```
 1 *------------------------------------------------------------
   --*
 2 User:                   u63452984
 3 Date:                   07 January 2024
 4 Time:                   05:54:31
 5 Site:                   70094220
 6 Platform:               Linux
 7 Maintenance Release: 9.04.01M7P080620
 8 EM Version:             15.2
 9 *
10 *------------------------------------------------------------
   --*
11 * Training Log
12 Date:                   07 January 2024
13 Time:                   05:54:27
14 *------------------------------------------------------------
   --*
15 15241  proc freq data=EMWS4.VarClus_VariableSet noprint;
16 15242  table ROLE*LEVEL/out=WORK.VarClusMETA;
17 15243  run;
18 15244  proc print data=WORK.VarClusMETA label noobs;
19 15245  var ROLE LEVEL COUNT;
20 15246  label ROLE = "%sysfunc(sasmsg(sashelp.dmine, meta_ro
   le_vlabel, NOQUOTE))" LEVEL = "%sysfunc(sasmsg(sashelp.dmin
   e, meta_level_vlabel, NOQUOTE))" COUNT = "%sysfunc(sasmsg(s
   ashelp.dmine, rpt_count_vlabel, NOQUOTE))";
21 15247  title9 ' ';
22 15248  title10 "%sysfunc(sasmsg(sashelp.dmine, rpt_varSumma
   ry_title  , NOQUOTE))";
23 15249  run;
24 15250  title10;
25 15251  %let EMEXCEPTIONSTRING=;
26 PERFORMANCE  DETAILS
27 15595  *------------------------------------------------------
   ---------*;
28 15596  * VarClus: Generation of macros and macro variables;
```

```
29 15597  * To see the code generated, set the EM_DEBUG macro
   variable to SOURCE or _ALL_;
30 15598  *-------------------------------------------------
   ---------*;
31
32 15599  %let EMEXCEPTIONSTRING=;
33 15600  *-------------------------------------------------
   ---------*;
34 15601  * TRAIN: VarClus;
35 15602  *-------------------------------------------------
   ---------*;
36 15603  %let EM_ACTION = TRAIN;
37 15604  %let syscc = 0;
38 15605
39 15606  %macro main;
40 15607
41 15608     filename temp catalog 'sashelp.emexpl.variableclu
   stering_macros.source';
42 15609     %include temp;
43 15610     filename temp catalog 'sashelp.emexpl.variableclu
   stering_macros2.source';
44 15611     %include temp;
45 15612     filename temp;
46 15613
47 15614    %SetProperties;
48 15615
49 15616     %if %upcase(&EM_ACTION) = CREATE %then %do;
50 15617         filename temp catalog 'sashelp.emexpl.variabl
   eclustering_create.source';
51 15618         %include temp;
52 15619         filename temp;
53 15620         %create;
54 15621     %end;
55 15622     %else
56 15623     %if %upcase(&EM_ACTION) = TRAIN %then %do;
57 15624          filename temp catalog 'sashelp.emexpl.variab
```

```
   leclustering_train.source';
58 15625             %include temp;
59 15626             filename temp;
60 15627             %train;
61 15628     %end;
62 15629     %else
63 15630     %if %upcase(&EM_ACTION) = SCORE %then %do;
64 15631             filename temp catalog 'sashelp.emexpl.var
   iableclustering_score.source';
65 15632             %include temp;
66 15633             filename temp;
67 15634             %score;
68 15635     %end;
69 15636     %else
70 15637     %if %upcase(&EM_ACTION) = REPORT %then %do;
71 15638             filename temp catalog 'sashelp.emexpl.var
   iableclustering_report.source';
72 15639             %include temp;
73 15640             filename temp;
74 15641             %report;
75 15642     %end;
76 15643     /*
77 15644     %if %upcase(&EM_ACTION) = OPENTESTTABLE %then %do
   ;
78 15645         %put 'OPENING TABLE';
79 15646     %end;
80 15647     %if %upcase(&EM_ACTION) = CLOSETESTTABLE %then %d
   o;
81 15648         %put 'CLOSE TABLE';
82 15649     %end;
83 15650     */
84 15651  %mend main;
85 15652  %main;
86 NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMEXPL.V
   ARIABLECLUSTERING_MACROS.SOURCE.
