NODIS PROCEDURES FOR DEMOGRAPHIC ANALYSIS

Janice S. Tobin, Cleveland State University (NODIS)
Mark J. Salling, Cleveland State University (NODIS)

INTRODUCTION: SAS & NODIS

Statistical Analysis System (SAS) software is used at many research institutions. Its ease and flexibility in file maintenance and processing, its strong features as a high level programming language, and its extensive statistical analysis capabilities, make it probably the best single software package available for research organizations using large IBM, and now several other large computer systems. SAS is used extensively at the Northern Ohio Data & Information Service (NODIS) at Cleveland State University, for example, in processing data from the U.S. Bureau of Census, Bureau of Labor, parcel based county auditor's files, and a variety of other large data files.

NODIS is a data service both for research within The Urban Center in the College of Urban Affairs and for data users in northern Ohio. NODIS has been designated by the State of Ohio and the Bureau of the Census as the regional data center for northern Ohio. Among its functions, therefore, NODIS has the responsibility to provide census and other data to public and private organizations and businesses on an economical, cost-recovery basis. Data services include requests for standard printed reports as well as a variety of customized data reports, files and analyses. Clients include large and small municipalities, social planning agencies, law firms, hospitals, marketing firms, and developers. The majority of data services at NODIS require extracting, sorting, and aggregating data fields and records and then producing easily readable printed reports.

Staff at NODIS includes full-time and student part-time programmers, as well as data specialists and researchers with minimum programming skills. Although the programmers have the necessary skills for any programming required, it is often the non-programming staff which must employ SAS procedures to service data requests. Furthermore, most data service requests have several common components, making a modular approach, such as the SAS procedures library, ideal for effective and efficient response by an information service organization.

UNMET NEEDS

Although SAS is a very powerful software system we have found there are several programming needs at NODIS which are not easily satisfied by existing SAS procedures.

Aggregation

One of the unmet needs is the aggregation of data. The aggregation of data among observations using PROC SUMMARY is somewhat cumbersome and does not permit retention of unaggregated data fields in the output data set. For example, in aggregations of geographic census data it is usually necessary to retain an alphanumeric area name as well as other data with the summed variable(s). PROC SUMMARY does not permit carrying unaggregated fields to the output file. Furthermore, an additional DATA step is required in order to subset only aggregated records. Use of the created variable named TYPE, which defines the levels of aggregation in the output data set, can require considerable forethought and careful calculation in correctly subsetting the data.

Group Statistics

A second common programming function at NODIS, which is not available as a convenient SAS procedure, is the calculation of summary statistics such as means and medians for a set or group of variables on a single observation. Statistics are provided by SAS procedures for variables analyzed by observation. However, there is no SAS procedure which will calculate means and medians for an array of variables for a single observation.

For example, PROC MEANS will produce the mean number of persons under five years old among the sixty municipalities in Cuyahoga County, Ohio where each observation is a municipality and contains the total number of persons in age categories, including under five years. The calculated mean is based on the number of municipalities (60) in the county and the aggregation of the summary...
totals of population by age cohort. Similarly, medians can be easily produced using PROC UNIVARIATE.

However, computation of the mean or median age for a single municipality requires either special programming or at least very clever restructuring of the data. The need to calculate statistics for groups of variables, such as categories of age, income, rent, and housing value is frequent for organizations such as NODIS, which work with such demographic data.

Report Writing

Cleveland State University has been employing SAS79.5. This version has a very limited report writing procedure. Customized reports require more extensive planning and programming using PUT and other SAS statements. Under SAS79.5, PROC PRINT has very limited options for constructing data tables. Labels are not employed. The user has little control over the number of columns printed. There are no provisions for a title page, vertical or horizontal bars, border lines, spacing between columns, nor end of table "NOTE", "PREPARED BY" and "SOURCE" statements. The ability to easily specify these options is necessary to facilitate the production of clear and readable printed reports of Census and other data provided by NODIS.

More recent versions of SAS promise improved PROC PRINT flexibility. For example, "LABEL" statements may be used as column headings.

Population Pyramids

Demographic analysis is an important service provided by NODIS. The population pyramid is a significant tool for a demographer, providing a graphic description of a region's age and sex distribution. A specialized procedure to easily produce population pyramids is not provided by SAS.

