A strategy for understanding your data: Binary Flags and PROC MEANS
Glen Masuda, SRI International, Menlo Park, CA
Tejaswini Tiruke, SRI International, Menlo Park, CA

ABSTRACT
Many times projects have an enormous number of variables in the dataset so it's often necessary to create binary flags (a.k.a. indicator variables, 1=Yes 0=No) to better understand the data. These variables are a simple yet powerful tool for understanding, coding, and analyzing your data. This paper shows how to create binary flags in a DATA step; filter data based on the flags, and efficiently view patterns in the data using the flags with the Means and Summary Procedures. All examples use Base SAS© for PC, version 9.3.

INTRODUCTION

PROBLEM: TOO MUCH INFORMATION?

An indicator variable is an artificial variable created to represent an attribute with two or more distinct categories/levels. Consider data such as unemployment rate, GDP growth and government debt that are good indicators of the state of an economy. In contrast, a precise indicator shows the state of a binary or dichotomous variable (e.g. survived vs not, alive vs dead, present vs absent, male vs female) by 1 or 0. Often, which state is coded 1 and which is 0 is a matter of convention or convenience, as in the case of male or female. Indicator variables are perhaps more often called dummy variables or Binary Flags. Throughout the paper we use term "Binary Flags".

Often we have data coming from many places with lots of information and we need one dataset to handle the majority of analytical and reporting needs. So, with the help of binary flags we can get the data together. To save time and make your program less complicated, you might want to combine these flags and send several pieces of information in one go. Binary flags can be conveniently used to summarize the information with the help of FREQ and MEANS procedures. Specifically these flags are useful if you are dealing with the multilevel data. e.g Students , classrooms , test and course information.

SAMPLE DATA: STUDENT DEMOGRAPHIC INFORMATION

In this paper we will use the sample data shown in table 1: student demographic information for four years from 2010-11 to 2013-14 with the below information.

<table>
<thead>
<tr>
<th>ID</th>
<th>School</th>
<th>Grade</th>
<th>Gender</th>
<th>Ethnicity</th>
<th>read nrt scr</th>
<th>math nrt scr</th>
</tr>
</thead>
<tbody>
<tr>
<td>001</td>
<td>3456</td>
<td>01</td>
<td>Male</td>
<td>HISPANIC</td>
<td>.</td>
<td>520</td>
</tr>
<tr>
<td>002</td>
<td>2345</td>
<td>01</td>
<td>Female</td>
<td>BLACK</td>
<td>643</td>
<td>574</td>
</tr>
<tr>
<td>003</td>
<td>5643</td>
<td>K</td>
<td>Male</td>
<td>ASIAN</td>
<td>564</td>
<td>.</td>
</tr>
<tr>
<td>004</td>
<td>3456</td>
<td>PK</td>
<td>Male</td>
<td>HISPANIC</td>
<td>.</td>
<td>462</td>
</tr>
</tbody>
</table>
A strategy for understanding your data: Binary Flags and PROC MEANS, continued

Table 1. Sample student demographic information for 2010-11

<table>
<thead>
<tr>
<th>ID</th>
<th>Gender</th>
<th>Race</th>
<th>Score</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>005</td>
<td>K</td>
<td>Female</td>
<td>HISPANIC</td>
<td>554</td>
</tr>
<tr>
<td>006</td>
<td>PK</td>
<td>Female</td>
<td>BLACK</td>
<td>567</td>
</tr>
<tr>
<td>007</td>
<td>01</td>
<td>Male</td>
<td>WHITE</td>
<td>620</td>
</tr>
</tbody>
</table>

As an example, we want to see student attrition from 2010-11 to 2013-14 for each individual grade level and by treatment and control status for the schools.

I merged two datasets to combine the 2010-11 and 2013-14 year data (not shown here)

We can create flags to understand the data between two years. Flags can be created in below instances?

- Do the students have scores for the both years?
- Are the students in same school?
- Are the students at the expected grade level?
- Is the student in a treatment or control school?

