PharmaSUG China 2016 - 13

# Creating Reader-friendly Clinical Summary Statistics Tables Using PROC REPORT and ODS RTF

Shiran Chen, Cincinnati Children's Hospital Medical Center, Cincinnati, OH, US

## ABSTRACT

In clinical trial reporting, a standard summary table displays the continuous and/or categorical variables of interest and the statistics in rows, with the treatments and the associated p-values for treatment comparison in columns. Such a summary table with statistics comparison is always generated before conducting any inferential analysis among treatments. Therefore, it is beneficial to design the table so it enables statisticians and researchers to read the document easily and process information quickly, especially when there is a need to summarize a large number of variables.

One can enhance the readability of the summary statistics table by suppressing certain table gridlines to "group" results of the same variable together, formatting row headers as bold text for emphasis, creating leading space for sub row headers, and adding superscripts to indicated footnotes, etc. This paper applies these methods to create a reader-friendly version of the clinical summary table in Microsoft Word using ODS RTF and PROC REPORT techniques, with a focus on the usage of STYLES in the REPORT and COMPUTE statements.

## INTRODUCTION

A properly designed summary statistics table is important in clinical trial analysis, as it allows its users to locate and extract information easily, especially when users are summarizing a large amount of variables, such as demographics, baseline characteristics, laboratory measures, and questionnaire data.

The goal of this paper is to demonstrate a way to produce a more informative and visually friendly version of the summary statistics table, like the one shown in Table 1, using STYLES in PROC REPORT and ODS techniques.

The main features of the table are as follows:

- Suppressed horizontal gridlines inside variable blocks
- Bold row headers
- Blank space ahead of sub row headers
- Superscripts and footnotes

**Table**
**Summary of Demographic**
**(Intent-to-treat)**

| Parameter<br>Statistic | Active<br>(N=149) | Placebo<br>(N=148) | P-value |
|---|---|---|---|
| **Age** | | | 0.2215[a] |
| N | 149 | 148 | |
| Min-Max | 15.1-22.5 | 15.1-22.0 | |
| Mean (SD) | 18.3 (1.61) | 18.1 (1.79) | |
| Median | 18.3 | 17.9 | |
| **Sex** | | | 0.4108[b] |
| Male | 83 (55.7%) | 90 (60.8%) | |
| Female | 66 (44.3%) | 58 (39.2%) | |
| **Weight (kg)** | | | 0.3480[a] |
| N | 148 | 147 | |
| Min-Max | 39.0-122.2 | 37.6-145.6 | |
| Mean (SD) | 66.2 (14.70) | 65.3 (16.52) | |
| Median | 63.8 | 61.2 | |
| **Height (cm)** | | | 0.9267[c] |
| N | 148 | 147 | |
| Min-Max | 148.7-192.0 | 143.1-193.6 | |
| Mean (SD) | 168.8 (8.75) | 168.9 (9.41) | |
| Median | 168.3 | 168.8 | |

N = number of subjects within specified treatment.
[a] Two-sided Wilcoxon rank sum test P-value for the treatment comparison.
[b] Two-sided Fisher's exact test P-value for the treatment comparison.
[c] Two-sided ANOVA P-value for the treatment comparison.

**Table 1. Final RTF Table**

Now, let's start building our table step by step.

## STEP 1: FORMAT RESULTS DATASET AND CREATE ORDER VARIABLES

We first need to organize all of the summary statistics into the results dataset, shown in Table 2, before using PROC REPORT and ODS RTF to create our target table.

Notice that the results are ordered based on two order variables, VarOrder and RowOrder. VarOrder specifies the variables' grouping and order. RowOrder specifies the statistics' order (e.g., N, Mean, Median…) or the levels of discrete parameters (e.g., Yes, No, and N/A). These variables are not only helpful in grouping and sequencing all the fields the way we want it, but are also essential in customizing the table's appearance, which will be demonstrated in the later steps.

