SESUG Paper 057-2021

Using the SAS® HPBIN Procedure to Create Format Value Ranges for Numeric Variables

Michael A. Raithel, Westat, Inc.

Abstract

How would *you* go about determining the format value ranges for a new continuous numeric variable? You could run PROC MEANS to get the min, max, median, and quantiles; and then construct it from those metrics. That works, but it constrains you to having only four value ranges for your format. Also, the MEANS Procedure output is not in a structure conducive to creating the actual SAS Format Start/End/Label statements. It would be advantageous to have a methodology whereby a programmer could choose the desired number of value ranges and generate output close to what is needed for the PROC FORMAT VALUE statements.

One such method is to employ a binning technique available through SAS's High Performance Bin procedure. PROC HPBIN can be used to provide mathematically sound, defensible methodologies for creating the value ranges of numeric variables. Programmers can specify the number of bins (rows) they desire and PROC HPBIN computes the numerical boundaries that can used to define the Start/End values in PROC FORMAT statements.

This paper introduces the **Continuous Variable Format Start and End Values Creator** program; which contains a macro that utilizes the HPBIN Procedure to create suggested SAS format Start/End values. Users can specify either of two main binning techniques and the macro produces a spreadsheet of computed value ranges. The Start, End, and Label columns in the Excel spreadsheet can be transcribed into a SAS Format Value statement. SAS programmers can copy the macro program from Appendix A and begin using it right away to define their own numeric formats.

Introduction

The **Continuous Variable Format Start and End Values Creator** macro program utilizes the HPBIN Procedure. PROC HPBIN uses mainstream mathematical *binning* practices to compute the number of bins (rows), based on the characteristics of the actual data. That eliminates guesswork and provides a sound framework for scientifically computing the value ranges. There are four possible binning techniques that you can specify for PROC HPBIN, but this macro allows a user to employ either one of these two: Quantile or Bucket binning.

The **Continuous Variable Format Start and End Values Creator** macro program requires six parameters:

- **DIRNAME** – The directory of the target SAS data set
- **DSNAME** – The name of the SAS data set holding the numeric variable
- **VARNAME** – The name of the numeric variable whose range values will be computed
- **BINTYPE** – The type of binning to be performed. There are two possible values:
  - **BUCKET** – Compute rows based on the values of the variable
  - **QUANTILE** – Compute rows based on the number of observations in the SAS data set so every row has the same number of observations
- **NUMROWS** – The desired number of rows for the format
- **REPTDIR** – The directory where the Excel report file will be written

This is an example of an invocation of the macro:

```
%FORMATSE(DIRNAME=H:\MY DOCUMENTS\My SAS Programs\Binning Macro,
          DSNAME=orsales,
          VARNAME=profit,
          BINTYPE=quantile,
          NUMROWS=5,
          REPTDIR=H:\MY DOCUMENTS\My SAS Programs\Binning Macro);
```

The macro produces an Excel spreadsheet with the following columns:

- Variable – the name of the variable
- Bin – The bin number; 1, 2, 3, etc.

- Start Value – The lowest value for this bin
- End Value – the highest value for this bin
- Label – A label that can be used in the format
- Frequency – The number of observations in this bin
- Proportion – The percentage of the data set's observations that reside in this bin

Examples of the Excel output files for both Quantile and Bucket binning can be found in the sections of this paper that detail each method.

Once the Excel spreadsheet has been generated, you can use the Start, End, and Label columns to craft a SAS format VALUE statement. The only caveat is that you will need to adjust some of the Start values so that the format categories do not overlap. This is explained further in the sections on Quantile Binning and Bucket Binning.

The next two sections of this paper provide more information on Quantile and Bucket binning and how they are utilized by the **Continuous Variable Format Start and End Values Creator** macro program. Next, this paper reviews the code in the three sections of the SAS program. That is followed by a section on how you can implement the macro in your own computing environment. Finally, Appendixes A and B contain the **Continuous Variable Format Start and End Values Creator** macro program and a driver program for the macro.

Quantile Binning

Quantile binning aims to assign the same number of observations to each bin. As a result, each bin should have the same number of observations, provided that there are no tied values at the boundaries of the bins.

Quantile binning is based on the number of observations in the SAS data set. The HPBIN procedure essentially divides the Number of Observations by the number of requested bins and adjusts the lower/upper bounds and the ranges so that there is an equal number of observations in each bin. For example, if you had 912 oberations to be divided into 6 bins, it would be: 912/6 = **152** observations per bin.