87 15653 +
```

```
 88 15654 +/* Initialize property macro variables */
 89 15655 +%macro SetProperties;
 90 15656 +   %em_checkmacro(name=EM_PROPERTY_MAXCLUS,       gl
    obal=Y, value=DEFAULT);
 91 15657 +   %em_checkmacro(name=EM_PROPERTY_HIDEVARIABLE,  gl
    obal=Y, value=Y);
 92 15658 +   %em_checkmacro(name=EM_PROPERTY_PRINTOPTION,   gl
    obal=Y, value=SHORT);
 93 15659 +   %em_checkmacro(name=EM_PROPERTY_CLUSSOURCE,    gl
    obal=Y, value=CORR);
 94 15660 +   %em_checkmacro(name=EM_PROPERTY_CLUSCOMP,      gl
    obal=Y, value=PRINCIPAL);
 95 15661 +   %em_checkmacro(name=EM_PROPERTY_CLUSHIERACHY,
      global=Y, value=Y);
 96 15662 +   %em_checkmacro(name=EM_PROPERTY_INCLUDECLASSVAR,
         global=Y, value=N);
 97 15663 +   %em_checkmacro(name=EM_PROPERTY_EXPORTEDCOMP,
      global=Y, value=CLUSTERCOMP);
 98 15664 +   %em_checkmacro(name=EM_PROPERTY_MAXEIGEN,
     global=Y, value=DEFAULT);
 99 15665 +   %em_checkmacro(name=EM_PROPERTY_PROPORTION,
    global=Y, value=DEFAULT);
100 15666 +   %em_checkmacro(name=EM_PROPERTY_PRINTOPTION,
     global=Y, value=SHORT);
101 15667 +   %em_checkmacro(name=EM_PROPERTY_TWOSTAGECLUS,
      global=Y, value=AUTO);
102 15668 +   %em_checkmacro(name=EM_PROPERTY_SUPPRESSSAMPWARN,
          global=Y, value=N);
103 15669 +
104 15670 +%mend SetProperties;
105 15671 +
106 15672 +%Macro MakeDummyVariables(indata=,
107 15673 +                          outvar=,
108 15674 +                          outdata=,
109 15675 +                          fileref=,
110 15676 +                          recreatecmeta=N, /* option
```

```
    al */
111 15677 +                             incmeta=,     /* optional
    */
112 15678 +                             outcmeta=,    /* optional
    */
113 15679 +                             ndummyvars=_ndummyvars
114 15680 +                             );
115 15681 +    %global &ndummyvars;
116 15682 +
117 15683 +    proc dmdb batch data=&indata out=_dmdbdat dmdbca
    t=_dmdbcat classout=_classout;;
118 15684 +       class
119 15685 +       %EM_BINARY_INPUT %EM_NOMINAL_INPUT %EM_ORDINA
    L_INPUT
120 15686 +       %EM_BINARY_REJECTED %EM_NOMINAL_REJECTED %EM_
    ORDINAL_REJECTED
121 15687 +    ;
122 15688 +    run;
123 15689 +    %let &ndummyvars = 0;
124 15690 +    data _null_;
125 15691 +    %let dsid = %sysfunc(open(work._classout));
126 15692 +      %let &ndummyvars = %sysfunc(attrn(&dsid, NOBS)
    );
127 15693 +    %let dsid = %sysfunc(close(&dsid));
128 15694 +    run;
129 15695 +
130 15696 +    proc dmzip data=_dmdbdat dmdbcat=_dmdbcat;
131 15697 +       input
132 15698 +       %EM_BINARY_INPUT %EM_NOMINAL_INPUT %EM_ORDINA
    L_INPUT
133 15699 +       %EM_BINARY_REJECTED %EM_NOMINAL_REJECTED %EM_
    ORDINAL_REJECTED
134 15700 +        / level=nominal stdize=no;
135 15701 +       make outvar = &outvar;
136 15702 +       score data = &indata out =&outdata;
137 15703 +       code  file=  "&fileref";
```

```
138 15704 +    run;
139 15705 +    %if &recreatecmeta eq Y %then %do;
140 15706 +    proc contents data =&outvar out=_tmpds(keep=NAME
     LABEL);
141 15707 +    data _tmpds;
142 15708 +        set _tmpds;
143 15709 +           ROLE = 'INPUT';
144 15710 +           LEVEL = 'INTERVAL';
145 15711 +           CREATOR='DMZIP';
146 15712 +           if NAME = '_TYPE_' then delete;
147 15713 +    run;
148 15714 +    data &outcmeta;
149 15715 +         set &incmeta _tmpds;
150 15716 +    run;
151 15717 +    %end;
152 15718 +    proc datasets lib=work nolist;
