It's Alive!

Creating a Living Report from Various Parts

Sue Cross, The Pennsylvania State University, University Park PA

ABSTRACT

Faculty are increasingly experimenting with web-assisted courses, new tutorials, on-line surveys, and computer-based testing. All of these create tons of data and statistics, and usually a very plain report. Multiplied by the numbers of courses, students, and semesters, the faculty member ends up with a lot of very plain reports. I am using SAS® to glue all the pieces (the data, statistics, and graphs) into one interactive display that a faculty member can access using a browser. Using SAS/IntrNet® and the REPORT procedure, the result is a report with traffic lighting, embedded graphs and hyperlinks.

This paper is not really about teaching and tests. It is about freeing data from that two-dimensional report on your desk, giving it life, and sharing it over the Internet.

INTRODUCTION

Twenty-five years ago a freshman at Penn State expected to sit in a classroom copying notes from the blackboard, write a paper, take a midterm and final exam. A machine may have scored the exams and produced a report with pages of statistics and test scores, and the professor probably copied a grade for each student into his grade book.

Today a freshman brings a computer to campus, and is likely to participate in a computer-assisted course, take advantage of an on-line tutorial and multiple assessments, communicate on-line, and take some computer-based tests. The professor is e-mailed a report with pages of statistics and test scores, and he copies a grade for each student into his grade book. Increasingly, professors are entering scores into a spreadsheet and asking interesting questions that the spreadsheet can't answer. They ask, " Have the students mastered this material?", "Have I asked the right questions?", and "Have I done a good job teaching?".

THE DOWNSIDE OF EMERGING TECHNOLOGIES

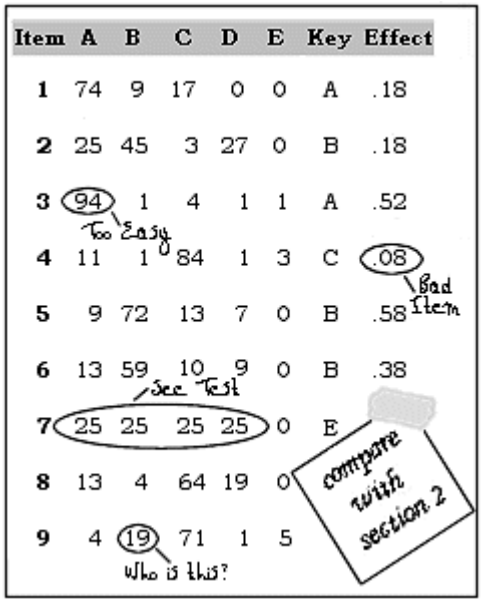
The problem is that there is so much information available in printouts, files, databases, and spreadsheets that the faculty member doesn't know where to look for answers. He can look at the columns and rows of several reports, make notes in the margins, highlight statistics, or check suspicious results. But, he needs help finding the right information, sifting through it, and making the right connections and conclusions.

Meanwhile, information sources are changing as databases are built, data are warehoused or the latest release of "Nicheware" is installed. Who can take the time to rewrite production jobs to react to emerging technologies?

I am using the SAS System to glue the new data sources, statistical methods and reports to the old. I envision a modular program with a data step that can be modified to read data from the latest version of software, surveys, or files; appropriate statistical procedures to analyze the data; and a main report which makes the connections between the data sources and provides drill-down capabilities that can be used interactively to both ask and answer the right questions. In this paper, I show how I plan to use SAS/IntrNet and PROC Report to present the results of statistical procedures, link graphs, and indicate strange and interesting results.

ONE SCENARIO

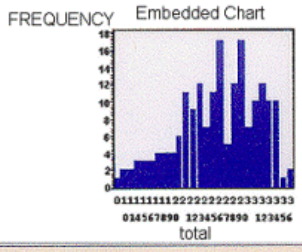
The professor teaching Anatomy 101 receives a report called an item analysis after students have taken the first test. The report is a spreadsheet with one row for each item and columns indicating the percent of students answering A-F, the correct answer, and the item effect. He makes some notes in the margins to edit some questions, check on some students, and wonders how these students compare with those in section 2. He locates the distribution of test scores for these students on page 27 of a printout, and on page 24 in another printout for section 2. He cuts and pastes these under the report and sees that the scores are quite different, with the average score in section 1 about 10% higher than in section 2. He finds the tests and sees item 3 was misprinted in the section 1 test. He'd like to delete item 3 and recalculate the distribution of scores. But he can't, because this is just a collage of printouts and Post-it® Notes.