Here is an example of the Excel file output of having the **Continuous Variable Format Start and End Values Creator** macro program run Quantile Binning for five bins against the PROFIT variable in the sashelp.orsales data set:

Variable	Bin	Start Value	End Value	Label	Frequency	Proportion
Profit	1	209.80	7,508.55	209.8 - 7508.55	183	20.07%
Profit	2	7,508.55	18,050.195	7508.55 - 18050.195	182	19.96%
Profit	3	18,050.195	43,213.32	18050.195 - 43213.32	183	20.07%
Profit	4	43,213.32	107,765.055	43213.32 - 107765.055	182	19.96%
Profit	5	107,765.055	552,970.51	107765.055 - 552970.51	182	19.96%
					912	

Note that SAS has adjusted the Start Values and the End Values so that each of the bins has about the same number of observations in it.

A programmer reading that output might create a SAS format from it that looks like this:

```
proc format;
value quantbin
      low - 7508.55          = "<= 7508.55"
      7508.56 - 18050.195    = "7508.56 - 18050.195"
      18050.196 - 43213.32   = "18050.196 - 43213.32"
      43213.33 - 107765.055  = "43213.33 - 107765.055"
      107765.056 - high      = ">= 107765.056";
      ;
run;
```

If you compare the Start values in the output Excel file to those in the PROC FORMAT, you will see that the programmer manually adjusted them slightly upwards in the Format Procedure. This is done to make sure that the format's value ranges are unique and do not overlap. For example, Bin 2 has a Start of 7,508.55; while the second "row" of the format has a Start of 7506.56.

Using the QUANTBIN format in a PROC FREQ against the PROFIT variable in the sashelp.orsales SAS data set produces the following:

Profit in USD				
Profit	**Frequency**	**Percent**	**Cumulative Frequency**	**Cumulative Percent**
<= 7508.55	184	20.18	184	20.18
7508.56 - 18050.195	181	19.85	365	40.02
18050.196 - 43213.32	183	20.07	548	60.09
43213.33 - 107765.055	182	19.96	730	80.04
>= 107765.056	182	19.96	912	100.00

The distribution of observations among the value ranges is slightly different than in the original macro output; but is overall reasonable.

Bucket Binning

Bucket binning creates equal-length bins and assigns the data to one of these bins. It is based on the actual values of the *variable* that is being binned. The calculation is essentially:

1. Subtract the minimum value from the maximum value
2. Divide by the requested number of bins
3. Add the minimum value to the result of the division and set Lower/Upper bounds and Range accordingly

Consequently, each range will be set equal to the number from the result of the final calculation (#3), above.

For example, take a variable with a maximum value of 9026 and a minimum value of 10. The calculation would be:

- (9026 - 10)/6 = **1502.67**.
  - Sets first boundary to 1502.67 + 10 = 1512.67 and goes from there.
  - Sets second boundary 1512.67 + **1502.67** = 3015.33

Here is an example of the output of having the **Continuous Variable Format Start and End Values Creator** macro program run Bucket Binning for five bins against the PROFIT variable in the sashelp.orsales data set:

Variable	Bin	Start Value	End Value	Label	Frequency	Proportion
Profit	1	209.80	110,761.942	209.8 - 110761.942	735	80.59%
Profit	2	110,761.942	221,314.084	110761.942 - 221314.084	126	13.82%
Profit	3	221,314.084	331,866.226	221314.084 - 331866.226	36	3.95%
Profit	4	331,866.226	442,418.368	331866.226 - 442418.368	9	0.99%
Profit	5	442,418.368	552,970.51	442418.368 - 552970.51	6	0.66%
					912	

A programmer reading that output might create a SAS format from it that looks like this:

```
proc format;
```

```
value bucketbin
      low - 110761.942          = "<= 110761.942"
      110761.943 - 221314.084   = "110761.943 - 221314.084"
      221314.085 - 331866.226   = "221314.085 - 331866.226"
      331866.227 - 442418.368   = "331866.227 - 442418.368"
      442418.369 - high         = ">= 442418.369"
      ;
run;
```

If you compare the Start values in the output Excel file to those in the PROC FORMAT, you will see that the programmer manually adjusted them slightly upwards in the Format Procedure. This is done to make sure that the format's value ranges are unique and do not overlap. For example, Bin 2 has a Start of 7,508.55; while the second "row" of the format has a Start of 7506.56.

Using that format in a PROC FREQ against the PROFIT variable in the sashelp.orsales SAS data set produces the following:

Profit in USD				
Profit	**Frequency**	**Percent**	**Cumulative Frequency**	**Cumulative Percent**
<= 110761.942	735	80.59	735	80.59
110761.943 - 221314.084	126	13.82	861	94.41
221314.085 - 331866.226	36	3.95	897	98.36
331866.227 - 442418.368	9	0.99	906	99.34
>= 442418.369	6	0.66	912	100.00

The Continuous Variable Format Start and End Values Creator macro

This part of the paper discusses the three sections of the **Continuous Variable Format Start and End Values Creator** macro program. The macro code can be found in Appendix A.

