

Customized Report Writing Using PROC PRINTTO

Kim L. Kolbe Ritzow of Systems Seminar Consultants, Kalamazoo, MI

Abstract

This paper will show how to take select information from the various SAS procedure's output (like PROC FREQ, PROC TABULATE, etc.) and combine the information into a single "customized" report by using PROC PRINTTO to capture the procedural output, the DATA Step to read in the external flat file that PROC PRINTTO creates, and the various DATA Step tricks and techniques used to put the information back together into a single "customized" report.

Background

Sometimes, no matter how hard we try and no matter how good of a SAS programmer we may be, we cannot seem to get a report looking exactly the way we want it.

Occasionally we may desire to have the best of both worlds: the ease of getting automatic and default output from the various SAS procedures with a minimal amount of effort, combined with the flexibility to do anything we want with our reports through customized report writing within the DATA Step. Truth is, the best of these two worlds does exist- it just takes a little bit of ingenuity and a few tricks to make it happen.

Problem

A user comes to me with the default reports from some SAS procedures (in my case, a PROC REG and a PROC TABULATE, but it could be output from any SAS procedure(s)). On these particular reports they have made some notes as to how they would ideally like their "ideal" report to look (see Tables 1-3 at the end of the paper). They would like to take a little bit of information from one report and cut and paste it with some information from another report and add to it some customized information. Your first inclination may be to think, "NO! It can't be done", or, "Yeah, I can do it, but it's going to take a LOT of programming because I'm going to have to do it all from scratch through customized report writing within the DATA Step". Actually, it's not really all that difficult to do when you know a little bit about PROC PRINTTO.

PROC PRINTTO

PROC PRINTTO provides the ability to specify a destination for the SAS LOG or procedural output. In this particular example, I am going to use PROC PRINTTO to save the output from my SAS procedure out to disk, use the DATA Step to turn around and read the report in from disk as if it were a flat file, and use a series of special input statements to help search for particular sequences of information from which a "customized" report will be built.

An example of how a procedure's output can be captured with PROC PRINTTO:

```
FILENAME OUTFILE 'C:\OUTFILE.TXT';
```

```
PROC PRINTTO NEW PRINT=OUTFILE;  
RUN;
```

```
PROC REG;  
  MODEL RESULT=CONC;  
RUN;
```

```
PROC PRINTTO;  
RUN;
```

The first PROC PRINTTO coded turns on the capturing facility. All output from this point on will be routed to the specified destination named on the PRINT= option until another PROC PRINTTO has been coded, which will turn off the capturing facility and terminate the process.

The NEW option on the PROC PRINTTO statement will clear any previous information in the destination file. Without this option, information would be appended on to the end of the specified file. I strongly recommend that you use this option, otherwise you may end up searching through a lot more information than you anticipated in the next step of this process where the file is then read back into a DATA Step.

While there are many different ways in which one can go about building the "customized" report from this point on, I chose to initially capture just the output from PROC REG with PROC PRINTTO.

Then, with a `_NULL_ DATA` Step, I went back through the file picking off the information that I needed and made their values into macro variables to be used when I'm building the report (see Table 4 for actual SAS code).

Another Example

Another user wants to combine the output from a PROC FREQ with some output of a PROC UNIVARIATE and add some customized information. In particular, PROC UNIVARIATE contains a graph that they would like to have present in the final report (see Table 5-7). No Problem! The graph will be treated like any other line of text (see Table 8 for actual SAS code).

Similar to the previous problem, I simply capture the output from one of the procedures and create some macro variables from the information to be used in the customized report. In order to accommodate for the graph, I search until I find the word "Normal" for "Normal Probability Plot" which is the heading at the top of the graph and then use an INPUT statement within a DO loop, outputting a record each time until I reach the end of the file. All other lines not relating to the graph are deleted after the macro variables have been created. Note in this example it was important to build a SAS data set as output because it's going to contain the graph. I then capture the output from the second procedure, a PROC FREQ. Because I would like the output on the final report from PROC FREQ centered and the output from PROC UNIVARIATE to be left aligned, I turned the centering off for PROC UNIVARIATE, but turned it back on for PROC FREQ. The centering was turned back off before writing out the customized report because I didn't want the titles centered.