THE UPSIDE OF EMERGING TECHNOLOGIES

This report can be brought to life by gluing the distribution of test scores on top of the page, and making connections to important pieces of information using embedded hyperlinks in any cell in the table. PROC Report creates the body of the report in which the item can be linked to the question in the item database, graphs can be linked to each cell in one column, student performance measures can be linked to the cells in another column, results from another section or test can be compared, and the output from statistical procedures can be selected. The professor can access the report using a browser, and can recreate it by deleting bad items, or selecting different column measures on the web form. The lifeless report is thus transformed into an interactive decision-making tool produced by an ever-evolving program. At this writing, the report looks like the one in the following figure (only larger). The distribution of test score is the " Embedded Chart" at the top of the page, and the "Embedded Hyperlink" is connected to another chart. The values in the last column are red, yellow, or green (like a traffic light). Red indicates that the students correctly answering this item

scored low on the test as a whole (this is a bad thing). A professor might want to follow an embedded hyperlink for these items.



Item Analysis

Test	Item	Difficulty	(std)	N	Discrimination
Embedded Hyperlink	1	0.44	0.50	184	0.21
	2	0.45	0.50	184	0.38
	3	0.69	0.46	184	0.51
	4	0.42	0.49	184	0.25

The barebones code used to create this report, including the charts and HTML file, is listed on the following pages. With a few changes to dataset and variable names, you can copy the program to bring your financial reports or medical records to life in a SAS IntrNet application, too.

AN EMBEDDED HYPERLINK

This code produces the chart that appears on the screen when you click "Embedded Hyperlink" in the first column labeled "Test". The data steps and statistical procedures that precede this are deleted here.

```
ods listing close;
ods htmlfile=chart1a.html style=styles.d3d
  path=&_tmpcat (url=&_replay);

*----the HTML device driver creates an
index.html and a gif for every graph produced
(Patel & Caira 1999);

goptions device=gif
  hsize=5in vsize=5in
  ftext=swiss htext=2 htitle=2 ;
pattern1  color=red;


proc gchart data=scored; /*change name*/
  vbar total /*change name*/
   /discrete frame;
   title1 "Embedded Hyperlink";

*---- &_tmpcat.chart1a.html points to the gif
image in  &_tmpcat.gchart.gif ;
*--- This gchart output also goes to
&tmpcat.gseg with the graph being named GCHART
by default;
run;
```

My intention is to create many charts with different views of the data and analyses, and with drill-down capabilities. They will be available to the professor with a click of the mouse as he follows a trail to explore the test results.

AN EMBEDDED CHART

This code produces the "Embedded Chart" that appears at the top of the page. I'm using this chart to profile the test distribution

```
goptions  device=gif nodisplay
  hsize=2in vsize=2in ftext=swiss;
  pattern1  color=blue;

proc gchart data=scored /*change name*/
   gout=gseg;
  vbar total  /*change name*/
  /discrete frame;
  title1 "Embedded Chart";
  run;

*----gchart output goes to &_tmpcat.gseg
catalog, gout= defines a temporary catalog name
(the default is work.gseg and type=grseg),
graphs are named gchart,gchart1,gchart2. So,
this chart is referred to as gseg.GCHART1.
```

PROC REPORT

The REPORT procedure is the backbone of this application. You can include data from any statistical procedure in the body of the report, connect any cell in the report to a graph, and use traffic lighting or other styles to draw attention to interesting or significant results. You can view a graph by clicking the embedded hyperlink, return to the body of the report, and then view a different graph.

