ABSTRACT

Two of the many challenges that exist for statisticians at AstraZeneca is one, generate statistical analysis without reinventing the wheel and two, how to convert text-based SAS output into Microsoft Word tables (and eliminate the manual typing process). This paper details the MacStat system was designed to solve these problems.

INTRODUCTION

The MacStat system was designed with two parts: a system of SAS Macros to generate various statistical analyses and a Visual Basic interface on the client to create formatted Microsoft Word tables from the analyses. Technologies used include:

- SAS Macros
- SAS/SHARE
- SAS ODBC Driver
- Visual Basic

A brief history of the two components of MacStat (SAS and Visual Basic) will be given, along with design considerations.

SAS MACROS

GENERAL DESIGN

It was decided early in the design phase that input analysis datasets were to be created outside of the MacStat system. This gives the user flexibility with regard to input. The only restrictions about the input dataset are:

- Treatment groups must be formatted
- An analysis variable must be supplied

Furthermore, it is recommended that all variables be labeled (the client uses the variable labels as column names).

It was also decided that running the system would be via a driver macro that would then call other necessary macros to perform the proper analysis. An example of this macro call is in Appendix A.

The parameter supplied to the macro would indicate the type(s) of statistics to be generated, input datasets, output directory, and other options depending on the type.

Output from the system consists of the SAS listing (PROC REPORT output) and output datasets with different suffixes to the base name depending on the type of analysis they contain. These datasets are for use with the client.

Until version 3.0, SAS 6.12 was used on HP/UX platform.

VERSION 1

The initial release of the MacStat macro was able to perform continuous data analysis, for example:

- Confidence intervals between treatment groups
- Descriptive statistics (including transposed)
- Overall and pairwise statistics
- Modeling (via PROC GLM)
- p-values
- Ranked transformation (Blom, Tukey, van der Waerden)
- Assumption testing (normality, variance)

VERSION 2

Version 2 added the capability to perform categorical data analysis:

- Counts tables (three layouts)
  - Treatment groups and response variables as columns
  - Treatment groups as rows, response variables as columns
  - Treatment groups as columns, response variables as rows
- Frequency counts
  - Cell count within treatment group
  - Row total within treatment group
  - Column total within treatment group
  - Grand total within treatment group
  - Totals across treatment groups
- Cell percentages
- Column total percentages
- Ability to include missing values in counts only, counts and percentages, or in neither counts nor percentages
- p-values and associated statistics
  - TABLE option CMH, CHISQ, or EXACT
VERSION 2.1
This version added enhancements to both continuous and categorical analysis capabilities.

For continuous data analysis:
• Confidence intervals on the mean within treatment groups

For categorical data analysis:
• Ability to define multiple subgroups for a single response
• Confidence intervals

VERSION 2.2
This was another incremental release focusing on adding new capabilities to continuous and categorical analysis.

For continuous data analysis:
• Crossover study designs (2 periods, 2 treatments)

For categorical data analysis:
• Overall p-values
• Derived estimations (scores option)
• Crossover study designs (2 periods, 2 treatments, McNemar’s test)
• Logistic regression modeling

VERSION 3.0
Version 3.0 was one of the most ambitious updates to the MacStat macros. It was decided to switch to SAS 8.0 at this time in order to take advantage of new features such as:
• Long variable names (more descriptive)
• Long dataset names
• ODS dataset output capability

Also, many of the macros were rewritten and restructured to make them more maintainable.

The biggest addition to this release was the addition of survival data analysis to the continuous and categorical capabilities, which received updates as well:

For continuous data analysis:
• Descriptive statistics now all calculated using PROC UNIVARIATE

For categorical data analysis:
• Relative risk and odds ratios
• Asymptotic confidence intervals for relative risk odds ratios

For survival data analysis:
• Three counts table layouts:
  o Treatment groups as columns, censoring variable as row
  o Treatment groups as column, censoring variable as row
• Frequency counts for censoring variable
• Percentages for censoring variable
• Minimum and maximum time events for censoring variable
• p-values and associated statistics (such as overall and pairwise)
• Survival estimates

VISUAL BASIC CLIENT
GENERAL DESIGN
In the beginning, the MacStat client was designed with the following features in mind:
• Ability to navigate UNIX directories and select datasets
• Ability to select datasets from the local computer
• Ability to choose column(s) to include them in report (in order selected)
• Ability to change titles and table notes from what was generated by the MacStat macros
• Ability to choose page-by variable
• Ability to change column headings and widths
• Ability to subset data
• Ability to store and load templates
• Generate Microsoft Word tables from selections with AstraZeneca desired formatting

The original incarnation of the client was not a standalone Visual Basic client as is the current version, although the “look and feel” was similar.

