

A Winning Combination: Generation of Listing and Descriptive Statistics Table in One Report

Chenille Lloyd, PharmaNet/i3, Cary, NC

ABSTRACT

Presenting clinical trials data clearly, but from more than one perspective can be very helpful to understanding the safety and efficacy of a new drug. This paper shows how to generate a report combining an individual subject listing with a descriptive statistics table, using only SAS®/MACRO language and other Base SAS features. Such a presentation is particularly useful for pharmacokinetic analysis studies, which require easy identification of outliers and trends in drug concentration data. The macro features options that allow the programmer to specify subject exclusions and the number of decimal places used to report results.

MACRO OVERVIEW

This macro is designed for presenting an individual listing and descriptive statistics table in the same report. To successfully use this macro, the following dataset is required: &INDATA which is the input dataset having the variables that will be used to generate the columns of the table.

Input dataset may need to be manipulated before calling the macro so that it is in the correct format of one observation per subject. The macro does not currently include the capability to check the format of the data. The sample data presented in Table 1 represents a typical pharmacokinetic parameter dataset for a Phase 1, three-way cross-over study.

For pharmacokinetic analysis, there will often be requests for particular subjects to be excluded from descriptive summarization. This macro allows the programmer to specify the subject ID of subject(s) to be excluded. Please note that the specified subjects will be excluded from the summary statistics, but will still be presented in the listing.

Summary statistics will be generated for the continuous variables in the input dataset so these variables must be named according to how many decimal places will be presented in the output. Last character of variable name designates decimal place precision, up to 5 decimal places. For example, if we want to see CMAX presented to 2 decimal places, the variable should be named cmax_2. Variable names cannot be greater than 8 characters.

Future enhancements could include formatting results based on significant digits instead of decimal places.

OUR RAW DATA

SUBJID	PERIOD	TRT	ANALYTE	AUCINF
1123	1	Treatment C	Analyte 1	9774.801
1123	2	Treatment A	Analyte 1	24285.600
1123	3	Treatment B	Analyte 1	24542.120
1168	1	Treatment C	Analyte 1	8934.827
1168	2	Treatment A	Analyte 1	23872.170
1168	3	Treatment B	Analyte 1	21131.390
1172	1	Treatment A	Analyte 1	31553.900
1172	2	Treatment B	Analyte 1	26097.270
1172	3	Treatment C	Analyte 1	10564.280

Table 1: Raw input dataset with one observation per treatment.

SUBJIDC	TRTA_1	TRTB_2	TRTC_3
1123	24285.600	24542.120	9774.801
1168	23872.170	21131.390	8934.827
1172	31553.900	26097.270	10564.280

Table 2: Input dataset (after manipulation) with one observation per subject.

MACRO PARAMETERS

Parameter	Description	Mandatory/Optional
INDATA	Name of input dataset that has the variables required for the listing/table	Mandatory
SUBJVAR	Name of subject identifier	Mandatory, variable must be character format
SUBJEXC	List subject identifier values to be included in listing and excluded from descriptive statistics table; multiple subject IDs to be separated by blank space	Optional, if no subjects are excluded then SUBJEXC should be kept blank
LISTORDN	Name, order and precision of continuous variables from input dataset	Mandatory
OUTDATA	Name of output dataset including combined listing and table data that can be fed directly to proc report	Mandatory

Table 3: Macro parameter descriptions

EXAMPLE OF HOW TO CALL MACRO

```
%mindstat (INDATA=inf,
           SUBJVAR=subjdc,
           SUBJEXC=1168,
           LISTORDN=trta_3 trtb_2 trtc_1,
           OUTDATA=outpk);
```

The above example macro call will output a SAS dataset named OUTPK that can be used in proc report to generate a customized listing/summary report.

MACRO OUTPUT DATASET

CHARVAR	SUBORDER	Statistic	# of Decimal Places
N	1	Number of subjects	0
Mean	2	Mean	+1
gm	3	Geometric Mean	+1
SD	4	Standard Deviation	+2
SE	5	Standard Error	+2
cv	6	Coefficient of Variation	2
Minimum	7	Minimum	Same # of decimal places
p25	8	25 th percentile	+1
Median	9	Median	+1
p75	10	75 th percentile	+1
Maximum	11	Maximum	Same # of decimal places

Table 4: Precision and order of each summary statistic in output dataset

Variable	Description
CHARVAR	Character variable populated for listing and table output; for listing subject id is presented; for table summary statistics label is presented
NUMVARx	Character variable populated for listing and table output and formatted to decimal place precision specified in macro call; for listing actual values are presented; for table statistics are presented. X corresponds to order of variables in output dataset
SUBORDER	Numeric variable populated for table output only; gives order of summary statistics in table
PAGENUM	Numeric variable populated for listing and table output; flags listing output from table output; can be used to separate listing and table sections in proc report by line or by page

