Texas A&M University System
Darrell Fannin, Department of Rural Sociology, Texas A&M University

ABSTRACT

This paper addresses the difficulties associated with analyzing multiple response data with SAS® software. After an examination of the existing method for handling multiple response data (contained in the *SAS® Applications Guide*), two alternative methods are offered.

The first alternative method greatly facilitates the creation of value labels (an inherent difficulty with the *SAS Applications Guide* method). The second alternative method uses a different algorithm that eliminates the need for value labels.

INTRODUCTION

Analyzing responses to multiple response survey questions with SAS software is a difficult task. This is due to the fact that SAS software does not provide a dedicated procedure for analyzing multiple response data.

Absent a dedicated procedure, many alternative methods are available. Keene (1989), for example, offered PROC TABULATE as a solution. However, this particular application of PROC TABULATE is cumbersome and may be too formidable a task for inexperienced users of SAS software.

A different approach to analyzing multiple response data is provided by the SAS Institute (1987) in the *SAS Applications Guide* (presented below).

SAS APPLICATIONS GUIDE METHOD

This method uses a formula to convert each combination of question items (selected by respondents) into the value of a new variable. Consider the following multiple response survey question:

Q1. Which of the following SUGI conferences have you attended? (circle all that apply).

- A. SUGI 17 (Honolulu)
- B. SUGI 16 (New Orleans)
- C. SUGI 15 (Nashville)
- D. SUGI 14 (San Francisco)

Typically, for multiple response questions such as this, ease of data entry dictates that a separate variable be created for each question item (with each variable coded as 1 if the item is selected or (.) if the

item is not selected). For example, if a respondent selected A, B, and D then that observation would be coded as: 1 1 . 1

Program 1 (below) is an application of the method contained in the *SAS Applications Guide*. The program recodes missing values (.) as zeros, defines the variable Q1 (based on a formula to transform binary numbers to a decimal equivalent) to create a unique value for each combination of question items selected by respondents, creates the format Q1_FMT, and executes a PROC FREQ on Q1 with the Q1_FMT format. Output 1 contains the results of Program 1.

Program 1: SAS Applications Guide Method

```
DATA TEST (KEEP=Q1);
  INPUT Q1A Q1B Q1C Q1D;
  ARRAY ZERO Q1A--Q1D;
  DO OVER ZERO;
    IF ZERO=. THEN ZERO=0;
  END;
  Q1=Q1A*2**3+Q1B*2**2+Q1C*2+Q1D;
CARDS;
. . . .
. . . 1
. . 1 .
. . 1 1
. 1 . .
. 1 . 1
. 1 . 1
. 1 1 .
. 1 1 1
. 1 1 1
1 . . .
1 . . 1
1 . 1 .
1 . 1 1
1 1 . .
1 1 . 1
1 1 1 .
1 1 1 1
;
PROC FORMAT; VALUE Q1_FMT 0='NONE'
1='D only'
2='C only'
3='C and D'
4='B only'
5='B and D'
6='B and C'
7='B,C, and D'
8='A only'
9='A and D'
10='A and C'
11='A,C, and D'
12='A and B'
13='A,B and D'
14='A,B and C'
15='A,B,C and D';

TITLE 'SAS APPLICATIONS GUIDE METHOD';
PROC FREQ; FORMAT Q1 Q1_FMT.; RUN;
```

Output 1: Results of Program 1

SAS APPLICATIONS GUIDE METHOD				
Q1	Frequency	Percent	Cumulative Frequency	Cumulative Percent
NONE	1	5.6	1	5.6
D only	1	5.6	2	11.1
C only	1	5.6	3	16.7
C and D	1	5.6	4	22.2
B only	1	5.6	5	27.8
B and D	2	11.1	7	38.9
B and C	1	5.6	8	44.4
B,C, and D	2	11.1	10	55.6
A only	1	5.6	11	61.1
A and D	1	5.6	12	66.7
A and C	1	5.6	13	72.2
A,C, and D	1	5.6	14	77.8
A and B	1	5.6	15	83.3
A,B and D	1	5.6	16	88.9
A,B and C	1	5.6	17	94.4
A,B,C and D	1	5.6	18	100.0

Program 1 demonstrates that creating Q1 is not a difficult task. The formula for Q1 is easily modified to accommodate a question with any number of items. Unfortunately, creating labels for all the possible values of Q1 is a difficult task. The value labels were derived through a tedious process of determining the binary equivalent of each (decimal) value of Q1.