**INTERNAL PROCEDURES WRITTEN BY NODIS**

In order to fill the gaps in SAS procedures (PROC's) noted above, NODIS developed its own internal procedures library which is accessed by an additional line of Job Control Language (JCL). NODIS procedures are very similar to SAS procedures in the way users invoke them and their options.

NODIS procedures include:

1. NDSAGG for aggregating;
2. NDSGST for group statistics;
3. NDSRPT for table report writing; and
4. NDSPLYRM for population pyramids.

They are described in more detail below.

1. **NDSAGG**

The NDSAGG procedure produces a SAS data set which includes the aggregate of observations within a "BY" group. NDSAGG will also output the first or last observation's non-aggregated variables for each "BY" group. Thus character variables may be retained with records of aggregated data. PROC NDSAGG is similar to PROC SUMMARY, except NDSAGG has the ability to carry variables other than the AGG (SUM) variables of PROC SUMMARY. NDSAGG does not produce a printed output except on the SAS log.

The following options may appear in the PROC NDSAGG statement.

- **AGGMISS=**
- **ALLOBS**
- **EXCMISS=**
- **MISSZERO**
- **SKIPMISS**

The AGGMISS= option allows the user to specify the value of the special missing value outputed by NDSAGG when an input AGGREGATE variable is encountered with a missing value or special missing value. The default special missing value is ".A". This value must be a single alpha character.

When ALLOBS is specified, observations from the input data set precede the aggregated observation of each "BY" group in the output data set. If ALLOBS is omitted, only aggregated observations are output.

The EXCMISS= option allows the user to specify the value of the special missing value output by NDSAGG when a missing value is encountered in the input of any EXCLUDE variable. The default special missing value is ".Z". This value must be a single alpha character.
If MISSZERO is specified, individual missing values are treated as zeroes for purposes of aggregation. Not specifying MISSZERO results in the use of a special missing value in place of aggregated values.

If SKIPMISS is specified, entire observations with any missing values will not be included in the aggregation. Observations will still be output if ALLOBJS is specified.

Statements applicable to NDSAGG include the following:

- **BY** variable list;
- **VARIABLES** variable list;
- **AGGREGATE** variable list;
- **EXCLUDE** variable list. The **BY** statement is normally used with PROC NDSAGG and causes an output observation to be generated when the **BY** group changes. NDSAGG assumes that the data set has been sorted by the variables in the **BY** group. Omission of the **BY** statement causes one aggregate observation to be output at the end of the output data set.

The **VARIABLES** list specifies the variables included in the output data set. If the **VARIABLES** statement is not included, all the variables in the input data set are included in the output data set.

The **AGGREGATE** variables list determines the variables from the input data set which are aggregated in the output data set. If the value of an input **AGGREGATE** variable is missing, a special missing value is output unless MISSZERO or SKIPZERO options have been specified.

The **EXCLUDE** statement is used to set variables to special missing value ".T". This can be useful when the first value of a non-aggregate variable is not appropriate.

2. NDSGST

The NDSGST procedure produces a SAS data set which includes the mean, and/or median, of a series of grouped variables, which represent a frequency distribution within each observation. NDSGST does not produce a printed output except on the SAS log.

**DATA=** and **OUT=** options specify input and output data sets, respectively. **MEAN=** and **MEDIAN=** statements provide lists of the variables for which means or medians are calculated. The **CENTRAL=** provides for lists of the variables for which both means and medians are computed. A **WT=** clause for each **MEAN=**, **MEDIAN=**, or **CENTRAL=** statement is required to specify the midpoint values for each variable/category.

3. NDSRPT

The NDSRPT procedure produces a standardized report from an input data set. If NDSRPT is unable to fit all of the requested variables on a page it divides them into as many pages as necessary.