153 15719 +      delete  _dmdbdat _dmdbcat _classout
154 15720 +    %if &recreatecmeta eq Y %then %do;
155 15721 +    _tmpds
156 15722 +    %end;
157 15723 +    ;
158 15724 +    quit;
159 15725 +%Mend MakeDummyVariables;
160 15726 +
161 15727 +/*--- Determine Optimal Number of Cluster ----
162 15728 +%macro FindClusNum(statds=, groupds=, minvariation=)
    ;
163 15729 +   %global optnclus;
164 15730 +   data varclus_tmp(drop=_NAME_);
165 15731 +      set &statDs;
166 15732 +      where _type_ ='PROPOR';
167 15733 +   run;
168 15734 +   proc sort data=varclus_tmp;
169 15735 +      by _NCL_;
170 15736 +   run;
171 15737 +   proc transpose data=varclus_tmp out=varclus_tmp;
```

```
172 15738 +      by _NCL_;
173 15739 +      var %EM_INTERVAL_INPUT
174 15740 +      %if &EM_PROPERTY_INCLUDECLASSVAR eq Y %then %d
    o;
175 15741 +      %let dsid = %sysfunc(open(&EM_USER_OUTDUMMY));
176 15742 +      %let nvar = %sysfunc(attrn(&dsid, NVAR));
177 15743 +          %do i = 2 %to &nvar;
178 15744 +          %let varname = %sysfunc(varname(&dsid, &i)
    );
179 15745 +          &varname
180 15746 +          %end;
181 15747 +      %end;
182 15748 +   ;
183 15749 +   run;
184 15750 +
185 15751 +   %if &minVariation eq %then %do;
186 15752 +       %let minVariation = &EM_PROPERTY_MINVARIATION
    ;
187 15753 +   %end;
188 15754 +   %if ^(0<&minVariation<100) %then %do;
189 15755 +       %let minVariation = 90;
190 15756 +   %end;
191 15757 +
192 15758 +   data _null_;
193 15759 +      set varclus_tmp end=eof;
194 15760 +      by _NCL_;
195 15761 +      retain flag 0;
196 15762 +      if first._ncl_ then flag=0;
197 15763 +      if .<col1 < &minVariation then flag=1;
198 15764 +      if last._ncl_ and ^flag then do;
199 15765 +         call symput('OPTNCL', _ncl_);
200 15766 +         stop;
201 15767 +      end;
202 15768 +     if eof then call symput('OPTNCL', _ncl_);
203 15769 +   run;
204 15770 +
```

```
205 15771 +   %let optnclus = &OPTNCL;
206 15772 +
207 15773 +   data varclus_tmp(drop=_NCL_ _NAME_);
208 15774 +      set &statDs;
209 15775 +      where _type_ in('RSQUARED' 'GROUP') and _NCL_=
    &OPTNCL;
210 15776 +   run;
211 15777 +   proc sort data=varclus_tmp;
212 15778 +      by _TYPE_;
213 15779 +   run;
214 15780 +   proc transpose data=varclus_tmp out=varclus_tmp;
215 15781 +      by _TYPE_;
216 15782 +   run;
217 15783 +   proc sort data=varclus_tmp;
218 15784 +      by _name_ _type_;
219 15785 +   run;
220 15786 +
221 15787 +  proc transpose data=varclus_tmp out=&groupds;
222 15788 +      by _NAME_;
223 15789 +   run;
224 15790 +   proc sort data=&groupDs(rename=(col1=Cluster col2
    =Rsquare _NAME_=VARIABLE));
225 15791 +      by Cluster descending Rsquare;
226 15792 +      where Cluster ne 0;
227 15793 +   run;
228 15794 +   proc datasets lib=work nolist mt=(DATA VIEW);
229 15795 +      delete varclus_tmp;
230 15796 +   run;
231 15797 +   quit;
232 15798 +%mend findClusNum;
233 15799 +*/
234 15800 +
235 15801 +%macro getNclusfromTrain(inoutstat=, nc=);
236 15802 +%global &nc;
237 15803 +data _null_;
238 15804 +    set &inoutstat end=eof;
```

```
239 15805 +    if eof then do;
240 15806 +    call symput("&nc", _ncl_);
241 15807 +    end;
242 15808 +run;
243 15809 +%mend  getNclusfromTrain;
244 15810 +
245 15811 +%macro MakeDeltaCode(groupds=, outstatscore=, deltac
    odefile=);
246 15812 +
247 15813 +     *--- Build Code to Modify Metadata ---*;
248 15814 +     filename X "&deltacodefile";
249 15815 +     data _null_;