Chart 1 (below) illustrates how this process becomes exceedingly impractical as the number of question items increases. A question with 12 items, for example, potentially can yield 4096 different responses (and value labels)!

Chart 1: Possible Number of Values Labels for a Question

# of Items	Possible # of Value Labels
2	4
3	8
4	16
5	32
6	64
7	128
8	256
9	512
10	1024
11	2048
12	4096

THE FIRST ALTERNATIVE METHOD (An Easier Way of Creating Value Labels)

A solution to the value label difficulty posed by the *SAS Applications Guide* method is the algorithm in Program 2 (below). This algorithm uses a slight modification of the Q1 formula within a series of DO-END statements to create a value statement for a multiple response question with four items. The value statement is placed in a file called FMTS.4.

Note that the value of the ITEMS variable and the number of DO-END and IF-THEN-ELSE statements (in shaded area) equal the number of question items (4).

Program 2: Value Labels for a Question with 4 Items

```

FILENAME FMTS 'FMTS.4';
DATA _NULL_ FILE FMTS;
LENGTH FMT $16.;
ITEMS=4;
DO AA=1 TO 2;
  DO BB=1 TO 2;
    DO CC=1 TO 2;
      DO DD=1 TO 2;
        IF AA=2 THEN A='A' ELSE A='';
        IF BB=2 THEN B='B' ELSE B='';
        IF CC=2 THEN C='C' ELSE C='';
        IF DD=2 THEN D='D' ELSE D='';
        NEWFMT=(AA-1)*2**3+(BB-1)*2**2+
          (CC-1)*2**1+(DD-1)*2**0;
        FMT=COMPRESS(A||B||C||D);
        IF LENGTH(FMT)=ITEMS THEN FMT='ALL';
        IF FMT EQ '' THEN FMT='None';
        IF NEWFMT EQ 0 THEN
          PUT 'VALUE Q4FMT' @12 NEWFMT 6. '=' FMT $16. ' ';
        ELSE
          PUT @12 NEWFMT 6. '=' FMT $16. ' ';
      END;
    END;
  END;
END;
PUT ' '; PUT ' '; RUN;

```

Output 2 (below) contains the contents of FMTS.4. Note that the value statement is functionally equivalent to the value statement used with the *SAS Applications Guide* method (in Program 1).

Output 2: Results of Program 2

```

VALUE Q4FMT 0='None'      /
              1='D'        /
              2='C'        /
              3='CD'       /
              4='B'        /
              5='BD'       /
              6='BC'       /
              7='BCD'      /
              8='A'        /
              9='AD'       /
             10='AC'       /
             11='ACD'      /
             12='AB'       /
             13='ABD'      /
             14='ABC'      /
             15='ALL'      /

```

Output 3 (below) contains a partial listing of the file FMTS.7, produced by Program 3.

Output 3: Results of Program 3

VALUE	Q7FMT	
0	= 'None	/
1	= 'G	/
2	= 'F	/
3	= 'FG	/
4	= 'E	/
5	= 'EG	/
6	= 'EF	/
7	= 'EFG	/
8	= 'D	/
9	= 'DG	/
10	= 'DF	/
11	= 'DFG	/
12	= 'DE	/
13	= 'DEG	/
14	= 'DEF	/
15	= 'DEFG	/
16	= 'C	/
17	= 'CG	/
18	= 'CF	/
19	= 'CFG	/
20	= 'CE	/
21	= 'CEG	/
22	= 'CEF	/
23	= 'CEFG	/
24	= 'CD	/
25	= 'CDG	/
26	= 'CDF	/
27	= 'CDFG	/
28	= 'CDE	/
29	= 'CDEG	/
30	= 'CDEF	/
31	= 'CDEFG	/
32	= 'B	/
33	= 'BG	/
34	= 'BF	/
35	= 'BFG	/
36	= 'BE	/
37	= 'BEG	/
38	= 'BEF	/
39	= 'BEFG	/
.		/
.		/
.		/
107	= 'ABDFG	/
108	= 'ABDE	/
109	= 'ABDEG	/
110	= 'ABDEF	/
111	= 'ABDEFG	/
112	= 'ABC	/
113	= 'ABCG	/
114	= 'ABCF	/
115	= 'ABCFG	/
116	= 'ABCE	/
117	= 'ABCEG	/
118	= 'ABCEF	/
119	= 'ABCEFG	/
120	= 'ABCD	/
121	= 'ABCDG	/
122	= 'ABCDF	/
123	= 'ABCDFG	/
124	= 'ABCDE	/
125	= 'ABCDEG	/
126	= 'ABCDEF	/
127	= 'All	/