In order to create the final report, I read in the flat file from PROC FREQ and wrote out each record until end of file. After reaching end of file on the output from PROC FREQ, I then wrote out some descriptive statistics using the macro variables created in a previous step. In order to write the graph out, I use a SET statement nested within a DO loop until it reaches the end of file indicator. When that's complete I add some more descriptive information and the report is done!

Special Hints When Building Customized Reports With PROC PRINTTO

- ❑ it's a good idea to always turn off titles and footnotes before capturing a report and pay particular attention to any options that may be in effect (CENTERING, DATE, etc.)
- ❑ it's a good idea to always use the NEW option on PROC PRINTTO to prevent reports appending themselves on one another in the output destination file
- ❑ try not to capture too much information in a single PROC PRINTTO, it will make searching through the file in subsequent steps more difficult
- ❑ re-use output flat files as much as possible on PROC PRINTTO to save resources
- ❑ code flexible input statements (don't use column or pointer references)
- ❑ avoid using list input because it has some severe limitations that can cause problems when trying to read the report back into the DATA Step
- ❑ one of the best ways I've found to read in this type of data is to treat it as a variable length field and then use a series of functions to strip off the information
- ❑ because the SCAN function is case sensitive the UPCASE function is often used in conjunction with it to help assure perfect matches
- ❑ when using the SCAN function you may have to specify a third parameter to only use blanks as a valid delimiter
- ❑ do not use the NOPRINT option on those procedures that support it (like MEAN, FREQ, etc.), otherwise the PRINTTO will not capture any output

- ❑ PRINTTO will only capture procedural output, not output SAS data sets
- ❑ several more functions like PUT, INPUT, LEFT, TRIM, COMPRESS, INDEX, INDEXC, etc. might need to be used to make the code more flexible and bullet proof
- ❑ macros can be incorporated to build in even more flexibility
- ❑ the more general the scan through process is coded, the easier it will be to maintain the code in the long run- keep in mind version changes and platform changes may affect the code!
- ❑ this is a great technique to use when you need statistics from a SAS procedure than cannot be obtained on an output SAS data set (ie: quantiles)
- ❑ also useful in creating macro variables for annotating a SAS graph
- ❑ maintains data integrity by letting the procedures do all the calculations and with a minimal amount of work, create a customized report

In Summary

PROC PRINTTO, in addition to the use of customized reporting tools can help to create some unique and otherwise difficult reports, in a relatively quick and easy manner.

Programs developed using PROC PRINTTO as an alternate means for customized report writing may require a little more maintenance between releases and platforms of SAS if the structure of the report should change than if the report were written strictly from scratch within the DATA Step. However, this is sometimes outweighed by the fact that if a majority of the information was derived from the output of a SAS procedure it tends to be more reliable than if you were to generate it all yourself. Obviously, the more general you can code the syntax for the scan through process, the more bullet proof and easier to maintain the code will be.

Related Literature

The PRINTTO Procedure, BASE SAS Procedures Guide Version 6 Third Edition, pgs. 483-492.

The PRINTTO Procedure (how to route log or output files to catalog entries), SAS Technical Report P-222 Changes and Enhancements to BASE SAS Software, Version 6.07, pgs. 261-265.

Trademark Notice

SAS is a registered trademark of the SAS Institute Inc., Cary, NC, USA and other countries.