250 15816 +        FILE X;
251 15817 +        set &groupds end=eof;
252 15818 +        /*by Cluster;*/
253 15819 +         if _N_=1 then do;
254 15820 +           %if &EM_PROPERTY_INCLUDECLASSVAR eq Y %th
    en %do;
255 15821 +            put "if upcase(strip(ROLE)) ='INPUT' and
     upcase(strip(LEVEL)) ^='INTERVAL' then ROLE ='REJECTED' ;"
    ;
256 15822 +           %end;
257 15823 +           put "if upcase(strip(ROLE))='INPUT' and u
    pcase(strip(LEVEL))='INTERVAL' then do;";
258 15824 +           put "if upcase(strip(NAME)) in (";
259 15825 +        end;
260 15826 +        if Strip(upcase(Selected)) eq 'YES' then do;
261 15827 +           string = '"'!!trim(left(VARIABLE))!!'"';
262 15828 +           put string;
263 15829 +        end;
264 15830 +        if eof then do;
265 15831 +           put ') then ROLE="INPUT";';
266 15832 +           put 'else ROLE="REJECTED";';
267 15833 +           put 'end;';
268 15834 +
269 15835 +           %if %upcase(&EM_PROPERTY_HIDEVARIABLE) eq
```

```
     Y %then %do;
270  15836 +             put 'if upcase(strip(ROLE)) = "REJECTED
     " then delete ;';
271  15837 +           %end;
272  15838 +        end;
273  15839 +     run;
274  15840 +     quit;
275  15841 +
276  15842 +     filename X;
277  15843 +
278  15844 +     quit;
279  15845 +%mend MakeDeltaCode;
280  15846 +
281  15847 +%macro MakeVarClusCorrData(statds=, corrds=, corrplo
     tds= );
282  15848 +    %if ^%sysfunc(exist(&statds)) %then %do;
283  15849 +         %goto doendc;
284  15850 +    %end;
285  15851 +
286  15852 +    data &corrds(drop=_TYPE_ _NCL_) ;
287  15853 +       set &statds;
288  15854 +       where _type_ eq 'CORR' ;
289  15855 +    run ;
290  15856 +    proc sort data=&corrds;
291  15857 +       by _NAME_ ;
292  15858 +    run ;
293  15859 +    proc transpose data=&corrds out=&corrplotds name
     =_TMP_;
294  15860 +      BY _NAME_ ;
295  15861 +    run ;
296  15862 +    data &corrplotds;
297  15863 +       length _Y_ $100;
298  15864 +       set &corrplotDs;
299  15865 +       if _LABEL_ ne '' then _Y_=_LABEL_ ; else _Y_=
     _TMP_ ;
300  15866 +    run ;
```

```
301  15867 +    data varclus_match(rename=(_TMP_= _NAME_ _LABEL_
     =_X_));
302  15868 +       set &corrplotds;
303  15869 +       where _LABEL_ ne '' ;
304  15870 +       keep _TMP_ _LABEL_ ;
305  15871 +    run ;
306  15872 +    data _null_;
307  15873 +       nobs=0;
308  15874 +       dsid = open('varclus_match');
309  15875 +       if dsid then do;
310  15876 +          nobs = attrn(dsid, 'NOBS');
311  15877 +          dsid = close(dsid);
312  15878 +       end;
313  15879 +       call symput ('CORR_NOBS', nobs);
314  15880 +    run;
315  15881 +    %if &corr_nobs %then %do;
316  15882 +        proc sort data=varclus_match;
317  15883 +           by _name_;
318  15884 +        run ;
319  15885 +        proc sort data=&corrplotds;
320  15886 +           by _name_;
321  15887 +        run ;
322  15888 +        data &corrplotds(keep= _X_ _Y_ col1 rename=(
     col1=Correlation)) ;
323  15889 +           merge varclus_match &corrplotds;
324  15890 +           by _NAME_ ;
325  15891 +           if _X_ eq '' then _X_=_NAME_ ;
326  15892 +           label _X_ = "%sysfunc(sasmsg(sashelp.dmin
     e, rpt_varclus_label_variable, noquote))";
327  15893 +           label _Y_ = "%sysfunc(sasmsg(sashelp.dmin
     e, rpt_varclus_label_variable, noquote))";
328  15894 +           label col1 = "%sysfunc(sasmsg(sashelp.dmi
     ne, rpt_correlation_vlabel, noquote))";