Table 5: Output dataset variables

CHARVAR	NUMVAR1	NUMVAR2	NUMVAR3	SUBORDER	PAGENUM
1123	24285.6	24542.12	9774.801		1
1168	23872.2	21131.39	8934.827		1
1172	31553.9	26097.27	10564.280		1
N	2	2	2	1	2
Mean	27919.75	25319.695	10169.5405	2	2
gm	27682.22	25307.752	10161.8765	3	2
SD	5139.464	1099.6571	558.24595	4	2
SE	3634.150	777.5750	394.73950	5	2
cv	18.41	4.34	5.49	6	2
Minimum	24285.6	24542.12	9774.801	7	2
p25	24285.60	24542.120	9774.8010	8	2
Median	27919.75	25319.695	10169.5405	9	2
p75	31553.90	26097.270	10564.2800	10	2
Maximum	31553.9	26097.27	10564.280	11	2

Table 6: Output dataset with listing of one observation per subject and summary statistics

CONCLUSION

This paper has described how a macro can combine listing and table output into one report. This combined report allows for more efficient cross-checking of data points and contributes to a more automated programming process overall. The output dataset generated by the macro easily feeds into the SAS report procedure where the report formatting can be customized for the particular clinical trial.

REFERENCE

- SAS/BASE Software: Version 9, SAS® Institute Inc., Cary NC
- SAS/MACRO Software: Version 9.1.3, SAS® Institute Inc., Cary NC

CONTACT INFORMATION

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

Name: Chenille Lloyd
 Enterprise: PharmaNet/i3
 Address: 1001 Winstead Drive, Suite 200
 City, State ZIP: Cary, NC 27513
 Work Phone: 919-677-5985
 Email: clloyd@pharmanet-i3.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration. Other brand and product names are trademarks of their respective companies.

MACRO CODE

```

%macro mindstat (INDATA=,
                 SUBJVAR=,
                 SUBJEXC=,
                 LISTORDN=,
                 OUTDATA=);

*****;
* LISTING
*****;

** Count number of numeric variables;
%let i=1;
%do %while (%scan(&listordn,&i) ne %str());
    %let _var&i = %scan(&listordn,&i);
    %let i = %eval(&i + 1);
%end;
%let pnum = %eval(&i - 1);

data list1;
    set &indata;
    keep &subjvar &listordn;
run;

data LISTOUT;
    length charvar $20;
    ** Create column(s) for numeric vars;
    %do j = 1 %to &pnum;
        length numvar&j $20 dec&j $8;
    %end;

    set list1;
    charvar = compress(&subjvar);
    drop &subjvar;

    ** Display numeric vars with correct number of decimal places;
    %do j = 1 %to &pnum;
        call vname(%scan(&listordn,&j),dec&j);
        if substr(right(dec&j),8,1) = '1' then numvar&j=
            put(%scan(&listordn,&j),8.1);
        else if substr(right(dec&j),8,1) = '2' then numvar&j=
            put(%scan(&listordn,&j),9.2);
        else if substr(right(dec&j),8,1) = '3' then numvar&j=
            put(%scan(&listordn,&j),10.3);
        else if substr(right(dec&j),8,1) = '4' then numvar&j=
            put(%scan(&listordn,&j),11.4);
        else if substr(right(dec&j),8,1) = '5' then numvar&j=
            put(%scan(&listordn,&j),12.5);
        else if substr(right(dec&j),8,1) not in ('1','2','3','4','5') then numvar&j=
            put(%scan(&listordn,&j),6.);
        drop dec&j %scan(&listordn,&j);
    %end;
run;

*****;
* TABLE
*****;

** Count number of subjects excluded from stats;
%let i = 1;
%do %while (%scan(&subjexc,&i) ne %str() );
    %let _var&i = %scan(&subjexc,&i);
    %let i = %eval(&i + 1);

```

```

%end;
%let enum = %eval(&i - 1);

** Exclude specified subjects from stats only;
%if &enum ne 0 %then %do;
    data &indata;
        set &indata;
        %do j = 1 %to &enum;
            if &subjvar="%scan(&subjexc,&j)" then delete;
        %end;
    run;
%end;

%else %do;
    data &indata;
        set &indata;
    run;
%end;

** Calculate stats;
proc transpose data=&indata out=stat1(rename=(COL1=val _NAME_=parname));
    by &subjvar;
    var &listordn;
run;

data temp;
    set stat1;
    if val > 0 then ln=log(val);
run;

proc means data=stat1 nway missing noprint;
    class parname;
    var val;
    output out = stat2
        n = N
        mean = Mean
        median = Median
        min = Minimum
        max = Maximum
        std = SD
        stderr = SE
        cv = cv
        p25 = p25
        p75 = p75;
run;

proc means data=temp nway missing noprint;
    class parname;
    var ln;
    output out = gm1
        mean = gmean;
run;

data gm1;
    set gm1;
    gm = exp(gmean);
    keep parname gm;
run;

data stat3;
    merge stat2
        gm1;
    by parname;

