Paper 3296-2015

Out of Control! A SAS® Macro to Recalculate QC Statistics

Jesse Pratt, Colleen Mangeot, Kelly Olano, Cincinnati Children's Hospital Medical Center, Cincinnati, OH, USA

ABSTRACT

SAS/QC® provides procedures, such as PROC SHEWHART, to produce control charts with centerlines and control limits. When quality improvement initiatives create an out-of-control process of improvement, centerlines and control limits need to be recalculated. While this is not a complicated process, producing many charts with multiple centerline shifts can quickly become difficult. This paper illustrates the use of a macro to efficiently compute centerlines and control limits when one or more recalculations are needed for multiple charts.

INTRODUCTION

Historically, applications of Statistical Process Control have focused on maintaining the centerline and keeping processes in control. SAS QC provides procedures to produce control charts, such as PROC SHEWHART. Control charts, however, may also be used to measure performance improvement. The Institute for HealthCare Improvement has been suggesting the use of control charts to measure improvement in healthcare processes for over 10 years. The Data Management Center at Cincinnati Children's Hospital produces a total of approximately 1000 control charts each month for five separate collaboratives involving from 20 to 40 sites. Other registries and other industry or government groups that work to improve processes may also be monitoring improvement with control charts.

For improvement initiatives the emphasis is on creating an out of control process in which the centerline shifts in the desired direction. Hence, our focus is to monitor centerline shifts rather than notice and correct an out of control situation. While this is not difficult, it quickly becomes cumbersome if the control charts need to be produced for multiple measures and multiple sites, each with their own unique centerline movements. Using a relatively simple macro, this paper will demonstrate how to easily compute and shift centerlines and control limits for multiple charts, and make maintenance of these centerlines simple, efficient, and flexible. This paper will focus on p-charts, however the macro can easily be edited to take other types of control charts into account.

WITHOUT THE MACRO

Prior to the macro development, shifts in control limits and centerlines were programmed using simple, yet repetitive code. To create a control chart, a data set is created with numerators and denominators for each measure. If the centerline needs to be shifted, dates are either hard coded in the program or if many shifts are needed, a second data set containing the dates (referred to in this paper as break dates) is created. Assuming a separate data set is created for the break dates, the first step is to merge this file with the numerator and denominator file. The next step is to create a group variable which categorizes all of the points greater than the first break date and up to the next.

The number of points used to calculate the centerline must be identified. To do this, a counting variable is created and the correct number of points is output to the dataset. Summing the numerators and denominators can then be achieved by running PROC MEANS for each group and outputting the sum. Using the sums from PROC MEANS, the centerline, standard deviation and control limits can be calculated.

There are several limitations to using this process as opposed to the macro. If break dates are hard coded in a program, updates require searching, changing and testing the applicable code. If break dates are stored in a data set, the maximum number of break dates for all graphs needs to be determined in order to use the grouping variable. Also, the number of points used in calculating each centerline must be hardcoded into the program. While individually these are simple processes, the difficulty comes when there are multiple sites with several different centerline shifts and each is unique to the site. Additionally, not having to make changes to a production program is more efficient and effective. With this macro, changes and updates are made to a spreadsheet rather than the program.

THE MACRO – A BETTER SOLUTION

To simplify maintenance of centerline shift data, a spreadsheet with columns for the data set name (DSN), the dependent variable on the p chart (YVar), the independent variable on the p chart (XVar), the break date(s) (Break_date), and the number of points (Pts) to be used to calculate the centerline was created. Each row represents one chart to be produced. The first break date is the starting point of the x axis on the chart. Note that for this macro, the values of the variable XVar must be a SAS® keyword for a time interval, such as month, quarter, week, etc. The data set DSN must contain values of the variable YVar must also include the denominators. In the code presented in the appendix, this is represented by the variable YN.

DSN	YVar	XVar	Break_date	Pts
a9995	Pct_infection	month	1/1/2007 2/1/2009 6/1/2010	20
a9996	Pct_remission	Quarter	2/1/2007	10
a9997	Pct_growth	week	1/1/2007 7/1/2011	30

Table 1, Example of Spreadsheet Layout

The macro requires one macro variable, the file name of the spreadsheet, which is imported to create a SAS data set.