329  15895 +
330  15896 +       run ;
331  15897 +    %end;
```

```
332  15898 +    %else %do;
333  15899 +        proc sort data=&corrplotds;
334  15900 +           by _name_;
335  15901 +        run ;
336  15902 +        data &corrplotds(keep= _NAME_ _Y_ col1 renam
     e=(_NAME_=_X_ col1=Correlation)) ;
337  15903 +           set &corrplotds;
338  15904 +           label _NAME_ = "%sysfunc(sasmsg(sashelp.d
     mine, rpt_varclus_label_variable, noquote))" ;
339  15905 +           label _Y_ = "%sysfunc(sasmsg(sashelp.dmin
     e, rpt_varclus_label_variable, noquote))" ;
340  15906 +           label col1 = "%sysfunc(sasmsg(sashelp.dmi
     ne, rpt_correlation_vlabel, noquote))";
341  15907 +
342  15908 +        run ;
343  15909 +    %end;
344  15910 +    proc sort data=&corrplotds;
345  15911 +       by _X_ _Y_;
346  15912 +    run ;
347  15913 +    proc datasets lib=work nolist mt=(DATA VIEW);
348  15914 +       delete varclus_match;
349  15915 +    run;
350  15916 +    quit;
351  15917 +
352  15918 +%doendc:
353  15919 +
354  15920 +%mend MakeVarClusCorrData;
355  15921 +
356  15922 +%macro MakeStatPlotData(statds= , outstatplotds=);
357  15923 +   %if %sysfunc(exist(&statds)) %then %do;
358  15924 +
359  15925 +       data varclus_tmp(drop=_NAME_ _NCL_) ;
360  15926 +          set &statDs;
361  15927 +          where _type_ in('MEAN', 'STD', 'N');
362  15928 +       run ;
363  15929 +       proc transpose data=varclus_tmp out=&outstatp
```

```
    lotds;
364 15930 +          id _TYPE_ ;
365 15931 +       run ;
366 15932 +       data &outstatplotds;
367 15933 +          set &outstatplotds(obs=1000);
368 15934 +          label _name_= "%sysfunc(sasmsg(sashelp.dmi
    ne, rpt_varclus_label_name, noquote))";
369 15935 +          label _label_="%sysfunc(sasmsg(sashelp.dmi
    ne, rpt_varclus_label_label, noquote))" ;
370 15936 +          if MEAN ne 0 then SCALEDSTD= STD / MEAN ;
371 15937 +          else SCALEDSTD= STD ;
372 15938 +          label SCALEDSTD = "%sysfunc(sasmsg(sashelp
    .dmine, rpt_varclus_label_scaledstd, noquote))";
373 15939 +       run ;
374 15940 +       proc sort data=&outstatplotds;
375 15941 +          by descending SCALEDSTD ;
376 15942 +       run ;
377 15943 +       proc datasets lib=work nolist mt=(DATA VIEW);
378 15944 +          delete varclus_tmp;
379 15945 +       run;
380 15946 +       quit;
381 15947 +    %end;
382 15948 +
383 15949 +%mend MakeStatPlotData;
384 15950 +
385 15951 +
386 15952 +%macro CreateScoreCode(indata=, ncluscomp=, fileref=
    );
387 15953 +     %EM_GETNAME(KEY=OUTSTATSCORE, type=DATA);
388 15954 +    data &EM_USER_OUTSTATSCORE;
389 15955 +          set &indata;
390 15956 +          if (_TYPE_ in ('SCORE' 'MEAN' 'STD') and _
    NCL_ = &ncluscomp ) or (_TYPE_ in ('MEAN' 'STD'));
391 15957 +          if _TYPE_ = 'MEAN' then _NAME_='MEAN';
392 15958 +          if _TYPE_ = 'STD' then _NAME_='STD';
393 15959 +          DROP _TYPE_ _NCL_;
```

```
394 15960 +     run;
395 15961 +
396 15962 +     filename _file_  "&fileRef";
397 15963 +
398 15964 +     data _null_;
399 15965 +        FILE _file_ MOD;
400 15966 +        put ' ';
401 15967 +        put '/*-----------------------------------
    ------------*/';
402 15968 +        put '/* ' "%sysfunc(sasmsg(sashelp.dmine, rp
    t_varclus_score_title_begin , noquote))" '*/';
403 15969 +        put '/*-----------------------------------
    ------------*/';
404 15970 +        put ' ';
405 15971 +        %let dsid = %sysfunc(open(&EM_USER_OUTSTATSC