ADAPTING THE FIRST ALTERNATIVE METHOD TO ANY MULTIPLE RESPONSE QUESTION

The first alternative method's algorithm is easily adaptable to questions with up to 17 items (value labels are limited to 16 characters but "ALL" can be used as a label if all 17 items are selected).

To adapt the algorithm to questions with a different number of items, simply modify the ITEMS, NEWFMT and FMT statements and include the appropriate number of DO-END and IF-THEN-ELSE statements.

For example, from Program 2 (for a question with four items), an algorithm for a question with seven items is created by changing the value of ITEMS (to 7), inserting three additional DO-END and IF-THEN-ELSE statements, and modifying the formulas for NEWFMT and FMT (see shaded area of Program 3 below).

Program 3: Value Labels For a Question with 7 Items

```

FILENAME FMTS 'FMTS.7';
DATA NULL; FILE FMTS;
LENGTH FMT $16.;
ITEMS=7;
DO AA=1 TO 2;
DO BB=1 TO 2;
DO CC=1 TO 2;
DO DD=1 TO 2;
DO EE=1 TO 2;
DO FF=1 TO 2;
DO GG=1 TO 2;
IF AA=2 THEN A='A'; ELSE A='';
IF BB=2 THEN B='B'; ELSE B='';
IF CC=2 THEN C='C'; ELSE C='';
IF DD=2 THEN D='D'; ELSE D='';
IF EE=2 THEN E='E'; ELSE E='';
IF FF=2 THEN F='F'; ELSE F='';
IF GG=2 THEN G='G'; ELSE G='';
NEWFMT=(AA-1)*2**6+(BB-1)*2**5+
(CC-1)*2**4+(DD-1)*2**3+(EE-1)*2**2+
(FF-1)*2**1+(GG-1)*2**0;
FMT=COMPRESS(A||B||C||D||E||F||G);
IF LENGTH(FMT)=ITEMS THEN FMT='All';
IF FMT EQ '' THEN FMT='None';
IF NEWFMT EQ 0 THEN
PUT 'VALUE Q7FMT' @12 NEWFMT 6. ' '
FMT $16. ' ' ;
ELSE
PUT @12 NEWFMT 6. " " FMT $16. " " ;
END;
END;
END;
END;
END;
END;
PUT ';;'; PUT ' ';
RUN;

```

THE SECOND ALTERNATIVE METHOD

For SAS software users who prefer not to use *SAS Applications Guide* method (even with the value label facilitation), another simple, alternative method is available. This method eliminates the need for value labels.

This method, like the *SAS Applications Guide* method, creates a variable to represent each combination of question items selected by respondents. However, rather than defining the variable to be numeric (for which value labels must be created), the variable is defined to be character. The value of the variable is the combination of item labels (ie. A,B, etc.) circled on the survey instrument. This is accomplished with a series of statements using concatenation and the COMPRESS function. The example below (for a question with four items) illustrates this method.

Program 4: For a Question with 4 Items

```
DATA TEST (KEEP=Q1);
  LENGTH Q1 $6.;
  INPUT Q1A Q1B Q1C Q1D;
  IF Q1A=1 THEN Q1='A';
  IF Q1B=1 THEN Q1=COMPRESS(Q1||'B');
  IF Q1C=1 THEN Q1=COMPRESS(Q1||'C');
  IF Q1D=1 THEN Q1=COMPRESS(Q1||'D');
  IF LENGTH(Q1)=0 THEN Q1='-None-';
CARDS;
. . . .
. . . 1
. . 1 .
. . 1 1
. 1 . .
. 1 . 1
. 1 . 1
. 1 1 .
. 1 1 1
. 1 1 1
1 . . .
1 . . 1
1 . 1 .
1 . 1 1
1 1 . .
1 1 . 1
1 1 1 .
1 1 1 1
;

TITLE 'SECOND ALTERNATIVE METHOD';
PROC FREQ; RUN;
```

Note that this method produces a frequency distribution of the new variable (Q1) that is functionally equivalent to that produced by the *SAS Applications Guide* method, with the only difference being the order of the value labels in the table (see Output 4).