The first step in the macro uses the SQL procedure to determine the number of rows in the spreadsheet, which corresponds to the number of charts to be produced:

```
PROC SQL noprint;
   select count(*)
      into: Nobs
      from breakptsexcel;
quit;
```

The macro variable Nobs created above identifies the end point of a loop for each chart that needs to be produced (each row in the dataset).

Using CALL SYMPUTX, a macro variable is created for each of the column variables in the spreadsheet.

The number of break dates is calculated using the COUNTW function, and is then stored as the macro variable N:

```
data _null_;
   num=countw("&break_date",' ');
   call symputx('N',num);
run;
```

A second loop is started for each break date with a %DO loop incrementing with the macro variable &i.

A SAS data set, period_&i. is created for each break date, containing the observations needed for the period from one break date to the next break date, or if no next break date, the end of the dataset. A macro variable, &nobs2, is created with the number of observations in this data set for later use in the limits data set.

The centerline is calculated using the number of points listed in the spreadsheet and is stored in a SAS data set, centerline_&i.

The limits data set is created assigning the following variables that need to be used with the SHEWHART procedure:

var is the y axis variable

subgrp is the x axis variable

limitn is set equal to .V so the limits vary with the value of n

type is set to ESTIMATE to represent estimated values (Other options are 'STANDARD', 'STDMU', and 'STDSIGMA')

These values and the centerline are output to a data set &nobs2 number of times, so the limits data set has the same number of observations as period_&i. data set. PROC SHEWHART is run with the calculated limits for this period:

```
proc shewhart data = period_&i limits=lim_&i;
   pchart &VAR * &XVAR / tests = 1,2,3
        testnmethod = standardize
        test2run = 8
        subgroupn = &VAR.N
        dataunit = proportion
        outtable = CC_data_&i
        maxpanels=34
        nochart;

run;
```

This concludes the loop for this break date.

Finally, data sets are concatenated for all periods completing the loop for this chart. The table below displays the SAS data set created from the first row of the spreadsheet and contains the variables necessary to graph the control chart. The graph created from this SAS data set is shown in Figure 1.

	VAR	month	_SIGMAS_	_LIMITN_	Pct_infection_SUBN_	Pct_infection_LCLP_	Pct_infection_SUBP_	Pct_infection_P_	Pct_infection_UCLP_	_EXLIM_	_TESTS_
1	pct_infection	01OCT2006	3	60	60	0.490969696	0.54	0.6727007299	0.8544317638		
2	pct_infection	01NOV2006	3	68	68	0.5019941481	0.57	0.6727007299	0.8434073117		
3	pct_infection	01DEC2006	3	76	76	0.5112284689	0.56	0.6727007299	0.8341729909		
4	pct_infection	01JAN2007	3	77	77	0.5122804167	0.6	0.6727007299	0.8331210432		
5	pct_infection	01FEB2007	3	80	80	0.5153170379	0.61	0.6727007299	0.8300844219		
6	pct_infection	01MAR2007	3	90	90	0.5243179621	0.67	0.6727007299	0.8210834978		
7	pct_infection	01APR2007	3	92	92	0.5259396804	0.69	0.6727007299	0.8194617795		
8	pct_infection	01MAY2007	3	97	97	0.5297722235	0.65	0.6727007299	0.8156292363		
9	pct_infection	01JUN2007	3	103	103	0.5339976483	0.67	0.6727007299	0.8114038116		
10	pct_infection	01JUL2007	3	110	110	0.5384834664	0.72	0.6727007299	0.8069179934		
11	pct_infection	01AUG2007	3	120	120	0.5441974835	0.69	0.6727007299	0.8012039763		
12	pct_infection	01SEP2007	3	126	126	0.5472944024	0.77	0.6727007299	0.7981070575		
13	pct_infection	01OCT2007	3	134	134	0.5510954804	0.79	0.6727007299	0.7943059795		
14	pct_infection	01NOV2007	3	139	139	0.7700824479	0.8	0.8587138927	0.9473453375		
15	pct_infection	01DEC2007	3	146	146	0.7722332714	0.85	0.8587138927	0.945194514		
16	pct_infection	01JAN2008	3	152	152	0.7739573106	0.84	0.8587138927	0.9434704748		
17	pct_infection	01FEB2008	3	159	159	0.7758440209	0.83	0.8587138927	0.9415837645		
18	pct_infection	01MAR2008	3	163	163	0.776867145	0.86	0.8587138927	0.9405606404		
19	pct_infection	01APR2008	3	168	168	0.7780943021	0.84	0.8587138927	0.9393334833		
20	pct_infection	01MAY2008	3	171	171	0.7788046207	0.92	0.8587138927	0.9386231647		
21	pct_infection	01JUN2008	3	176	176	0.7799478738	0.9	0.8587138927	0.9374799116		
22	pct_infection	01JUL2008	3	180	180	0.7808279687	0.87	0.8587138927	0.9365998167		
23	pct_infection	01AUG2008	3	185	185	0.8407979762	0.75	0.9053607143	0.9699234523	LOWER	1
24	pct_infection	01SEP2008	3	192	192	0.8419858284	0.84	0.9053607143	0.9687356001	LOWER	1
25	pct_infection	01OCT2008	3	198	198	0.8429534408	0.91	0.9053607143	0.9677679878		
26	pct_infection	01NOV2008	3	201	201	0.8434209176	0.92	0.9053607143	0.967300511		
27	pct_infection	01DEC2008	3	210	210	0.8447627329	0.95	0.9053607143	0.9659586956		
28	pct_infection	01JAN2009	3	220	220	0.8461559762	0.85	0.9053607143	0.9645654523		
29	pct_infection	01FEB2009	3	234	234	0.8479543721	0.91	0.9053607143	0.9627670565		
30	pct_infection	01MAR2009	3	244	244	0.8491430379	0.9	0.9053607143	0.9615783907		
31	pct_infection	01APR2009	3	259	259	0.8507952415	0.89	0.9053607143	0.959926187		
32	pct_infection	01MAY2009	3	270	270	0.8519183181	0.94	0.9053607143	0.9588031105		
33	pct_infection	01JUN2009	3	286	286	0.853434724	0.96	0.9053607143	0.9572867046	UPPER	1
34	pct_infection	01JUL2009	3	301	301	0.8547450948	0.97	0.9053607143	0.9559763338	UPPER	1