```

```

run;

proc sort data=stat3;
  by parname;
run;

proc transpose data=stat3 out=stat4;
  by parname;
  var N Mean Median Minimum Maximum SD SE CV P25 P75 GM;
run;

data stat5;
  length statname $8 statval $20;
  set stat4;
  statname = _name_;
  _name_ = upcase(_name_);
  select (_name_);
  when ('N') suborder = 1;
  when ('MEAN') suborder = 2;
  when ('GM') suborder = 3;
  when ('SD') suborder = 4;
  when ('SE') suborder = 5;
  when ('CV') suborder = 6;
  when ('MINIMUM') suborder = 7;
  when ('P25') suborder = 8;
  when ('MEDIAN') suborder = 9;
  when ('P75') suborder = 10;
  when ('MAXIMUM') suborder = 11;
  otherwise; end;

  if substr(right(parname),8,1) = '1' then do;
    if suborder = 1 then statval = put(coll,6.);
    else if suborder in (7,11) then statval = put(coll,8.1);
    else if suborder in (2,3,8,9,10) then statval = put(coll,9.2);
    else if suborder in (4,5) then statval = put(coll,10.3);
    else if suborder in (6) then statval = put(coll,9.2);
  end;
  else if substr(right(parname),8,1) = '2' then do;
    if suborder = 1 then statval = put(coll,6.);
    else if suborder in (7,11) then statval = put(coll,9.2);
    else if suborder in (2,3,8,9,10) then statval = put(coll,10.3);
    else if suborder in (4,5) then statval = put(coll,11.4);
    else if suborder in (6) then statval = put(coll,9.2);
  end;
  else if substr(right(parname),8,1) = '3' then do;
    if suborder = 1 then statval = put(coll,6.);
    else if suborder in (7,11) then statval = put(coll,10.3);
    else if suborder in (2,3,8,9,10) then statval = put(coll,11.4);
    else if suborder in (4,5) then statval = put(coll,12.5);
    else if suborder in (6) then statval = put(coll,9.2);
  end;
  else if substr(right(parname),8,1) = '4' then do;
    if suborder = 1 then statval = put(coll,6.);
    else if suborder in (7,11) then statval = put(coll,11.4);
    else if suborder in (2,3,8,9,10) then statval = put(coll,12.5);
    else if suborder in (4,5) then statval = put(coll,13.6);
    else if suborder in (6) then statval = put(coll,9.2);
  end;
  else if substr(right(parname),8,1) = '5' then do;
    if suborder = 1 then statval = put(coll,8.);
    else if suborder in (7,11) then statval = put(coll,12.5);
    else if suborder in (2,3,8,9,10) then statval = put(coll,13.6);
    else if suborder in (4,5) then statval = put(coll,14.7);
  end;

```

```

        else if suborder in (6) then statval = put(coll,9.2);
    end;
    else if substr(right(parname),8,1) not in ( '1','2','3','4','5') then do;
        if suborder = 1 then statval = put(coll,6.);
        else if suborder in (7,11) then statval = put(coll,6.);
        else if suborder in (2,3,8,9,10) then statval = put(coll,8.1);
        else if suborder in (4,5) then statval = put(coll,9.2);
        else if suborder in (6) then statval = put(coll,9.2);
    end;
    if statval = ' ' then statval = '   ';
    keep parname statname statval COL1 suborder;
run;

proc sort data=stat5;
    by suborder statname;
run;

proc transpose data=stat5 out=stat6(drop=_name_);
    by suborder statname;
    id parname;
    var statval;
run;

data STATOUT;
    length charvar $20;
    * Create column(s) for numeric vars;
    %do j = 1 %to &pnum;
        length numvar&j $20;
    %end;
    set stat6;
    * Stat names will always go under subject identifier;
    charvar = statname;
    * Assign parameter stat values to variables in specified order;
    %do j = 1 %to &pnum;
        numvar&j = %scan(&listordn,&j);
    %end;
    drop statname &listordn;
run;

** Add blank line between listing and stats;
data BLANK;
    length charvar $20;
    %do j = 1 %to &pnum;
        length numvar&j $20;
    %end;

    charvar = ' ';
    %do j = 1 %to &pnum;
        numvar&j = ' ';
    %end;
    output;
run;

** Combine listing and stats;
data &outdata;
    set listout (in=a)
        blank   (in=b)
        statout (in=c);
    if a or b then pagenum=1;
    else if c then pagenum=2;
run;

%mend mindstat;

```