Be careful about the missing values while creating flags as it might lead to the undesired results.

```plaintext
/* DO THEY HAVE SCORES? */
if nmiss(of read_nrt_1011 math_nrt_1011) = 2 then fl_1011_scr = 0;
else fl_1011_scr = 1;

if nmiss(of read_nrt_1314 math_nrt_1314 ) = 2 then fl_1314_scr = 0;
else fl_1314_scr = 1;

/* ARE THEY IN THE SAME SCHOOL? */
if sch_1011 = sch_1314 then fl_same_sch = 1;
else fl_same_sch = 0;

/* TX OR CTRL? */
if sch_1011 in('3456', '5643') then fl_sch_tx = 0; /*Control*/
else if sch_1011 in('2345', '2354') then fl_sch_tx = 1; /*Treatment*/

/* EXPECTED GRADE LEVEL */
if (gr_1011 = 'PK' and gr_1314 = '02') or
  (gr_1011 = '00' and gr_1314 = '03') or
  (gr_1011 = '01' and gr_1314 = '04') or
  (gr_1011 = '02' and gr_1314 = '05')
then fl_exp_gr = 1;
else fl_exp_gr = 0;

Once you have binary flags for several variables you can create more combined flags. As shown below

/* MORE COMBINED FLAGS */
/*have both 10/11 and 13/14 scores*/
if fl_1011_scr = 1 and fl_1314_scr = 1 then fl_1011_1314_scr = 1;
else fl_1011_1314_scr = 0;

/*have both scores and in the same school*/
if fl_1011_1314_scr = 1 then do;
  if fl_same_sch = 1 then fl_1011_1314_scr_same_sch = 1;
  else fl_1011_1314_scr_same_sch = 0;
  if fl_exp_gr = 1 then fl_1011_1314_scr_exp_gr = 1;
  else fl_1011_1314_scr_exp_gr = 0; /*in expected grade*/;
```
A strategy for understanding your data: Binary Flags and PROC MEANS, continued

```plaintext
if fl_same_sch = 1 and fl_exp_gr = 1 then fl_1011_1314_scr_sch_gr = 1;
   else fl_1011_1314_scr_sch_gr = 0; /*in same school and in expected grade*/;
end;
else do:
   fl_1011_1314_scr_same_sch = 0;
   fl_1011_1314_scr_exp_gr = 0;
   fl_1011_1314_scr_sch_gr = 0;
end;

/* restricting to students only to those with 2013-14 score, */
/* but not necessarily with 2010-11 score*/
/* - N with 2010-11 and 2013-14 test scores and in the same school */
/* - N with 2010-11 and 2013-14 data with 2013-14 test scores and at expected grade level */
/* - N with 2010-11 and 2013-14 data with 2013-14 test scores, in the same school, and at expected grade level*/

if fl_1314_scr = 1 then do;
   if fl_same_sch = 1 then fl_1314_scr_same_sch = 1;
   if fl_exp_gr = 1 then fl_1314_scr_exp_gr = 1;
   if fl_same_sch = 1 and fl_exp_gr = 1 then fl_1314_scr_sch_gr = 1;
end;
else do:
   fl_1314_scr_same_sch = 0;
   fl_1314_scr_exp_gr = 0;
   fl_1314_scr_sch_gr = 0;
end;

if gr_1011 = 'PK' then grade_1011 = 1;
else if gr_1011 = '00' then grade_1011 = 2;
else if gr_1011 = '01' then grade_1011 = 3;
else if gr_1011 = '02' then grade_1011 = 4;
else if gr_1011 = '03' then grade_1011 = 5;

DESCRIPTIVE STATISTICS USING BINARY FLAGS USING PROC FREQ AND PROC MEANS

The LIST and MISSPRINT options in FREQ procedure are very useful in cases where an n-way table is produced but only the frequency counts and percent are required.

Below is the output from FREQ procedure with the LIST option

<table>
<thead>
<tr>
<th>fl_exp_gr</th>
<th>gr_1011</th>
<th>gr_1314</th>
<th>Frequency</th>
<th>Percent</th>
<th>Cumulative</th>
<th>Cumulative</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>PK</td>
<td>Grade 1</td>
<td>779</td>
<td>1.05</td>
<td>9809</td>
<td>13.23</td>
</tr>
<tr>
<td>0</td>
<td>PK</td>
<td>Grade 3</td>
<td>2</td>
<td>0.00</td>
<td>9811</td>
<td>13.23</td>
</tr>
<tr>
<td>0</td>
<td>PK</td>
<td>PK</td>
<td>125</td>
<td>0.17</td>
<td>9936</td>
<td>13.40</td>
</tr>
<tr>
<td>1</td>
<td>K</td>
<td>Grade 3</td>
<td>19907</td>
<td>26.85</td>
<td>29843</td>
<td>40.25</td>
</tr>
<tr>
<td>1</td>
<td>Grade 1</td>
<td>Grade 4</td>
<td>19303</td>
<td>26.04</td>
<td>49146</td>
<td>66.29</td>
</tr>
<tr>
<td>1</td>
<td>Grade 2</td>
<td>Grade 5</td>
<td>20204</td>
<td>27.25</td>
<td>69350</td>
<td>93.54</td>
</tr>
<tr>
<td>1</td>
<td>PK</td>
<td>Grade 2</td>
<td>4791</td>
<td>6.46</td>
<td>74141</td>
<td>100.00</td>
</tr>
</tbody>
</table>

Frequency Missing = 785

Table 2

Output in table 2 shows how many students are at the expected grade level over the years.
A strategy for understanding your data: Binary Flags and PROC MEANS, continued

Table 3
Output in table 3 shows how many students have 2013-14 data with test scores, in the same school, and at expected grade level.

After exploring the data, we can summarize the data by grade and Condition (treatment/control) using MEANS Procedure.