Section I

Section I calculates the target variable's minimum and maximum values and plugs them into macro variables that will be used to modify the data set created by PROC HPBIN in Section II. It does so by running PROC MEANS against the variable specified by input parameter &VARNAME and storing the output in a SAS data set named MEANSOUT.

Next, a DATA _NULL_ step reads through the MEANSOUT data set, stores the minimum value in macro variable &MINVAL, and stores the maximum value in macro variable &MAXVAL. Both of these macro variables are now ready to be used in Section II.

Section II

This section begins by executing PROC HPBIN and storing the output in a SAS data set named FormatStartEnd. Four of the input macro parameters are used in the invocatin of the HPBIN Procedure:

- &DSNAME – The name of the SAS data set holding the variable
- &NUMROWS – The number of bins, which is also the number of rows for the format
- &BINTYPE – The type of binning to be performed; either BUCKET or QUANTILE.
- &VARNAME – The name of the variable to be binned

In addition to creating the aforementioned SAS data set, PROC HPBIN creates several reports that make interesting reading.

Next, a DATA step processes the FormatsStartEnd data set in order to update the values stored in the RANGE variable that was created by HPBIN. This is necessary because we are going to use the RANGE variable as our future format's LABEL and need to make some structural adjustments.

This is what the range variable looks like coming directly out of PROC HPBIN:

Range
Profit < 7508.55
7508.55 <= Profit < 18050.195
18050.195 <= Profit < 43213.32
43213.32 <= Profit < 107765.055
107765.055 <= Profit

…and we need to transform it to look like this:

Label
209.8 - 7508.55
7508.55 - 18050.195
18050.195 - 43213.32
43213.32 - 107765.055
107765.055 - 552970.51

…in order to have proper Format LABEL values.

Consequently, the DATA step replaces:

- "Profit <" with the value in the &MINVAL macro variable—preceded by a hyphen
- "<= Profit <" with a hyphen
- "<= Profit" with the value of the &MAXVAL macro variable—preceeded by a hyphen

All of these transformations are performed with judicious usage of the TRANWRD function.

Section III

This section builds the Excel report file using the FormatStartEnd SAS data set that was updated in the previous section. It creates the Excel file in the directory specified by the &REPTDIR macro variable using an ODS statement. The number of rows, variable name, and binnng type are all embeded in the name of the Excel file via use of the &NUMROWS, &VARNAME, and &BINTYPE macro variables, respectively.

The PRINT Procedure is employed to create the report. It specifies the order of the variables, the format of the Start, End, and Proportion columns; and applies labels. The report is "printed" to the Excel file specified by the previously described ODS statement.

Operationalizing the Continuous Variable Format Start and End Values Creator macro program

Here are some simple steps you can consider implementing in order to operationalize the **Continuous Variable Format Start and End Values Creator** macro program in your own environment.

1. Copy the macro from Appendix A into a SAS program on your computer. You can save it in your autocall macro library or save it in a directory of your choosing.

2. If you do not choose to put the program in one of your organization's macro libraries, take a look at Appendix B, which contains the **Execute Continuous Variable Format Start and End Values Creator** SAS program. This is a driver program for the **Continuous Variable Format Start and End Values Creator** macro program. Simply copy the SAS code from Appendix B to a SAS program in your environment. Then, update the %INCLUDE statement in the driver program to specify the full path to where you placed the utility macro.

3. Once you have "downloaded" the macro program and the driver program, simply specify the six parameters in the macro call:

    - **DIRNAME** – The directory of the target SAS data set
    - **DSNAME** – The name of the SAS data set holding the numeric variable
    - **VARNAME** – The name of the numeric variable whose range values will be computed
    - **BINTYPE** – The type of binning to be performed; **BUCKET** or **QUANTILE**
    - **NUMROWS** – The desired number of rows for the format
    - **REPTDIR** – The directory where the Excel report file will be written

    …and execute the macro to create your format Start/End/Label Excel report file.

Conclusion

This paper introduced the **Continuous Variable Format Start and End Values Creator** macro program; which can be used to compute numerical boundaries that can used to define the Start/End values in PROC FORMAT statements. The program uses the SAS High Performance Bin Procedure—PROC HPBIN—to calculate value ranges using either Quantile or Bucket binning. The macro creates an Excel output file that programmers can use to craft the value ranges and labels on the FORMAT Procedure's VALUE statements.