Note that there is also a leading blank space for each sub row-header where RowOrder ≠ 0. Be sure to create that blank space when managing the results dataset for the layout to be displayed in our RTF table.

| | Pagebreak | VarOrder | RowOrder | Label | colA | colB | Pvalue |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | Age | | | 0.2215^{super a} |
| 2 | 1 | 1 | 1 | N | 149 | 148 | |
| 3 | 1 | 1 | 2 | Min-Max | 15.1-22.5 | 15.1-22.0 | |
| 4 | 1 | 1 | 3 | Mean (SD) | 18.3 (1.61) | 18.1 (1.79) | |
| 5 | 1 | 1 | 4 | Median | 18.3 | 17.9 | |
| 6 | 1 | 2 | 0 | Sex | | | 0.4108^{super b} |
| 7 | 1 | 2 | 1 | Male | 83 (55.7%) | 90 (60.8%) | |
| 8 | 1 | 2 | 2 | Female | 66 (44.3%) | 58 (39.2%) | |
| 9 | 1 | 3 | 0 | Weight (kg) | | | 0.3480^{super a} |
| 10 | 1 | 3 | 1 | N | 148 | 147 | |
| 11 | 1 | 3 | 2 | Min-Max | 39.0-122.2 | 37.6-145.6 | |
| 12 | 1 | 3 | 3 | Mean (SD) | 66.2 (14.70) | 65.3 (16.52) | |
| 13 | 1 | 3 | 4 | Median | 63.8 | 61.2 | |
| 14 | 1 | 4 | 0 | Height (cm) | | | 0.9267^{super c} |
| 15 | 1 | 4 | 1 | N | 148 | 147 | |
| 16 | 1 | 4 | 2 | Min-Max | 148.7-192.0 | 143.1-193.6 | |
| 17 | 1 | 4 | 3 | Mean (SD) | 168.8 (8.75) | 168.9 (9.41) | |
| 18 | 1 | 4 | 4 | Median | 168.3 | 168.8 | |

**Table 2. Results SAS Table**

## STEP 2: GENERATE RTF FILES

ODS RTF is a powerful tool. It allows tables, listings and graphics generated by SAS procedures to be output to Microsoft Word. All you need is two additional lines of code like this:

```
ods rtf path="&outdir" file="unformatted.rtf";

proc report data = forreport split = "|";
 columns ("Table|Summary of Demographic|(Intent-to-treat)"
          label colA colB pvalue);
 define label /display style(column)={cellwidth=170pt} "Parameter|Statistic";
 define colA /display center style(column)={cellwidth=90pt} "Active|&nA";
 define colB /display center style(column)={cellwidth=90pt} "Placebo|&nB";
 define pvalue /display center style(column)={cellwidth=75pt} "P-value";
run;

ods rtf close;
```

Furthermore, through the OPTIONS statement, we can define page settings such as the paper size, paper margins, orientation, page number, date and time, etc.

```
options nonumber nodate papersize = A4 orientation = portrait;
```

That code produces the table shown in Table 3, whose style and format can be hard to read. Now let's apply it as our base program: We will incorporate additional code to add extra information and to make the visualization more reader friendly.

| Table<br>Summary of Demographic<br>(Intent-to-treat) | | | |
|---|---|---|---|
| **Parameter<br>Statistic** | **Active<br>(N=149)** | **Placebo<br>(N=148)** | **P-value** |
| Age | | | 0.2215 |
| N | 149 | 148 | |
| Min-Max | 15.1-22.5 | 15.1-22.0 | |
| Mean (SD) | 18.3 (1.61) | 18.1 (1.79) | |
| Median | 18.3 | 17.9 | |
| Sex | | | 0.4108 |
| Male | 83 (55.7%) | 90 (60.8%) | |
| Female | 66 (44.3%) | 58 (39.2%) | |
| Weight (kg) | | | 0.3480 |
| N | 148 | 147 | |
| Min-Max | 39.0-122.2 | 37.6-145.6 | |
| Mean (SD) | 66.2 (14.70) | 65.3 (16.52) | |
| Median | 63.8 | 61.2 | |
| Height (cm) | | | 0.9267 |
| N | 148 | 147 | |
| Min-Max | 148.7-192.0 | 143.1-193.6 | |
| Mean (SD) | 168.8 (8.75) | 168.9 (9.41) | |
| Median | 168.3 | 168.8 | |

**Table 3. Unformatted RTF Table**

## STEP 3: CREATE SUPERSCRIPTS AND FOOTNOTES

Sometimes p-values for treatment comparisons need to be reported in the clinical summary table. We can add superscripts to the p-values and the associated statistical method descriptions in the footnotes using the inline format function `^{super Symbol}`to denote the different statistical models used for testing different variables.

