Getting an Overview of SAS® Data in Three Steps

Yu Fu[1], Shirmeen Virji[1], Miriam McGaugh[2], and Goutam Chakraborty[2]
[1]Management Science and Information Systems, Oklahoma State University, Stillwater, OK 74078
[2]Marketing, Oklahoma State University, Stillwater, OK 74078

Introduction

We have all been in a situation where we are given a very big library full of files and don't know where to begin our analysis. Or worse, if a newly hired employee has been given disks full of datasets, but he has no idea what variables are common in the files. What about the datasets that have to be merged with the existing dataset, but the types of the variables are different and it will take a long time to go into every file and check for particular details? While working SAS on daily basis, many users face these problems and the only option available to them to answer these questions is to spend more and more of their precious time to process this mundane information. We have created this macro to resolve all these problems in a three-step macro program. The three steps that our macro program consists of are variables, statistics, and relationships.

Methods

Step one: Generating Column Properties

Firstly, macro program produces a table of column properties that contains the number of variables, name of the dataset of which they belong to, variable name, type, length of the variable, starting point, format structure (if any), and label.

The output produced by the CONTENTS Statement combines several parts. The code to specify data sets to which CONTENTS data is directed is shown Figure1.

```
ods output attributes=atr
           variables=var
           enginehost=eng
           indexes=ind
           integrityconstraints = ic
           sortedby= sb;
```

Figure 1. The code to specify data sets.

Normally, the CONTENTS Statement is used to write out the contents of one or more SAS data sets, but in order to include only variable information in our output, we use the following code:

```
ods output variables= var;
ods listing close;
proc contents data=dsn;
run;
```

Step two: Generating Summary Statistics

Secondly, macro program produces analysis of the variables in which it prints out statistics of all the numeric variables present in the dataset. The resulting columns are number of observations, mean, standard deviation, and minimum value, and maximum value of the variable.

By running the code below, it would generates the statistical description of the SAS data set and gives a statistical overview of numeric variables in the data set.

```
proc means data=dsn maxdec=2;
run;
```

Step three: Generating Relationship Diagram

Lastly, macro, by using the power of GraphicViz, produces a relationship diagram where the relationships among datasets are shown graphically. If any dataset has a variable that has the same name and type that another dataset in the library also has, macro will draw that out on the report.

In this section, the macro program interacts with an external graph visualization application to create the relationship image and then produces the image onto our output. We use open source visualization software called "Graphviz" to generate graphical relationships in three steps.

The first step is to create the Graphviz description text language for each dataset in the library. The next step is to save the completed Graphviz program into a dot file. The third step is to call the Graphviz software to run the program that was generated in the second step. The detailed code is shown in the additional code section.

Results

The sample output for this macro program in pdf is shown in Figure 2 below.

Data Dictionary For test Data Set city 14:35 Monday, November 1

Column Properties Of city

Obs	Member	Num	Variable	Type	Len	Pos	Format	Label
1	TEST.CITY	1	City_ID	Num	8	0	12.	City ID
2	TEST.CITY	2	City_Name	Char	30	8		City Name
3	TEST.CITY	3	Country	Char	2	38		Country

Summary Statistics Of city

Analysis Variable : City_ID City ID				
N	Mean	Std Dev	Minimum	Maximum
31	9260000829.9	436.43	9260000017.0	9260001549.0

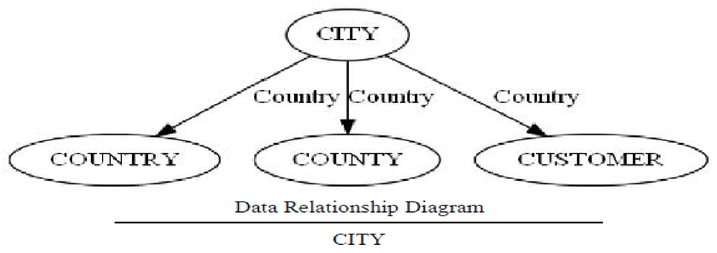

Figure 2. The sample output in pdf file.

Limitations

In this SAS macro, an overview can only be created for sas7bdat file. Other types of dataset files will be ignored. Most of the work to support the other type of data files will be on the dataset reading. We are working to add support for other types of data files especially for the raw data file.

Conclusion

In this paper, we introduced a macro program to generate the information for a SAS data set including variables information, statistical description, and relationship diagram. This program can help programmers to build the data dictionary of data set quickly and to be familiar with the data set in a short time.

Fu,
Virji,
McGaugh,
And Dr. Chakraborty-

Additional Code

Code to generate the Relationship Diagram:

```
%let header= digraph G {;
%let tail = };
%let command=;
%let id=%sysfunc(open(rmap));
%let NObs=%sysfunc(attrn(&id,NOBS));
%let rc = %sysfunc(close(&id));
data _null_;
    set Rmap;
    call symputx("fds"||left(_n_),membera,'L');
    call symputx("sds"||left(_n_),memberb,'L');
    call symputx("variable"||left(_n_),variable,'L');
run;
%do m=1 %to &num;
%let command=;
%let dsn=%scan(&dslist,&m);
%do n=1 %to &NObs;
    %let fmem = %sysfunc(tranwrd(&&fds&n,%upcase(&lib..),));
    %let smem = %sysfunc(tranwrd(&&sds&n,%upcase(&lib..),));
        %if %upcase(&dsn)= &fmem %then
            %do;
                %let str = &fmem.->&smem.[label=&&variable&n]%str(;);
                %let command = &command&str;
                %put ERROR: &command;
            %end;
        %else %if %upcase(&dsn)= &smem %then
            %do;
                %let str = &smem.->&fmem.[label=&&variable&n]%str(;);
                %let command = &command&str;
                %put ERROR: &command;
            %end;
       %else %put ERROR: &dsn &&fds&n &&sds&n &&variable&n;
%end;

data _null;
    file "c:\temp\relation.dot";
    put "&header&command&tail";
run;

data _null_;
X 'cd G:\Graphviz\bin';
call system("G:\Graphviz\bin\dot -Tjpg c:\temp\relation.dot -o
c:\temp\&dsn..jpg");
run;
```

Reference

Bessler,LeRoy. 2005. “Getting Started with, and Getting the Most out of, SAS® ODS PDF: No Mastery of PROC TEMPLATE Required. ” SAS Conference Proceedings: Technical Solutions, Phuse, Heidelberg, Germany

“DOT Language Document [Internet].” 2012[cited 2012 Nov 16]. Available at http://www.graphviz.org/content/dot-language