    ORE));
406 15972 +
407 15973 +        %let nvar = %sysfunc(attrn(&dsid, NVAR));
408 15974 +        %let vn_name =%sysfunc(varnum(&dsid, _NAME_)
    );
409 15975 +
410 15976 +        %let k = 1;
411 15977 +        %do %while(^%sysfunc(fetch(&dsid)));
412 15978 +                %let _name = %sysfunc(getvarc(&dsid,
     &vn_name));
413 15979 +                %if &k > 2 %then %do;
414 15980 +                 %let cn = %eval(&k-2);
415 15981 +                 put "&_name = 0 ; /*---" "%sysfunc(
    sasmsg(sashelp.dmine, rpt_varclus_score_cluscompnum, noquot
    e, &cn))"  "------ */";
416 15982 +                %end;
417 15983 +                %let k = %eval(&k+1);
418 15984 +        %end;
419 15985 +
420 15986 +        %let rc = %sysfunc(rewind(&dsid));
421 15987 +
```

```
422  15988 +        %do i= 2 %to &nvar;
423  15989 +            %let _varname =  %sysfunc(varname(&dsid,
      &i));
424  15990 +            %do %while(^%sysfunc(fetch(&dsid)));
425  15991 +                %let _name = %sysfunc(getvarc(&dsid,
      &vn_name));
426  15992 +                %if &_name = MEAN %then
427  15993 +                %let _mean = %sysfunc(getvarn(&dsid,
      &i));
428  15994 +                %else %if &_name = STD %then
429  15995 +                %let _std = %sysfunc(getvarn(&dsid,
     &i));
430  15996 +                %else %do;
431  15997 +                      %let coeff =  %sysfunc(getvarn
     (&dsid, &i));
432  15998 +                      %let abscoeff = %sysfunc(abs(&
     coeff));
433  15999 +                         %if &abscoeff >  0 %then %
     do;
434  16000 +                          put "&_name = &_name+&coeff *
      (&_varname - &_mean)/&_std;";
435  16001 +                           %end;
436  16002 +                %end;
437  16003 +             %end;
438  16004 +             %let rc = %sysfunc(rewind(&dsid));
439  16005 +         %end;
440  16006 +
441  16007 +        %let dsid= %sysfunc(close(&dsid));
442  16008 +       run;
443  16009 +%mend CreateScoreCode;
444  16010 +
445  16011 +
446  16012 +
447  16013 +/*-------------------------------------------------
     --------
448  16014 +    Instead of using %MakeRSquareData,
```

```
449 16015 +    %MakeVarClusResultTable at macro2.source is used
450 16016 + +-----------------------------------------------
    --------*/
451 16017 +
452 16018 +
453 16019 +%macro MakeRSquareData(indata=, inClusRSquare=, outd
    ata=, ncluster=);
454 16020 +
455 16021 +/* modifying  from ods rsquare = data */
456 16022 +
457 16023 +data &outdata(drop= ControlVar  NumberOfClusters Cur
    rentCluster);
458 16024 +    Length Cluster $16;
459 16025 +    length Variable $32;
460 16026 +    Length VariableLabel $64;
461 16027 +    set &indata; retain CurrentCluster;
462 16028 +    if NumberOfClusters ^= &ncluster then delete;
463 16029 +    if strip(Cluster) eq '' then Cluster = CurrentCl
    uster;
464 16030 +    CurrentCluster = Cluster;
465 16031 +   run;
466 16032 +proc sort data =&outdata ;
467 16033 +     by Cluster RsquareRatio;
468 16034 +run;
469 16035 +data _tmprsq(drop=index);
470 16036 +     set &outdata; by Cluster;
471 16037 +     if first.Cluster then do;
472 16038 +     index = strip(scan(Cluster,2 ));
473 16039 +     Variable = "Clus"||index;
474 16040 +     VariableLabel = "Cluster Component "||index;
475 16041 +     OwnCluster = 1;
476 16042 +     NextClosest = .;
477 16043 +     RsquareRatio = 0;
478 16044 +     output;
479 16045 +     end;
480 16046 +run;