VERSION 2.0
This was a VBA (Visual Basic for Applications) application contained in a Microsoft Word 97 document on a Windows 95 platform. When the Word document was loaded, the VBA application would run and present the login screen and the rest of the user interface. Interface to datasets was using DAO (Data Access Objects) through the SAS ODBC driver connecting to a UNIX SAS/SHARE server.

VERSION 2.1
At this point, the size of the Word document including the VBA application became unwieldy; there were performance issues as well. The MacStat client application became a hybrid Visual Basic 5/VBA application. The user interface was handled with Visual Basic; the document creation was handled with VBA in a Word 97 document (which was much smaller now).
VERSION 2.2

This version made a break from using VBA once and for all. Everything from the user interface to document creation was done from the Visual Basic 5 client. Programmatic control of Word 97 was via OLE automation.

VERSION 2.3

This release incorporated two major changes: platform change to Windows 2000 Professional and changing to Visual Basic 6 as the development platform. The version of Microsoft Word was now Word 2000. The technology used to communicate with the SAS ODBC driver changed to ADO (ActiveX Data Objects). This change was made because it was a more updated interface to data than DAO was.

VERSION 3.0

This was a minor release compared to previous releases (the jump in version number was to keep the version number in line with the MacStat macro system). Minor interface changes were made; it now used the SAS 8 ODBC driver and UNIX SAS 8 SHARE server in order to access SAS 8 datasets generated by version 3.0 of the macro system.

CLIENT DETAILS

LOGIN PROCEDURE

When the application is run, a login screen is presented where the user can choose from a list of servers (or local) and enter their user id and password (Figure 1).

After a successful login, the user is presented with the main interface with six tabs, and viewing their UNIX home directory.

 CONNECTION TO UNIX

UNIX directories and files are retrieved via the Reflection FTP control (made by WRQ, Inc.) and placed in a list box. The list is filtered to only show directories and SAS datasets (version 6 or 8). Double-clicking on a directory (or typing in the text box directly above) changes the working directory. Double-clicking on a datasets retrieves the variables and labels for the dataset (from SASHELP.VCOLUMN) and places this information in the right list box (Figure 2).

USER INTERFACE

The user interface consists of six tabs. Clicking on each tab displays different input fields. A brief description of the functionality of each tab:

- Dataset -- initial display, choose directories, datasets, and columns to include in report
- General -- change title and table notes, page-by variable, include SAS dataset name and/or SAS program name in footnotes
- Format -- change column headings or size, group variables
- Subset -- subset values for report
- Merge -- show pairwise or overall data separately on report
- Template -- save or apply template
WORD DOCUMENT GENERATION

As previously mentioned the MacStat client uses OLE automation to control Microsoft Word. Basically, Word "exposes" some of its internal functionality to be used by external programs (such as those written in Visual Basic). After the user choose options for the report and clicks the Generate Report button, the general program flow is as follows:

- Best fit column widths are calculated based on data and column headings
- A new Word document is opened
- A Word template is loaded which contains styles specified by AstraZeneca
- The table is written
- The document is paginated

It should be noted that the client is a reporting tool only; no statistics are calculated, the report is based on datasets generated by the MacStat macros.

An example Microsoft Word document is in Appendix B.

CONCLUSION

The MacStat system design has supplied a solutions to existing problems: have statistic generation macros available so the statistician does not have to write SAS programs for this purpose (and be able to generate statistics in a consistent manner from study to study) and via selections from a graphical user interface generated a properly formatted Microsoft Word document, eliminating the need for someone to type in the data by hand and apply the formatting.

Future development of the MacStat system has various possibilities. The initial design of the macro system (and what is being used currently) has a single entry point. This was fine when there was only continuous analysis, but now that categorical and survival analyses have been added as well, the number of parameters has grown significantly. One possibility is to have three driver macros, one for each type. Another possibility is to supply a GUI front-end to the driver macros, allowing a point-and-click interface to the parameters. Possibilities for this would be SAS/AF or Visual Basic and SAS Integration Technologies.