Any questions or comments regarding the paper may be directed to the author:

Kim L. Kolbe Ritzow
Systems Seminar Consultants
Kalamazoo Office
927 Lakeway Drive
Kalamazoo, MI 49001
Phone: (616) 345-6636
Fax: (616) 345-5793
E-mail: KRITZOW@AOL.COM

Model: MODEL1
Dependent Variable: RESULT

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Prob>F
Model	1	4224476.4575	4224476.4575	1466.742	0.0001
Error	19	54723.35203	2880.17642		
C Total	20	4279199.8095			
Root MSE	53.66728	R-square	0.9872		
Dep Mean	458.09524	Adj R-sq	0.9865		
C.V.	11.71531				

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	T for H0: Parameter=0	Prob > T
INTERCEP	1	62.314059	15.61882273	3.990	0.0008
CONC	1	1.168974	0.03052305	38.298	0.0001

Table 1

	RESULT				
	N	MEAN	STD	VAR	CV
CONC					
10	3	25.67	6.43	41.33	25.0
20	3	52.00	3.46	12.00	6.7
60	3	94.00	4.58	21.00	4.9
120	3	311.00	1.00	1.00	0.3
360	3	511.67	8.08	65.33	1.6
720	3	909.67	12.50	156.33	1.4
1080	3	1302.67	2.52	6.33	0.2

Table 2

Concentration Study

I. Descriptive Statistics.

	RESULT				
	N	MEAN	STD	VAR	CV
CONC					
10	3	25.67	6.43	41.33	25.0
20	3	52.00	3.46	12.00	6.7
60	3	94.00	4.58	21.00	4.9
120	3	311.00	1.00	1.00	0.3
360	3	511.67	8.08	65.33	1.6
720	3	909.67	12.50	156.33	1.4
1080	3	1302.67	2.52	6.33	0.2

Analysis Results.

A. Prediction Equation: $62.314059 + 1.168974 * (\text{concentration})$

B. Summary of fit:

1. Mean of result: 458.09524
2. Root MSE: 53.66728
3. R-squared: 0.9872

C. Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	T for H0: Parameter =0	Prob > T	Sign.
INTERCEP	1	62.314059	15.61882273	3.990	0.0008	Yes
CONC	1	1.168974	0.03052305	38.298	0.0001	Yes

Table 3

FILENAME OUTFILE 'C:\OUTFILE';	*point to external flat file ;
OPTIONS NOCENTER NODATE;	*whenever using PRINTTO its a good idea to turn off centering and suppress the date for ease of scanning later on ;
TITLE;	*also a good idea to turn off title;
PROC PRINTTO NEW PRINT=OUTFILE;	*turn on capture facility ;
RUN;	
PROC REG;	*capture output of regression ;
MODEL RESULT=CONC;	
RUN;	
PROC PRINTTO;	*turn off capture facility ;
RUN;	
DATA _NULL_;	*no need to build SAS data set ;
INFILE OUTFILE LENGTH=LRECL;	*find the length of each rec. read;
INPUT LINE \$VARYING200. LRECL;	*reads a varying length field ;
IF SCAN(LINE,1)='Root' THEN	
DO;	*search string is case sensitive ;
CALL SYMPUT('ROOTMSE',SCAN(LINE,3,' '));	*assign macro var. values ;
CALL SYMPUT('RSQUARE',SCAN(LINE,5,' '));	*third parameter on scan ;
END;	*function tells SAS only use blanks as valid delimiters ;
ELSE IF SCAN(LINE,1)='Dep' THEN	
CALL SYMPUT('DEP_MEAN',SCAN(LINE,3,' '));	
ELSE IF SCAN(LINE,1)='INTERCEP' THEN	
DO;	
CALL SYMPUT('INTERCPT',SCAN(LINE,3,' '));	
CALL SYMPUT('INTERLNE',LINE);	
IF SCAN(LINE,6,' ') < .05 THEN	
CALL SYMPUT('INTERPRB','YES');	
ELSE CALL SYMPUT('INTERPRB','NO');	
END;	
ELSE IF SCAN(LINE,1)='CONC' THEN	
DO;	
CALL SYMPUT('SLOPE',SCAN(LINE,3,' '));	
CALL SYMPUT('SLOPELNE',LINE);	
IF SCAN(LINE,6,' ') < .05 THEN	
CALL SYMPUT('SLOPEPRB','YES');	
ELSE CALL SYMPUT('SLOPEPRB','NO');	
END;	
RUN;	*write macro vars out to make sure I got the right values ;
%PUT ROOTMSE=&ROOTMSE RSQUARE=&RSQUARE DEP_MEAN=&DEP_MEAN	
INTERCPT=&INTERCPT INTERLNE=&INTERLNE SLOPE=&SLOPE SLOPELNE=&SLOPELNE	
INTERPRB=&INTERPRB SLOPEPRB=&SLOPEPRB;	