```

```
481  16047 +
482  16048 +proc sort data = _tmprsq ;
483  16049 +     by Cluster RsquareRatio;
484  16050 +run;
485  16051 +data &outdata;
486  16052 +     set &outdata _tmprsq;
487  16053 +by Cluster;
488  16054 +run;
489  16055 +
490  16056 +
491  16057 +/* Just create the Selected variable with all YES */
492  16058 +
493  16059 +data &outdata;
494  16060 +    set &outdata; by cluster;
495  16061 +    length Selected $8;
496  16062 +    Selected = 'YES';
497  16063 +    label  OwnCluster = 'R-Sqaure with Cluster Compo
     nent';
498  16064 +    label  NextClosest = 'R-Sqaure with Next Cluster
      Component';
499  16065 +    rename OwnCluster = RSqWithClusterComp;
500  16066 +    rename NextClosest = RSqWithNextClusComp;
501  16067 +run;
502  16068 +
503  16069 +
504  16070 +/* Selected = Y/N will be done %score section -----
505  16071 +
506  16072 +%if &EM_PROPERTY_EXPORTEDCOMP ne CLUSTERCOMP %then %
     do;
507  16073 +data &outdata;
508  16074 +    set &outdata; by cluster;
509  16075 +    length Selected $8;
510  16076 +    if first.Cluster then Selected = 'YES';
511  16077 +    else Selected = 'NO';
512  16078 +    label  OwnCluster = 'R-Sqaure with Cluster Compo
     nent';
```

```
513 16079 +    label  NextClosest = 'R-Sqaure with Next Cluster
     Component';
514 16080 +    rename OwnCluster = RSqWithClusterComp;
515 16081 +    rename NextClosest = RSqWithNextClusComp;
516 16082 +run;
517 16083 +%end;
518 16084 +%else %do;
519 16085 +data &outdata;
520 16086 +    set &outdata; by cluster;
521 16087 +    if last.Cluster then Selected = 'YES';
522 16088 +    else Selected = 'NO';
523 16089 +    label  OwnCluster = 'R-Sqaure with Cluster Compo
    nent';
524 16090 +    label  NextClosest = 'R-Sqaure with Next Cluster
     Component';
525 16091 +    rename OwnCluster = RSqWithClusterComp;
526 16092 +    rename NextClosest = RSqWithNextClusComp;
527 16093 +run;
528 16094 +%end;
529 16095 +-----------------------------------------------------
    -----------*/
530 16096 +
531 16097 +%if %sysfunc(exist(&inClusRSquare)) %then %do;
532 16098 +/* to calculate NextClosestClusRsq */
533 16099 +proc transpose data = &inClusRSquare  out=_clusRsq;
534 16100 +      by cluster;
535 16101 +      run;
536 16102 +data _clusRsq;
537 16103 +     set _clusRsq;
538 16104 +     if strip(upcase(Cluster)) eq strip(upcase(_NAME
    _)) then delete;
539 16105 +run;
540 16106 +
541 16107 +proc sort data=_clusRsq;
542 16108 +    by cluster col1;
543 16109 +    run;
```

```
544 16110 +data _clusRsq(drop=_NAME_ _LABEL_);
545 16111 +     set _clusRsq; by cluster;
546 16112 +     if last.Cluster then output;
547 16113 +     label  COL1 = 'R-Sqaure with Next Cluster Compo
    nent';
548 16114 +     rename COL1 = RSqWithNextClusComp;
549 16115 +     rename Cluster = Variable;
550 16116 +     label  Cluster = "Variable";
551 16117 +run;
552 16118 +
553 16119 +proc sort data =&outdata;
554 16120 +     by Variable;
555 16121 +run;
556 16122 +data &outdata;
557 16123 +    merge &outdata _clusRsq;
558 16124 +    by Variable;
559 16125 +run;
560 16126 +proc sort data =&outdata;
561 16127 +by Cluster  RsquareRatio;
562 16128 +run;
563 16129 +quit;
564 16130 +%end;
565 16131 +
566 16132 +proc datasets lib = work nolist;
567 16133 +     delete _tmprsq _clusRsq;
568 16134 +     run;
569 16135 +quit;
570 16136 +
571 16137 +%mend MakeRSquareData;
572 16138 +
573 16139 +
574 16140 +/*-----------------------------------------------
    ----------------------*/