The MacStat client is capable of reading any SAS dataset, however, for full functionality it expects certain column names for full functionality (e.g. it automatically selects the treatment variable and by variables based on name). Changes could be made to make it more general in this regard. Also, it could be made to allow for different document template and to be able to generate output other than Microsoft Word (such as HTML).

ACKNOWLEDGMENTS

SAS and all other SAS Institute, Inc. product or service names are registered trademarks or trademarks of SAS Institute, Inc. in the United States and other countries. ® indicates USA registration.

Microsoft and all other Microsoft, Inc. product or service names are registered trademarks or trademarks of Microsoft, Inc. in the United States and other countries. ® indicates USA registration.

Other brand and product names are registered trademarks or trademarks of their respective company.

CONTACT INFORMATION

Richard Thomas
Palarco, Inc.
725 Chesterbrook Blvd.
Wayne, PA 19087
rich.thomas@astrazeneca.com
610-695-4457

Nick Ronca
AstraZeneca Pharmaceuticals LP
725 Chesterbrook Blvd.
Wayne, PA 19087
nick.ronca@astrazeneca.com
options sasautos=(auto,'/users/macstat/dev');
libname library '/users/macstat/test/v2.2/formats';
libname testdata '/users/macstat/test/v2.2/data';

* Remove missing values to avoid sparse data for proc genmod.;
data test;
   set testdata.cattest;
   if sex=. or race=. or race=U or age=. then delete;
%
%analysis(
   analtype = category,
   dsn = test,
   trtment = treat,
   bystmt = race,
   catvar = binnum,
   misfmt = nopct,
   trttot = y,
   bytot = n,
   pctnfmt = n,
   subgrp = ,
   method = wald,
   baseline = ,
   stratum = ,
   keywords = ,
   aggreg = y,
   class = age,
   model = treat age,
   modeladd = treat*age,
   exactci = ,
   scores = ,
   levelci = 0.99,
   alphap = 0.1,
   style = pvalast2,
   sigp = 4,
   sigds = 3,
   title1 = Sample Document,
   title2 = ,
   prtstat = ,
   sasout = /users/macstat/test/v3.0/output/cat06
);
run;
## APPENDIX B – SAMPLE MICROSOFT WORD OUTPUT

### Sample Document

27OCT2000

<table>
<thead>
<tr>
<th>Race num</th>
<th>Response</th>
<th>a treat n (%)</th>
<th>placebo n (%)</th>
<th>z treat n (%)</th>
<th>Total n (%)</th>
<th>Overall ChiSquare</th>
<th>Overall DF</th>
<th>Overall Pr&gt;Chi</th>
</tr>
</thead>
<tbody>
<tr>
<td>White</td>
<td>N</td>
<td>588</td>
<td>827</td>
<td>787</td>
<td>2202</td>
<td>3.326</td>
<td>2</td>
<td>0.1896</td>
</tr>
<tr>
<td></td>
<td>.</td>
<td>6</td>
<td>2</td>
<td>13</td>
<td>21</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>161 (27.381)</td>
<td>264 (31.923)</td>
<td>237 (30.114)</td>
<td>662 (30.064)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>427 (72.619)</td>
<td>563 (68.077)</td>
<td>550 (69.886)</td>
<td>1540 (69.936)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Black</td>
<td>N</td>
<td>162</td>
<td>295</td>
<td>241</td>
<td>698</td>
<td>0.582</td>
<td>2</td>
<td>0.7474</td>
</tr>
<tr>
<td></td>
<td>.</td>
<td>0</td>
<td>3</td>
<td>1</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>46 (28.395)</td>
<td>74 (25.085)</td>
<td>63 (26.141)</td>
<td>183 (26.218)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>116 (71.605)</td>
<td>221 (74.915)</td>
<td>178 (73.859)</td>
<td>515 (73.782)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Other</td>
<td>N</td>
<td>20</td>
<td>30</td>
<td>11</td>
<td>61</td>
<td>1.463</td>
<td>2</td>
<td>0.4812</td>
</tr>
<tr>
<td></td>
<td>.</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>7 (35.000)</td>
<td>6 (20.000)</td>
<td>3 (27.273)</td>
<td>16 (26.230)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>13 (65.000)</td>
<td>24 (80.000)</td>
<td>8 (72.727)</td>
<td>45 (73.770)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>