Table 4

```
PROC PRINTTO NEW PRINT=OUTFILE;
RUN;
```

*re-use output file for effic.;

```
PROC TABULATE;
  CLASS CONC;
  VAR RESULT;
  TABLE CONC,
    (RESULT) * (N*F=8.0 MEAN*F=8.2 STD*F=8.2 VAR*F=8.2 CV*F=8.1)/RTSPACE=10;
RUN;
```

*capture output from tabulate ;

```
PROC PRINTTO;
RUN;
```

*turn off capture facility ;

```
DATA _NULL_ ;
  LENGTH LINE1 LINE2 $70;
  INFILE OUTFILE END=EOF LENGTH=LRECL;
  INPUT LINE $VARYING200. LRECL;
  LINE1=&INTERLNE";
  LINE2=&SLOPELNE";
  FILE PRINT;
  TITLE2;
  TITLE3;
  TITLE4 "CONCENTRATION STUDY";
  TITLE5;
  TITLE6 "1. DESCRIPTIVE STATISTICS.";
  PUT LINE $CHAR80.;
  IF EOF THEN
    DO;
      PUT //;
      PUT "ANALYSIS RESULTS." /;
      PUT " A. PREDICTION EQUATION: &INTERCPT + &SLOPE * (CONCENTRATION)" /;
      PUT " B. SUMMARY OF FIT:" /;
      PUT "      1. MEAN OF RESULT:      &DEP_MEAN" /;
      PUT "      2. ROOT MSE:              &ROOTMSE" /;
      PUT "      3. R-SQUARED:             &RSQUARE" /;
      PUT " C. PARAMETER ESTIMATES" //;
      PUT "      PARAMETER      STANDARD      T FOR HO:";
      PUT "VARIABLE DF ESTIMATE      ERROR PARAMETER =0  PROB > |T|  SIGN." /;
      PUT LINE1 $1-70 " &INTERPRB";
      PUT LINE2 $1-70 " &SLOPEPRB";
    END;
RUN;
```

*no need to build SAS data set ;

*trim up length of vars for eff;

*read in output from tabulate ;

*read in varying length record ;

*assign values to variables ;

*route results to output dest. ;

*drop down some lines, could ;

*have also done it with a PUT ;

*writes out line by line the tabulate output;

*when tabulates done, add on ;

Table 4 Cont'd

GENDER	Frequency	Percent	Cumulative Frequency	Cumulative Percent
f	32	47.1	32	47.1
m	36	52.9	68	100.0