Inline formatting functions can be used to modify text strings within table cells, titles and footnotes. They provide the ability to enhance and interpret text strings used by statements and variables. To apply them with ODS, you must first specify the character that will be used as the escape character value with an ODS ESCAPECHAR= statement. Here we use:

```
ods escapechar="^";
```

Then we can create a character variable for p-value, like the one in Table 2, by concatenating the number and text string `^{super Symbol}` and using the same symbol in the footnote. The following code produces footnotes for the table using COMPUTE block in PROC REPORT.

```
compute after _page_ ;
  line @1 "N = number of subjects within specified treatment." ;
  line @1 "^{super a} Two-sided Wilcoxon rank sum test P-value for the treatment
comparison.";
  line @1 "^{super b} Two-sided Fisher's exact test P-value for the treatment
comparison.";
  line @1 "^{super c} Two-sided ANOVA P-value for the treatment comparison.";
endcomp;
```

## STEP 4: USING STYLE IN PROC REPORT STATEMENT

If we compared Table 1 with Table 3, we can note that our final table

1. does not have horizontal gridlines within each variable block,
2. does not have a background color in the table header,
3. has a blank space ahead of each sub row header.

We can achieve these features with STYLE (REPORT), STYLE (HEADER), and STYLE (COLUMN), respectfully, in the PROC REPORT statement by adding the following code to our base program.

Table 4 displays the resulting output. We are not finished yet but are very close to the final.

```
proc report data = forreport split = "|"
 style(report)={rules=cols cellspacing=2 cellpadding=2}
 style(header)={background=white borderbottomwidth=0.75pt}
 style(column)={asis=on};
```

### REMOVE INSIDE HORIZONTAL GRIDLINES

The default settings of the PROC REPORT generate gridlines around all cells, which make the visualization difficult to read for the repeating group variable. Thus, we want to remove these extra gridlines to improve readability.

In our case, VarOrder is the grouping variable, with each number representing a block in the table. The goal is to keep the outside borders for each block but to remove the inside horizontal ones to ensure that all cells are touching each other vertically. The code `style(report)={rules=cols}` removes all of the table's inside horizontal lines.

### ADD HEADER BORDERS AND CHANGE BACKGROUND COLOR

Since code `style(report)={rules=cols}` removes all inside horizontal gridlines of the table, we have to add back borders for the header. To do this, simply specify the border bottom width using the BORDERBOTTOMWIDTH= option in STYLE (HEADER).

The default RTF style shades the header in grey. To change the color, simply specify in the BACKGROUND= style attribute option.

### MAKE LEADING SPACES VISIBLE IN THE OUTPUT

In step 1, when managing the results dataset, we created leading spaces for all statistic and category cells, because we want this column layout reflected in the output table, as in Table 1. However, that is not the case in Table 3, where the leading blanks are trimmed. To avoid spaces being trimmed, specify ASIS=ON in STYLE (COLUMN).

**Table**
**Summary of Demographic**
**(Intent-to-treat)**

| Parameter<br>Statistic | Active<br>(N=149) | Placebo<br>(N=148) | P-value |
|---|---|---|---|
| Age | | | 0.2215[a] |
| N | 149 | 148 | |
| Min-Max | 15.1-22.5 | 15.1-22.0 | |
| Mean (SD) | 18.3 (1.61) | 18.1 (1.79) | |
| Median | 18.3 | 17.9 | |
| Sex | | | 0.4108[b] |
| Male | 83 (55.7%) | 90 (60.8%) | |
| Female | 66 (44.3%) | 58 (39.2%) | |
| Weight (kg) | | | 0.3480[a] |
| N | 148 | 147 | |
| Min-Max | 39.0-122.2 | 37.6-145.6 | |
| Mean (SD) | 66.2 (14.70) | 65.3 (16.52) | |
| Median | 63.8 | 61.2 | |
| Height (cm) | | | 0.9267[c] |
| N | 148 | 147 | |
| Min-Max | 148.7-192.0 | 143.1-193.6 | |
| Mean (SD) | 168.8 (8.75) | 168.9 (9.41) | |
| Median | 168.3 | 168.8 | |

N = number of subjects within specified treatment.
[a] Two-sided Wilcoxon rank sum test P-value for the treatment comparison.
[b] Two-sided Fisher's exact test P-value for the treatment comparison.
[c] Two-sided ANOVA P-value for the treatment comparison.