To create this functionality, you only need to know where to find the various parts you want to include in the body of the report or its embedded hyperlinks. Once you find the parts, it is easy to bring the application to life using compute blocks in the Report procedure.

```
 ods html  body=_webout (dynamic) rs=none
path=&_tmpcat (url=&_replay);
---- Application Dispatcher generates a unique
catalog name in the WORK library and stores this
two-level catalog name in the macro variable
&_tmpcat;

proc report data=XXX_table1 /*change name*/
  nowd
  style(REPORT)={background=grey}
  style(HEADER)={background=blue
  foreground=white}
  style(COLUMN)={foreground=white}
  style(LINES)={foreground=black}
  style(SUMMARY)={foreground=pink};
  column Test Item col1 col2 col3 col8;
  /* change these column names for you app.*/
 /* column names are input from PROC CORR */
  define Test/ group
    style(COLUMN)={foreground=red};
  define Item/ group
    style(COLUMN)={foreground=blue};
  define col1/ analysis sum 'Difficulty'
    format=5.2;
  define col2/ analysis sum '(std)' format=5.2;
  define col3/ analysis sum 'N' ;
  define col8/ analysis sum 'Discrimination'
    format= 5.2
       style(HEADER)={foreground=white};

   *---Compute block for embedded chart
   found in row before page;
     compute before _page_/
     style={foreground=pink background=oldlace
      font_weight=bold} left;
     call define (_ROW_,"GRSEG","gseg.GCHART1");
     line 'Item Analysis';
     endcomp;

   *--Compute block for traffic lighting
            in col8 (Discrimination);
```

```
compute col8;
  if col8.sum <.30 then call define(
   _COL_,'STYLE','style={foreground=red}');
  if col8.sum >.50 then call define(
   _COL_,'STYLE','style={foreground=green}');
  if .30 <= col8.sum <.40 then call define(
  _COL_,'STYLE','style={foreground=yellow}');
  endcomp;

*-- Compute block for embedded hyperlink
               in test column;
compute test;
  urlstring="chart1a.html";
  /*this file points to &_tmpcat.gchart.GIF*/
  call define (_COL_,'URLBP',urlstring);
  endcomp;

run; /*** Run report ***/

ods html close;
ods listing ;
quit;
```

I plan to use embedded hyperlinks in the "Item" column to show the actual question coming from an item database, in the "Difficulty" column to cross tabulate student characteristics by difficulty categories, and in the "Discrimination" column to show the probability of choosing each response choice to this question by test score.

HTML FILE

This creates the web form from which you will be able to make selections (e.g., course number, section, and statistics) to include in the report. The selections have been deleted for the purposes of this paper. As written here, the form created by this file serves only to submit the SAS job. When complete, it will allow a professor to view the report and graphs, go back to the web form to make modifications (e.g., delete test items), and rerun the procedure.

```
<HTML>
<HEAD>
 <TITLE>Test and Item Analysis Program by S. Cross</title>
</HEAD>
<BODY>
<H1>Item Analysis</h1>
<H2>This is a prototype last revised June 15, 2001</h2>
<I>The purpose of this program is to demonstrate how to include
embedded graphs, embedded hyperlinks, and traffic lighting in a
report.
</I>
<FORM ACTION="http://www.sas.psu.edu/cgi/broker.cgi">
<!-- Change this line for your site -->

<HR>
(selections are deleted)<BR>
<INPUT TYPE=HIDDEN" NAME="_PROGRAM"
VALUE="slc.scan2r1c.sas">
<!-- Change this line for your site –->

<INPUT TYPE="HIDDEN" NAME="_DEBUG" VALUE="131">
<hr>
<INPUT TYPE="SUBMIT" VALUE="Run Procedure">
</form>
</body>
</html>
```

This is the web form:

Item Analysis

This is a prototype last revised June 15, 2001

The purpose of this program is to demonstrate how to include embedded graphs, embedded hyperlinks, and traffic lighting in a report.

(selections are deleted)

Run Procedure

CONCLUSION

I have been experimenting with SAS/IntrNet to improve data analysis and reporting of test and survey results at the Pennsylvania State University because it is just too cool to ignore. Although I have only written a few prototypes, I believe this technology adds value to the reports that professors receive, and is flexible enough to handle just about any data source or analysis that emerges.

Having little experience with web applications (HTML or CGI), I have been pleased with the ease of creating reports on the web with the SAS System. The biggest obstacle has been figuring out where my graphs go in cyberspace, and how to refer to them. At this point, I am not sure how elaborate this SAS/IntrNet application can become, or what security issues have to be overcome.

I hope my example will be enough to get you started so that you can bring your own reports to life. As I develop my own testing applications, I will include them on the website listed below.

REFERENCES

Patel, Himesh, and Caira, David (1999), "Using SAS/GRAPH Software to Create Graphs on the Web", in the Proceedings of the Twenty-Fourth Annual SAS Users Group International Conference, Paper 189.

Post-it® is a registered trademark of 3M.

SAS® and SAS/IntrNet® are registered trademarks of SAS Institute, Inc.

CONTACT INFORMATION

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

Sue Cross
The Pennsylvania State University
23 Willard Building
University Park PA 16802
(814) 863-2598
slc@psu.edu
http:// www.personal.psu.edu/~slc