Table 5

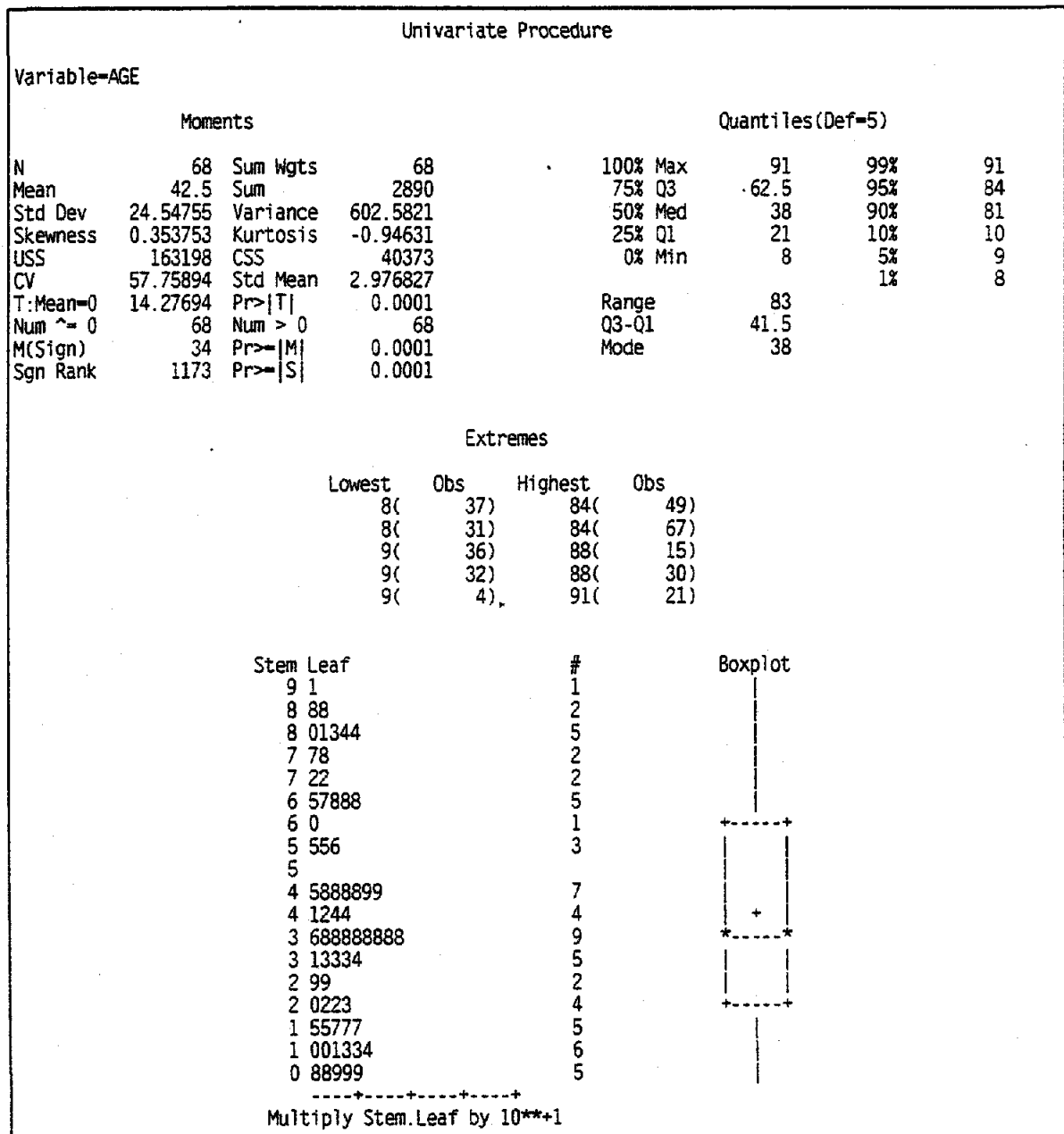


Table 6

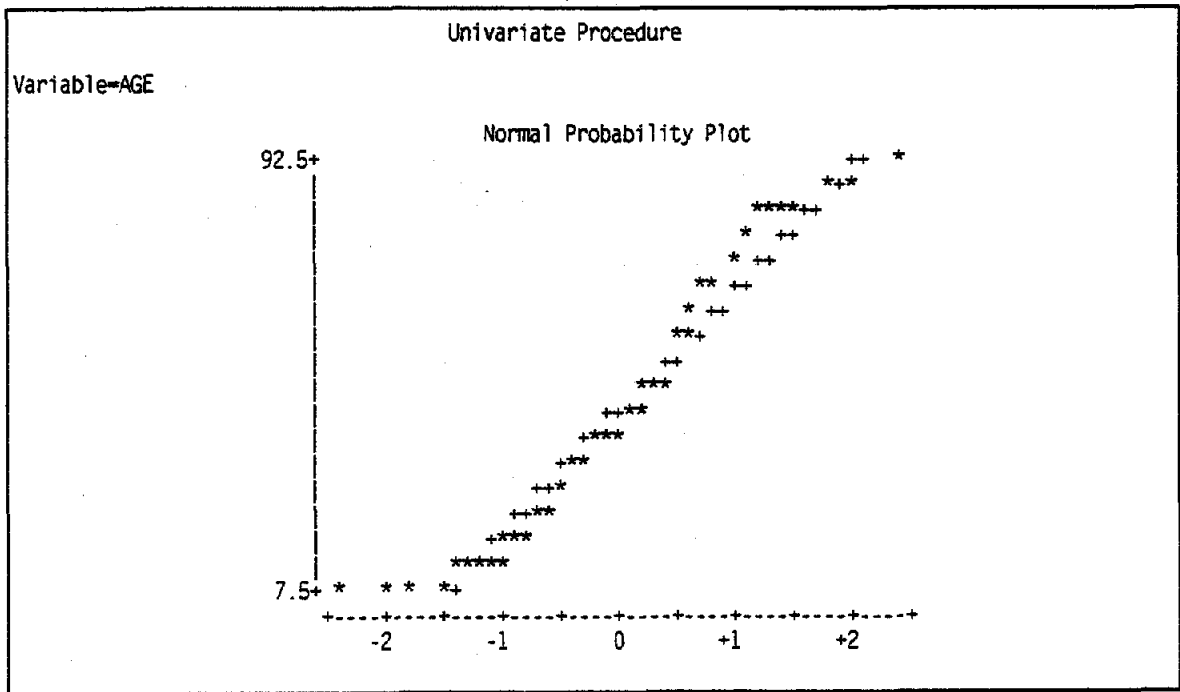


Table 6 Cont'd

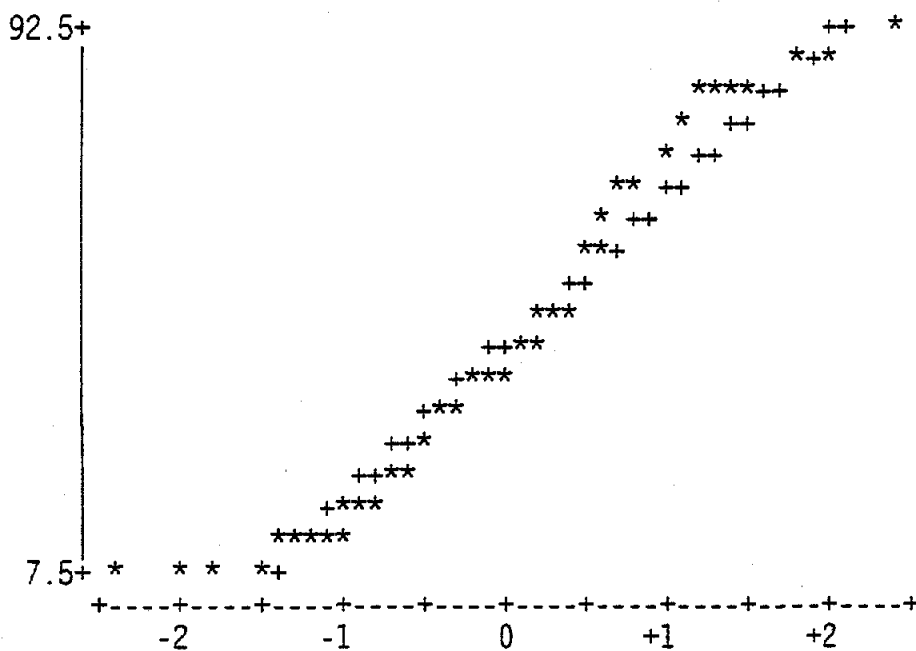
STUDY BALANCE VERIFICATION REPORT

GENDER	Frequency	Percent	Cumulative Frequency	Cumulative Percent
f	32	47.1	32	47.1
m	36	52.9	68	100.0

I. Descriptive Statistics:

Total Number of Non-Missing Values: 68
 Sample Standard Deviation : 24.54755
 Sample Variance : 602.5821

II. Normal Probability Plot



III. Distribution Statistics

Pr>|T| : 0.0001
 Skewness : 0.353753
 Kurtosis : -0.94631

Prepared by the Statistics and Business Research Dept.