The options below may appear in the PROC NDSRPT statement:

- **BYPAGE**
- **DATA=**
- **FIRSTID**
- **LPIB**
- **ONESPACE**
- **NOBORDER**
- **NOHBARS**
- **NOSPACE**
- **NOTPAGE**
- **NOVBARS**

The **BYPAGE** option forces NDSRPT to start a new page at the end of each **BY** group. The **DATA=** clause informs NDSRPT which data set to use. The **FIRSTID** option tells NDSRPT to use the first two ID variables instead of the last two. The **LPIB** option tells NDSRPT that the output is printed at 8 lines per inch instead of 6 lines per inch. The **ONESPACE** option specifies that one space is desired between the columns. If **NOBORDER** is specified, the five leading spaces, normally allotted for three-hole punching, do not appear. **NOHBARS** suppresses the output of any horizontal bars/lines which is normally produced by the procedure. When **NOSPACE** is used, the printed output will not contain extra spaces between columns. **NOTPAGE** turns off the request for a title page. **NOVBARS** excludes vertical bars separating each column.

Statements used with PROC NDSRPT include the following:

- **BY** variable lists;
- **FORMAT** variable list;
- **LABEL** variable list;
- **VARIABLES** list;
- **ID** variable list;
- **NOTE** text of footnote;
- **PREP** text of organization;
- **SOURCE** text of source;

The **BY**, **FORMAT**, and **VARIABLES** statements may be used in the same way...
as they are used with PROC PRINT. The ID statement may be used to assign identification variables to each observation for sectionalized output. Normally, the first section will contain as many ID variables as will fit on the page, and subsequent sections will contain only the first (or last) ID variables in the list.

4. NDSPYRM

The NDSPYRM procedure produces a population pyramid from each observation of an input data set. It will permit two populations to be compared in one pyramid by overlaying two separate symbols. Both total and nonwhite age and sex distributions, for example, can be portrayed in one pyramid.

The options below may appear in the PROC NDSPYRM statement.

- **BARCHAR=a**: The BARCHAR=a option permits the user to specify the character used to build bars. If BARCHAR is not specified an asterisk (*) is used.
- **BPERCENT**: The BPERCENT option specifies the use of percentages rather than frequencies.
- **DATA=**
- **OVRCHAR=a**: The OVRCHAR=a option permits the user to specify the character used to build bars for a second population, if desired. If OVRCHAR is not specified a pound sign (#) is used.
- **PACK**: The PACK option will allow more than one population pyramid to appear on a single page.

**Statements applicable to PROC NDSPYRM include:**

- **BY variable lists;**
- **LABEL variable = label text;**
- **LEFT left variable list** (youngest to oldest);
- **LLABEL text of left label;**
- **LOVRLY left variable lists for second population (youngest to oldest);**
- **NOTE text of footnote;**
- **PREP text describing organization;**
- **RIGHT right variable list** (youngest to oldest);
- **RLABEL text of right label;**
- **ROVRLY left variable list for a second population (youngest to oldest);**
- **SOURCE text of source statement;**

The BY statement may be used to change the title at the top of each page. The LABEL statement provides category labels in the trough. The LEFT statement is used to specify (the first set of) variables used on the left side of the population pyramid. The LLABEL statement is used to label the left side (first variable set). The LOVRLY statement is used to specify the overlay variables (the second set of variables) on the left side of the population pyramid. The RIGHT statement is used to specify variables used on the right side of the population pyramid. The RLABEL statement is used to label the right side of the population pyramid. The ROVRLY statement is used to specify the overlay (second set of) variables on the right side of the population pyramid.

**EXAMPLE**

The SAS code in Figure 1 generates the reports found in Figure 2. The input data set contains 1980 population by age category (variables) for each minor civil division (observation) in Geauga County, Ohio. The printed reports include a table of these data with aggregations for eastern and western halves of the county. Median ages for each municipality and region are also produced and printed. A population pyramid is also produced.

```sas
PROC NDASG;  AGG POP;  BY NAME; 
PROC NDSSST;  MEAN POPMEAN = POPOS POP610 POP1115 POP1620  WT = 2.5 7.5 12.5 17.5 ;
PROC NDSPRT;  VARS POP05 POP1115;  BY NAME; 
PROC NDSPYRM;  RIGHT F1 -- F20;  LEFT M1 -- M20; 
```