Table 2, Macro Output Data Set

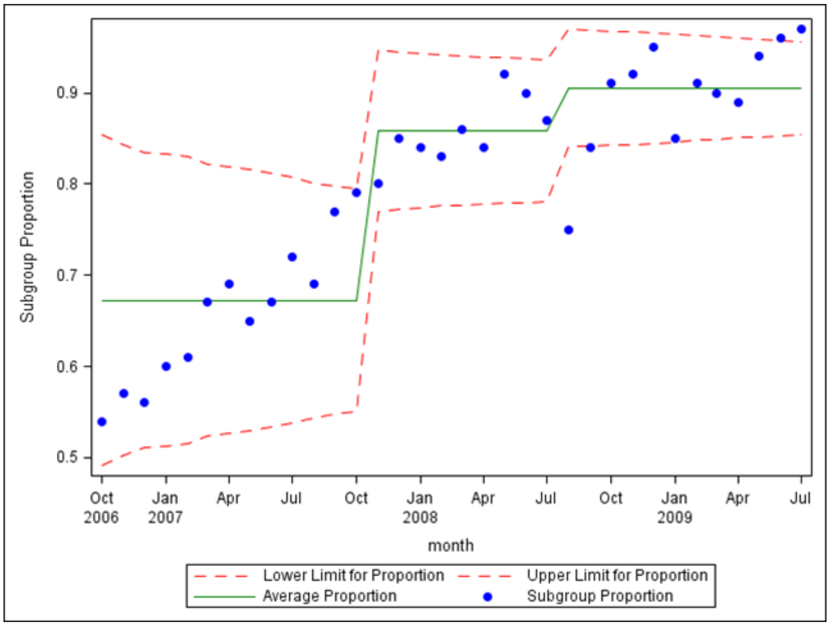

Figure 1, Graph of Control Chart Data from Table 2

CONCLUSION

Statistical Process Control charts were originally intended to help organizations reduce variation and maintain a centerline. However, quality improvement initiatives create out of control process changes to move centerlines in the desired direction. Hence, when using statistical process control charts for quality improvement, shifts (often multiple ones) in centerlines are common. Using this macro simplifies the programming and maintenance of these centerline shifts on p-charts. The spreadsheet provides a simple way to review and add new centerline shifts. Many organizations produce multiple charts based on different measures, sites or other groupings. The value provided by the macro is amplified in these situations. While this paper addresses p-charts only, this macro can be easily modified for other types of control charts as needed.