Table 7

```

FILENAME OUTFILE 'C:\OUTFILE.TXT';

OPTIONS NODATE NOCENTER;

TITLE;

PROC PRINTTO NEW PRINT=OUTFILE;
RUN;

PROC UNIVARIATE DATA=DEMO PLOT;
RUN;

PROC PRINTTO;
RUN;

DATA PLOTDATA;
  INFILE OUTFILE LENGTH=LRECL END=EOF;
  INPUT LINE $VARYING200. LRECL @;
  IF SCAN(LINE,1)='N' THEN
    CALL SYMPUT('NOBS',SCAN(LINE,2,' '));
  ELSE IF SCAN(LINE,1)='Mean' THEN
    CALL SYMPUT('MEAN',SCAN(LINE,2,' '));
  ELSE IF SCAN(LINE,1)='Std' THEN
    DO;
      CALL SYMPUT('STDDEV',SCAN(LINE,3,' '));
      CALL SYMPUT('VAR',SCAN(LINE,5,' '));
    END;
  ELSE IF SCAN(LINE,1)='Skewness' THEN
    DO;
      CALL SYMPUT('SKEW',SCAN(LINE,2,' '));
      CALL SYMPUT('KURT',SCAN(LINE,4,' '));
    END;
  ELSE IF SCAN(LINE,1)='T:Mean=0' THEN
    CALL SYMPUT('PRB_T',SCAN(LINE,4,' '));
  ELSE IF SCAN(LINE,1)='Normal' THEN
    DO UNTIL(EOF);
      INPUT LINE $VARYING200. LRECL;
      OUTPUT;
    END;
  ELSE DELETE;
RUN;

%PUT NOBS=&NOBS MEAN=&MEAN STDDEV=&STDDEV VAR=&VAR SKEW=&SKEW
      KURT=&KURT PRB_T=&PRB_T;

```

Table 8

```

OPTIONS NODATE CENTER;

PROC PRINTTO NEW PRINT=OUTFILE;
RUN;

PROC FREQ DATA=DEMO;
  TABLE GENDER;
RUN;

PROC PRINTTO;
RUN;

OPTIONS NOCENTER;

DATA _NULL_;
  INFILE OUTFILE LENGTH=LRECL END=EOF;
  INPUT LINE $VARYING200. LRECL;
  FILE PRINT;
  TITLE1 'STUDY BALANCE VERIFICATION REPORT';
  TITLE2;
  TITLE3;
  PUT LINE $CHAR80.;
  IF EOF THEN
    DO;
      PUT 'I. DESCRIPTIVE STATISTICS:' /;
      PUT "  TOTAL NUMBER OF NON-MISSING VALUES:  &NOBS";
      PUT "  SAMPLE STANDARD DEVIATION      :  &STDDEV";
      PUT "  SAMPLE VARIANCE                :  &VAR" //;
      PUT 'II. NORMAL PROBABILITY PLOT' //;
      DO WHILE(EOF2 NE 1);
        SET PLOTDATA END=EOF2;
        PUT LINE $CHAR80.;
      END;
      PUT 'III. DISTRIBUTION STATISTICS' /;
      PUT "  PR>|T|      :  &PRB T";
      PUT "  SKEWNESS    :  &SKEW";
      PUT "  KURTOSIS    :  &KURT" //;
      PUT "PREPARED BY THE STATISTICS AND BUSINESS RESEARCH DEPT.";
    END;
  RUN;

```

Table 8 Cont'd