Output 4: Comparison of Results (Program 1 v. Program 4)

SAS APPLICATIONS GUIDE METHOD				
Q1	Frequency	Percent	Cumulative Frequency	Cumulative Percent
NONE	1	5.6	1	5.6
D only	1	5.6	2	11.1
C only	1	5.6	3	16.7
C and D	1	5.6	4	22.2
B only	1	5.6	5	27.8
B and D	2	11.1	7	38.9
B and C	1	5.6	8	44.4
B,C, and D	2	11.1	10	55.6
A only	1	5.6	11	61.1
A and D	1	5.6	12	66.7
A and C	1	5.6	13	72.2
A,C, and D	1	5.6	14	77.8
A and B	1	5.6	15	83.3
A,B and D	1	5.6	16	88.9
A,B and C	1	5.6	17	94.4
A,B,C and D	1	5.6	18	100.0

SECOND ALTERNATIVE METHOD				
Q1	Frequency	Percent	Cumulative Frequency	Cumulative Percent
-None-	1	5.6	1	5.6
A	1	5.6	2	11.1
AB	1	5.6	3	16.7
ABC	1	5.6	4	22.2
ABCD	1	5.6	5	27.8
ABD	1	5.6	6	33.3
AC	1	5.6	7	38.9
ACD	1	5.6	8	44.4
AD	1	5.6	9	50.0
B	1	5.6	10	55.6
BC	1	5.6	11	61.1
BCD	2	11.1	13	72.2
BD	2	11.1	15	83.3
C	1	5.6	16	88.9
CD	1	5.6	17	94.4
D	1	5.6	18	100.0

ASSESSMENT OF THE ALTERNATIVE METHODS

The second alternative method offers a number of advantages over the *SAS Applications Guide* and first alternative methods. These advantages include the following:

- It is an easily understood (or at least easily followed) method that inexperienced SAS software users can handle.
- It does not require a conversion of missing values to zeros.
- The value labels are arranged in more aesthetically pleasing (alphabetical) order.
- It does not require formats (releasing memory that, otherwise, would be consumed by the format).

- Because the new variable is defined as character and contains items labels (ie. A,B, etc.), this method allows for frequency distributions on the new variable for a subset of observations (an impossible task for the other two methods). Program 5 performs this task:

Program 5: Frequency Distributions For Subsets

```
DATA TEST;
LENGTH Q1 $6.;
INPUT Q1A Q1B Q1C Q1D;
  IF Q1A=1 THEN Q1='A';
  IF Q1B=1 THEN Q1=COMPRESS(Q1||'B');
  IF Q1C=1 THEN Q1=COMPRESS(Q1||'C');
  IF Q1D=1 THEN Q1=COMPRESS(Q1||'D');
  Q1A=INDEX(Q1,'A');
  Q1B=INDEX(Q1,'B');
  Q1C=INDEX(Q1,'C');
  Q1D=INDEX(Q1,'D');
CARDS;
. . . .
. . . 1
. . 1 .
. . 1 1
. 1 . .
. 1 . 1
. 1 . 1
. 1 1 .
. 1 1 1
. 1 1 1
1 . . .
1 . . 1
1 . 1 .
1 . 1 1
1 1 . .
1 1 . 1
1 1 1 .
1 1 1 1
;
TITLE "Q1 - RESPONSES CONTAINING 'A'";
PROC FREQ; TABLES Q1; WHERE Q1A>0; RUN;

TITLE "Q1 - RESPONSES CONTAINING 'B'";
PROC FREQ; TABLES Q1; WHERE Q1B>0; RUN;

TITLE "Q1 - RESPONSES CONTAINING 'C'";
PROC FREQ; TABLES Q1; WHERE Q1C>0; RUN;

TITLE "Q1 - RESPONSES CONTAINING 'D'";
PROC FREQ; TABLES Q1; WHERE Q1D>0; RUN;

TITLE "Q1 - RESPONSES CONTAINING 'B' AND 'D'";
PROC FREQ; TABLES Q1; WHERE Q1B>0 AND Q1D>0; RUN;
```