The **Continuous Variable Format Start and End Values Creator** macro program is ideal for creating mathematically sound, defensible value ranges for continuous numeric variables. Consider how this program can be of help when you need to create new formats for numerical variables.

References

Raithel, Michael A. 2017. Did You Know That? Essential Hacks for Clever SAS Programmers: Over 100 Essential Hacks to Make Your Programs Leaner, Cleaner, and More Competitive. Bethesda, Maryland: Michael A. Raithel
Available: http://www.amazon.com/Michael-A.-Raithel/e/B001K8GG90/ref=ntt_dp_epwbk_0

Raithel, M.A. (2020). A Program to Compare Two SAS Format Catalogs. *Proceedings of the SAS Global Forum 2020 Conference*.
Available: A Program to Compare Two SAS Format Catalogs

Raithel, M.A. (2017). PROC DATASETS; The Swiss Army Knife of SAS Procedures. *Proceedings of the SAS Global Forum 2017 Conference*.
Available: http://support.sas.com/resources/papers/proceedings17/0963-2017.pdf

SAS Institute Inc. 2015. *Base SAS® 9.4 Procedures Guide, Seventh Edition*. Cary, NC: SAS Institute Inc.
Available: SAS Help Center: Base SAS 9.4 Procedures Guide, Seventh Edition

SAS Institute Inc. 2015. *SAS® 9.4 Language Reference: Concepts, Sixth Edition*. Cary, NC: SAS Institute Inc.
Available: SAS 9.4 Language Reference: Concepts, Sixth Edition

Acknowledgments

The author would like to thank Westat management for supporting his participation in SAS conferences and user groups.

Recommended Reading

- *Base SAS® Procedures Guide*
- *SAS® For Dummies®*

Contact Information

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

michaelraithel@westat.com

Appendix A – The Continuous Variable Format Start and End Values Creator SAS Macro Program

```
/*****************************************************************************************************/
/*  Program: Continuous Variable Format Start and End Values Creator.sas                             */
/*                                                                                                   */
/*  Author: Michael A. Raithel                                                                       */
/*                                                                                                   */
/*  Created: 5/24/2021                                                                               */
/*                                                                                                   */
/*  Purpose: This SAS program uses PROC HPBIN to create proposed START and END format values for a   */
/*           specified variable. Users provide the number of rows they want for the format, and this */
/*           macro calculates them using either HPBIN's BUCKET or QUANTILE methodology. More details */
/*           on these methodologies can be found in the PROC HPBIN documenation:                     */
/*                                                                                                   */
/*           https://documentation.sas.com/doc/en/pgmsascdc/v_006/prochp/prochp_hpbin_toc.htm        */
/*                                                                                                   */
/*  Parameters: Users must specify the following six parameters:                                     */
/*                                                                                                   */
/*             DIRNAME    - The directory of the target SAS data set                                */
/*             DSNAME     - The name of the SAS data set holding the variable                       */
/*             VARNAME    - The name of the variable whose values will be characterized             */
/*             BINTYPE    - The type of binning to be performed. There are two possible values:     */
/*                                                                                                   */
/*                          BUCKET - Compute rows based on the values of the variable                */
/*                          QUANTILE - Compute rows based on the number of observations in the SAS   */
/*                                     data set so every row has the same number of observations     */
/*                                                                                                   */
/*             NUMROWS    - The desired number of rows for the format                               */
/*             REPTDIR    - The directory where the report will be written                          */
/*                                                                                                   */
/*  Outputs: This program creates a Excel spreadsheet with the specified number of rows and the     */
/*           calculated upper and lower boundary values. Those values can be used to specify the     */
/*           START and END values for a format.                                                      */
/*                                                                                                   */
/*  Change Log:                                                                                      */
/*                                                                                                   */
/*****************************************************************************************************/
options symbolgen mprint mlogic source2;

%MACRO FORMATSE(DIRNAME=, DSNAME=, VARNAME=, BINTYPE=, NUMROWS=, REPTDIR=);

libname INPUTDIR "&DIRNAME" access=readonly;

/*************************************************************************/
/*                              Section I                                */
/*************************************************************************/
/* Calculate the variable's minimum and maximum values and plug them into macro */
/* variables so they can be used in the data set created by PROC HPBIN.          */
/*************************************************************************/

/* Run PROC MEANS to get summary statistics */
proc means data=INPUTDIR.&DSNAME noprint;
       output out=meansout;
```

```
        var &VARNAME;
run;

data _null_;
set meansout;

if _STAT_ = "MIN" then call symput("MINVAL",profit);
if _STAT_ = "MAX" then call symput("MAXVAL",profit);

run;

/*****************************************************************************/
/*                              Section II                                   */
/*****************************************************************************/
/* Execute PROC HPBIN to calculate the upper/lower bounds for the specified  */
/* number of bins. Insert the minimum/maximum values in the resultant data set. */
/*****************************************************************************/

/*Execute HPBIN to determine the Lower and Upper Bounds values */
proc hpbin data=INPUTDIR.&DSNAME numbin=&NUMROWS &BINTYPE computestats;

input &VARNAME;

ods output mapping=FormatStartEnd;

run;

/* Insert the minimum/maximum values */
data FormatStartEnd;
set  FormatStartEnd;

range = upcase(range);

if missing(LB) then do;
        LB = &MINVAL;
        range = tranwrd(range,upcase("&VARNAME"),strip("&MINVAL"));
        range = tranwrd(range,"<","-");

end;
        else if missing(UB) then do;
                        UB = &MAXVAL;
                        range = tranwrd(range,upcase("&VARNAME"),strip("&MAXVAL"));
                        range = tranwrd(range,"<=","-");

                end;
                        else do;
                                range = tranwrd(range,upcase("&VARNAME"),"-");
                                range = tranwrd(range,"<=","");
                                range = tranwrd(range,"<","");
                        end;

run;

/*****************************************************************************/
/*                              Section III                                  */
/*****************************************************************************/
/* Create the Excel report file.                                             */
/*****************************************************************************/

/* Create the report file */
ODS EXCEL
file="&REPTDIR\Start_End_Values_for_a_&NUMROWS._Row_Format_for_Variable_&VARNAME._Using_&BINTYPE._Binning.xlsx";


ODS EXCEL options(sheet_name="&BINTYPE.Binning");

/* Run PROC PRINT for the Start/End values */
proc print noobs data=FormatStartEnd label;
        var variable bin lb ub range frequency proportion;
        sum frequency;
        label LB    = "Start Value"
              UB    = "End Value"
                  range = "Label"
              ;
        format LB UB      comma32.3
               proportion percent5.2;
```

```
run;

ODS Excel close;


%MEND FORMATSE;
```