Figure 1. Sample code for NDIS SAS Proc
--- Fig. 2a. PROC NDSAGG ---

**INPUT DATASET**

<table>
<thead>
<tr>
<th>OBS</th>
<th>NAME</th>
<th>TRACT</th>
<th>POP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CLEVELAND</td>
<td>1011</td>
<td>2345</td>
</tr>
<tr>
<td>2</td>
<td>CLEVELAND</td>
<td>1012</td>
<td>3456</td>
</tr>
<tr>
<td>3</td>
<td>CLEVELAND</td>
<td>1013</td>
<td>4256</td>
</tr>
<tr>
<td>4</td>
<td>SHAKER HEIGHTS</td>
<td>1831</td>
<td>2934</td>
</tr>
<tr>
<td>5</td>
<td>SHAKER HEIGHTS</td>
<td>1832</td>
<td>2437</td>
</tr>
</tbody>
</table>

**OUTPUT DATASET**

<table>
<thead>
<tr>
<th>OBS</th>
<th>NAME</th>
<th>TRACT</th>
<th>POP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CLEVELAND</td>
<td>1011</td>
<td>2345</td>
</tr>
<tr>
<td>2</td>
<td>CLEVELAND</td>
<td>1012</td>
<td>3456</td>
</tr>
<tr>
<td>3</td>
<td>CLEVELAND</td>
<td>1013</td>
<td>4256</td>
</tr>
<tr>
<td>4</td>
<td>SHAKER HEIGHTS</td>
<td>1831</td>
<td>2934</td>
</tr>
<tr>
<td>5</td>
<td>SHAKER HEIGHTS</td>
<td>1832</td>
<td>2437</td>
</tr>
</tbody>
</table>

--- Fig. 2b. PROC NDSGST ---

**INPUT DATASET**

<table>
<thead>
<tr>
<th>TRACT</th>
<th>POP05</th>
<th>POP10</th>
<th>POP115</th>
<th>POP1620</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1011</td>
<td>200</td>
<td>300</td>
<td>400</td>
</tr>
<tr>
<td>2</td>
<td>1012</td>
<td>100</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>3</td>
<td>1013</td>
<td>400</td>
<td>400</td>
<td>300</td>
</tr>
<tr>
<td>4</td>
<td>1014</td>
<td>500</td>
<td>300</td>
<td>300</td>
</tr>
</tbody>
</table>

**OUTPUT DATASET**

<table>
<thead>
<tr>
<th>OBS</th>
<th>TRACT</th>
<th>POP05</th>
<th>POP10</th>
<th>POP115</th>
<th>POP1620</th>
<th>POPMEAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1011</td>
<td>200</td>
<td>300</td>
<td>400</td>
<td>300</td>
<td>10.8</td>
</tr>
<tr>
<td>2</td>
<td>1012</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>10.0</td>
</tr>
<tr>
<td>3</td>
<td>1013</td>
<td>400</td>
<td>400</td>
<td>300</td>
<td>300</td>
<td>9.3</td>
</tr>
<tr>
<td>4</td>
<td>1014</td>
<td>500</td>
<td>300</td>
<td>300</td>
<td>200</td>
<td>8.3</td>
</tr>
</tbody>
</table>

--- Fig. 2c. PROC NDSRPT ---

**LABEL Statement**

**TITLE Statement**

**VARIABLES Statement**

**BY Statement**

**NOTE PREP SOURCE Statement(s)**
CONCLUSION

The power of SAS lies in its ability to service a variety of users with a considerable range in programming skills. Less skilled users have a large array of easy to use programmed procedures upon which to call. Those with more advanced skills and interests can design their own procedures to better suit specific needs.

NODIS, having a staff with a wide diversity of programming skills, has matched its own diversity in skill levels to the diversity offered by SAS. More skilled programmers have written internal procedures which better meet our needs. Meanwhile, staff with lesser programming skills have been able to employ SAS and NODIS developed procedures to serve clients.

Much of the success of NODIS in providing an information service has been due to the development and use of the four procedures described above.

MORE INFORMATION

NODIS is providing to other SAS users the four NODIS written internal procedures discussed above. All four programs are available, on tape in load module form, for a standard tape copying service charge. NODIS also provides Census Bureau Summary Tape (STF) and Micro data files in SAS data set form, as well as procedures for estimating suppressed cells on STF data sets.