Program 5 creates Q1 and replaces the values of Q1A, Q1B, Q1C, and Q1D with the results of an INDEX function. The INDEX function returns, for the variable in the first argument (Q1), the position of the character string in the second argument (ie. 'A'). For example, if Q1 contains an 'A' for a particular observation then Q1A=INDEX(Q1,'A') would return a value greater than zero (1, the position number of A) for that observation.

WHERE statements are used with PROC FREQ to include only those observations containing the desired item (ie. 'A') in Q1. Output 5 contains the results of Program 5.

Output 5: Results of Program 5

Q1 - RESPONSES CONTAINING 'A'

Q1	Frequency	Percent	Cumulative Frequency	Cumulative Percent
A	1	12.5	1	12.5
AB	1	12.5	2	25.0
ABC	1	12.5	3	37.5
ABCD	1	12.5	4	50.0
ABD	1	12.5	5	62.5
AC	1	12.5	6	75.0
ACD	1	12.5	7	87.5
AD	1	12.5	8	100.0

Q1 - RESPONSES CONTAINING 'B'

Q1	Frequency	Percent	Cumulative Frequency	Cumulative Percent
AB	1	10.0	1	10.0
ABC	1	10.0	2	20.0
ABCD	1	10.0	3	30.0
ABD	1	10.0	4	40.0
B	1	10.0	5	50.0
BC	1	10.0	6	60.0
BCD	2	20.0	8	80.0
BD	2	20.0	10	100.0

Q1 - RESPONSES CONTAINING 'C'

Q1	Frequency	Percent	Cumulative Frequency	Cumulative Percent
ABC	1	11.1	1	11.1
ABCD	1	11.1	2	22.2
AC	1	11.1	3	33.3
ACD	1	11.1	4	44.4
BC	1	11.1	5	55.6
BCD	2	22.2	7	77.8
C	1	11.1	8	88.9
CD	1	11.1	9	100.0

Q1 - RESPONSES CONTAINING 'D'

Q1	Frequency	Percent	Cumulative Frequency	Cumulative Percent
ABCD	1	10.0	1	10.0
ABD	1	10.0	2	20.0
ACD	1	10.0	3	30.0
AD	1	10.0	4	40.0
BCD	2	20.0	6	60.0
BD	2	20.0	8	80.0
CD	1	10.0	9	90.0
D	1	10.0	10	100.0

Q1 - RESPONSES CONTAINING 'B' AND 'D'

Q1	Frequency	Percent	Cumulative Frequency	Cumulative Percent
ABCD	1	16.7	1	16.7
ABD	1	16.7	2	33.3
BCD	2	33.3	4	66.7
BD	2	33.3	6	100.0

The first alternative method, however, is not without advantages of its own. These advantages include the following:

- ▶ Compared to the *SAS Applications Guide* method it greatly facilitates the creation of value labels.
- ▶ It is a one-time only endeavor. Once the formats are created (for questions with 1-17 items), they can be used with any multiple response question, in all future applications. This represents a significant advantage over the second alternative method which requires a set of statements (IF statements creating the new variable, ie. Q1) for each multiple response question on your survey instrument.

We do not recommend one alternative method over the other. The decision to use a particular alternative method depends on your requirements and/or preferences, relative to your particular application.

CONCLUSION

Using SAS software to analyze multiple response data can be a difficult, tedious process. However, in our view, the two alternative methods presented in this paper offer SAS software users flexibility and makes the task of analyzing multiple response data less formidable.

REFERENCES

Keene, T. (1989), "Analyzing Multiple Response Data Using the TABULATE PROCEDURE." *Proceedings of the SAS Users Group International Conference*, 14, Cary, NC: SAS Institute Inc.

SAS Institute, Inc. (1987), *SAS Applications Guide*, 1987 Edition. Cary, NC: SAS Institute, Inc.

SAS software is a registered trademark or trademark of SAS Institute Inc. in the USA and other countries.
® indicates USA registration.