**Table 4. Semi-formatted RTF Table**

## STEP 5: USING STYLE IN COMPUTE STATEMENT

Before generating the final table, we must complete the following in Table 4:

1. add borders for the boundaries of variable block,
2. bold the cells of variable name,
3. add a border for the footnote.

We accomplish these tasks by using STYLE in the COMPUTE statement.

The COMPUTE statement allows us to make programmatic changes for individual table cells, which adds a great deal of power to the PROC REPORT. It is usually applied along with the CALL DEFINE statement to control the format of certain rows or cells.

The code shown below achieves the first two tasks. The first CALL DEFINE statement draws the top borders for each row, where RowOrder = 0; this is the first row within each variable block or the row of the variable name. The second CALL DEFINE statement specifies that, instead of entire rows, we format only the cells of the variable name in bold.

Now you can see the use of the order variables in modifying values for specific cells. Before using them, we need to list them in the COLUMN statement and define their variable type as ORDER in the DEFINE statement and use the NOPRINT option to suppress the display in the output table.

```
define varorder /order order=formatted noprint;
define roworder /order order=formatted noprint;

compute label;
  if roworder = 0 then call define(_row_,"style","style={bordertopwidth=0.75pt}");
  if roworder = 0 then call define(_col_,"style","style={font_weight=bold}");
endcomp;
```

Notice here that, although we are not using the group/order variable VarOrder in the COMPUTE block, we still want to include and define it in PROC REPORT. This is because you don't need to sort the results dataset by the order variables in advance, which is required if you do not define them in the PROC REPORT, and people usually forget to sort them! The default sorting method for PROC REPORT is ORDER=FORMATTED, which sorts by a variable's formatted values in ascending order.

Lastly, to add a top border for the footnote, simply specify the width of border in STYPE with the COMPUTE AFTER _PAGE_ statement like the following.

```
compute after _page_ /style={bordertopwidth=0.75pt font_size=9.5pt};
  …
endcomp;
```

Now we have covered all the necessary steps! Here is the complete code for creating our task table in Table 1.

```
options nonumber nodate papersize = A4 orientation = portrait;
ods escapechar="^";
ods rtf path="&outdir" file="demog_sum.rtf";

proc report data = forreport split = "|"
 style(report)={rules=cols cellspacing=2 cellpadding=2}
 style(header)={background=white borderbottomwidth=0.75pt}
 style(column)={asis=on};

 columns ("Table|Summary of Demographic|(Intent-to-treat)"
         varorder roworder label colA colB pvalue);

 define varorder /order order=formatted noprint;
 define roworder /order order=formatted noprint;
 define label /display style(column)={cellwidth=170pt} "Parameter|Statistic";
 define colA /display center style(column)={cellwidth=90pt} "Active|&nA";
 define colB /display center style(column)={cellwidth=90pt} "Placebo|&nB";
 define pvalue /display center style(column)={cellwidth=75pt} "P-value";

 compute label;
  if roworder = 0 then call define(_row_,"style","style={bordertopwidth=0.75pt}");
  if roworder = 0 then call define(_col_,"style","style={font_weight=bold}");
 endcomp;

 compute after _page_ /style={bordertopwidth=0.75pt font_size=9.5pt};
  line @1 "N = number of subjects within specified treatment." ;
  line @1 "^{super a} Two-sided Wilcoxon rank sum test P-value for the treatment
comparison.";
```

```
  line @1 "^{super b} Two-sided Fisher's exact test P-value for the treatment
comparison.";
  line @1 "^{super c} Two-sided ANOVA P-value for the treatment comparison.";
 endcomp;
run;

ods rtf close;
```

## CONCLUSIONS

By applying the methods discussed here, we can generate more informative, organized, and visually-friendly clinical summary statistics tables. The display styles can easily be modified as needed. When reporting a long table, just create a page break variable, define the variable type as group or order in the PROC REPORT, and apply the code `break after pagebreak/page` at the end.

## REFERENCES

SAS Guide to the REPORT Procedure, Usage and Reference, Version 6. Cary, NC: SAS Institute Inc, 1990.

SAS Report Writing: A Programming Approach: Course Notes, Cary, NC: SAS Institute Inc,2001.

## CONTACT INFORMATION

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

Shiran Chen
Cincinnati Children's Hospital Medical Center
3333 Burnet Ave. MLC 5041
Cincinnati, OH 45229
(513) 636-7124
shiran.chen@cchmc.org