REFERENCES

Benneyan JC, Lloyd RC, Plsek PE. 2003. "Statistical Process Control as a Tool for Research and Healthcare Improvement." *Qual Safe Health Care*. 12:458-464.

CONTACT INFORMATION

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

Jesse Pratt, Colleen Mangeot, Kelly Olano
Cincinnati Children's Hospital Medical Center
3333 Burnet Avenue
MLC 5041
Cincinnati, OH 45229
513-803-3084
jesse.pratt_2@cchmc.org, colleen.mangeot@cchmc.org, kelly.olano@cchmc.org

APPENDIX

```
%macro cc_break(sheet=);

options merror mprint symbolgen spool;

* Create macro variable Nobs - Number of observations in breakptsexcel;

proc sql noprint;
      select count(*)
        into: Nobs
        from &sheet.;
quit;

* Loop through macro once for each observation;

%do j=1 %to &Nobs;

* Create macro variables for variables in breakptsexcel;
   data _null_;
      set &sheet.;
      if _N_=&j;
      call symputx('break_date',break_date);
      call symputx('DSN',dsn);
      call symputx('Yvar',Yvar);
      call symputx('Xvar',Xvar);
      call symputx('pts',pts);
      call symputx('dsnout',dsnout);
   run;

* Calculate number of break points - Create macro variable N;
   data _null_;
      num=countw("&break_date",' ');
      call symputx('N',num);
   run;

* Create a variable for each break point;
   %macro setdata;
      CC_data_&i
   %mend setdata;

   %macro breaks;
     %do i=1 %to &N;
        %let k=%sysevalf(&i+1);
     * Create file with everything in period i (up to next break date);
        data period_&i.;
           set &dsn;
           break_date&i=input(trim(scanq("&break_date",&i)),MMDDYY10.);
           break_date&k=input(trim(scanq("&break_date",&k)),MMDDYY10.);
           %if %sysevalf(&i) < %sysevalf(&N.) %then %do;
              if &XVAR ge BREAK_DATE&i and &XVAR lt BREAK_DATE&k.;
           %end;
           %else %do;
              if &XVAR ge BREAK_DATE&i;
           %end;
           format break_date&i break_date&k mmddyy10.;
```

```
* Create a macro variable with the number of observations in this file;
* This is needed to create the limits file below with the correct number of
observations in it;
          call symputx('nobs2',_n_);
       run;

* Calculate the centerline over baseline period for each i;
       proc sql;
          create table centerline_&i as
          select sum(&YVAR*YN)/sum(YN) as _P_
          %if %sysevalf(&i) < %sysevalf(&N.) %then %do;
             from period_&i. where &XVAR ge BREAK_DATE&i and &XVAR le
             min(intnx("&XVAR",BREAK_DATE&i,%sysevalf(&PTS-1),'b'),
             intnx("&XVAR",BREAK_DATE&k.,-1,'b'));
          %end;
          %else %do;
             from period_&i. where &XVAR ge BREAK_DATE&i and &XVAR le
             intnx("&XVAR",BREAK_DATE&i,%sysevalf(&PTS-1),'b');
          %end;
       quit;

* Create limit file with centerline and other variables needed;
       data lim_&i;
          set centerline_&i.;
          _VAR_="&YVAR";
          _SUBGRP_="&Xvar";
          _LIMITN_=.V;
          _TYPE_="ESTIMATE";
          do ii= 1 to &nobs2;
             output;
          end;
       run;

* Run Proc Shewhart;
       proc shewhart  data=period_&i limits=lim_&i;
          pchart &YVAR*&XVAR / tests=1,2,3
                               testnmethod=standardize
                               test2run=8
                               subgroupn=YN
                               dataunit=proportion
                               outtable=CC_data_&i
                               maxpanels=34
                               nochart;
       run;
    %end;

* Concatenate datasets;
    data &dsnout.;
       set
       %do i=1 %to &N;
          %setdata
       %end;
       ;
       rename _SUBN_=&YVAR._SUBN_
              _LCLP_=&YVAR._LCLP_
              _SUBP_=&YVAR._SUBP_
              _P_=&YVAR._P_
```

```
                _UCLP_=&YVAR._UCLP_;
      run;

%mend breaks;

%breaks;

%end;

%mend cc_break;
```