```sas
proc means data=match_1011 n mean std maxdec=2 ;
  var fl_1011_scr fl_1314_scr fl_1011_1314_scr
       fl_1011_1314_scr_same_sch fl_1011_1314_scr_exp_gr;
  class grade_1011 fl_sch_tx;
  output
    out=sum_tx
    sum=
; run;
```

<table>
<thead>
<tr>
<th>grade_1011</th>
<th>fl_sch_tx</th>
<th>N Obs</th>
<th>Variable</th>
<th>N</th>
<th>Mean</th>
<th>Std Dev</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0</td>
<td>1942</td>
<td>fl_1011_scr</td>
<td>1942</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>fl_1314_scr</td>
<td>1942</td>
<td>0.83</td>
<td>0.38</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>fl_1011_1314_scr</td>
<td>1942</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>fl_1011_1314_scr_same_sch</td>
<td>1942</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>fl_1011_1314_scr_exp_gr</td>
<td>1942</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1986</td>
<td>fl_1011_scr</td>
<td>1986</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>fl_1314_scr</td>
<td>1986</td>
<td>0.82</td>
<td>0.39</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>fl_1011_1314_scr</td>
<td>1986</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>fl_1011_1314_scr_same_sch</td>
<td>1986</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>fl_1011_1314_scr_exp_gr</td>
<td>1986</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>1949</td>
<td>fl_1011_scr</td>
<td>1949</td>
<td>0.96</td>
<td>0.19</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>fl_1314_scr</td>
<td>1949</td>
<td>0.83</td>
<td>0.38</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>fl_1011_1314_scr</td>
<td>1949</td>
<td>0.82</td>
<td>0.38</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>fl_1011_1314_scr_same_sch</td>
<td>1949</td>
<td>0.60</td>
<td>0.49</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>fl_1011_1314_scr_exp_gr</td>
<td>1949</td>
<td>0.72</td>
<td>0.45</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>2002</td>
<td>fl_1011_scr</td>
<td>2002</td>
<td>0.96</td>
<td>0.19</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>fl_1314_scr</td>
<td>2002</td>
<td>0.83</td>
<td>0.38</td>
</tr>
</tbody>
</table>
By specifying `grade_1011` and `fl_sch_tx` in the CLASS Statement, we now have the MEAN and N and STD of each variable specified in the VAR statement for each unique value of grade and condition (control/treatment). By default, PROC MEANS shows the number of observations for each value of the classification variable. The OUTPUT Statement directs the placement, in a temporary SAS data set, of new variables containing the results of the analyses requested elsewhere in the PROC MEANS Task. The Statistics Keyword SUM requests the sum, of the analysis variable be placed in a variable in output data set `sum_tx`.

Also, we can use percentile option (p10, p25, p50, p75) in the MEANS statement to get the distribution of the grouped data.

REFERENCES


ACKNOWLEDGMENTS
We would like to thank all the team members at SRI International for encouraging us for participation at WUSS.

A big thanks to Patrick Thornton, Cyndi Williamson and Mary McCracken for their review comments.

RECOMMENDED READING
• Base SAS® Procedures Guide
• SAS® For Dummies®

CONTACT INFORMATION
Your comments and questions are valued and encouraged. Contact the author at:

Tejaswini Tiruke
Scientific Programmer
SRI International, Center for Education and Human Services
Menlo Park, CA
Phone: 650 859-2906
E-mail: Tejaswini.tiruke@sri.com

Glen Masuda
Programmer/Analyst
SRI International, Center for Education and Human Services
Menlo Park, CA
Phone: 650 859-3949
E-mail: glen.masuda@sri.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.
Other brand and product names are trademarks of their respective companies.