Appendix B – Execute The Continuous Variable Format Start and End Values Creator Macro Program

```
/*****************************************************************************************************/
/*  Program: Execute Continuous Variable Format Start and End Values Creator.sas                     */
/*                                                                                                   */
/*  Author: Michael A. Raithel                                                                       */
/*                                                                                                   */
/*  Created: 5/21/2021                                                                               */
/*                                                                                                   */
/*  Purpose: This is a driver program for Continuous Variable Format Start and End Values Creator.sas,*/
/*           a macro program that uses PROC HPBIN to create proposed START and END format values for a*/
/*           specified variable. Users provide the number of rows they want for the format, and this  */
/*           macro calculates them using either HPBIN's BUCKET or QUANTILE methodology. More details  */
/*           on these methodologies can be found in the PROC HPBIN documenation:                     */
/*                                                                                                   */
/*           https://documentation.sas.com/doc/en/pgmsascdc/v_006/prochp/prochp_hpbin_toc.htm        */
/*                                                                                                   */
/*  Parameters: Users must specify the following six parameters:                                     */
/*                                                                                                   */
/*              DIRNAME    - The directory of the target SAS data set                               */
/*              DSNAME     - The name of the SAS data set holding the variable                      */
/*              VARNAME    - The name of the variable whose values will be characterized            */
/*              BINTYPE    - The type of binning to be performed. There are two possible values:    */
/*                                                                                                   */
/*                           BUCKET - Compute rows based on the values of the variable               */
/*                           QUANTILE - Compute rows based on the number of observations in the SAS  */
/*                                      data set so every row has the same number of observations    */
/*                                                                                                   */
/*              NUMROWS    - The desired number of rows for the format                              */
/*              REPTDIR    - The directory where the report will be written                         */
/*                                                                                                   */
/*  Outputs: This program creates a Excel spreadsheet with the specified number of rows and the     */
/*           calculated upper and lower boundary values. Those values can be used to specify the     */
/*           START and END values for a format.                                                      */
/*                                                                                                   */
/*  Change Log:                                                                                      */
/*                                                                                                   */
/*****************************************************************************************************/

%INCLUDE "<<Put your diectory path here>>\Continuous Variable Format Start and End Values Creator.sas";

%FORMATSE(DIRNAME=,
          DSNAME=,
          VARNAME=,
          BINTYPE=,
          NUMROWS=,
          REPTDIR=);
```