575 16141 +
576 16142 +
577 16143 +
```

```
578  16144 +%macro ModifyCorr(indata=,
579  16145 +                 outdata=,
580  16146 +                 rsquare = Y
581  16147 +                 );
582  16148 +    data corr_tmp;
583  16149 +        set &indata;
584  16150 +    run;
585  16151 +     proc sql;
586  16152 +          update &indata
587  16153 +          set
588  16154 +     %let dsid = %sysfunc(open(work.corr_tmp));
589  16155 +     %let nvar = %sysfunc(attrn(&dsid, NVAR));
590  16156 +          %do i = 4 %to &nvar;
591  16157 +          %let _name = %sysfunc(varname(&dsid, &i));
592  16158 +             %if &rsquare eq Y %then %let _name_md =
      &_name.**2;
593  16159 +           %else  %let _name_md = &_name;
594  16160 +           %if &i < &nvar %then %do;
595  16161 +              &_name = 1- &_name_md ,
596  16162 +           %end;
597  16163 +           %else %do;
598  16164 +              &_name = &_name_md where _TYPE_ conta
     ins 'CORR' ;
599  16165 +           %end;
600  16166 +          %end;
601  16167 +      %let dsid= %sysfunc(close(&dsid));
602  16168 +
603  16169 +      select * from &indata;
604  16170 +      run;
605  16171 +      proc datasets lib = work nolist;
606  16172 +           delete corr_tmp;
607  16173 +      run;
608  16174 +      quit;
609  16175 +
610  16176 + %mend ModifyCorr;
611  16177 +
```

```
612 16178 + %macro MakeClusStructCorrData(indata=,outdata=, ncl
    uster=, Rsquare=N);
613 16179 +    data &outdata(drop= _NCL_  _TYPE_);
614 16180 +       set &indata;
615 16181 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
    E_) eq 'STRUCTUR') then delete;
616 16182 +       rename _NAME_ = Cluster;
617 16183 +       label _NAME_ = "%sysfunc(sasmsg(sashelp.dmine
    , rpt_varclus_label_clustername,  noquote))";
618 16184 +    run;
619 16185 +    %if &RSquare eq Y %then %do;
620 16186 +     data corr_tmp;
621 16187 +        set &outdata;
622 16188 +     run;
623 16189 +
624 16190 +     data &outdata(drop=i);
625 16191 +          set &outdata;
626 16192 +          %let dsid = %sysfunc(open(work.corr_tmp));
627 16193 +          %let nvar = %sysfunc(attrn(&dsid, NVAR));
628 16194 +          %do i = 2 %to &nvar;
629 16195 +            %let _name = %sysfunc(varname(&dsid, &i)
    );
630 16196 +            %let _name_md = &_name.**2;
631 16197 +                &_name = &_name_md;
632 16198 +          %end;
633 16199 +      %let dsid= %sysfunc(close(&dsid));
634 16200 +      run;
635 16201 +      proc datasets lib = work nolist;
636 16202 +           delete corr_tmp;
637 16203 +      run;
638 16204 +
639 16205 +    %end;
640 16206 +     quit;
641 16207 +%mend MakeClusStructCorrData;
642 16208 +
643 16209 +%macro MakeInterClusCorrData(indata=, outdata=, nclu
```

```
    ster=, RSquare=N, makeplotds=N, plotds=);
644 16210 +    data &outdata(drop= _NCL_  _TYPE_);
645 16211 +       set &indata;
646 16212 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
    E_) eq 'CCORR') then delete;
647 16213 +       rename _NAME_ = Cluster;
648 16214 +       label _NAME_ = "%sysfunc(sasmsg(sashelp.dmine
    , rpt_varclus_label_clustername,  noquote))";
649 16215 +    run;
650 16216 +    data corr_tmp;
651 16217 +        set &outdata;
652 16218 +    run;
653 16219 +
654 16220 +    %let dsid = %sysfunc(open(work.corr_tmp));
655 16221 +    %let nclus2= %eval(&ncluster+1);
656 16222 +    data &outdata;
657 16223 +         set &outdata;
658 16224 +          %do i = 2 %to &nclus2;
659 16225 +          %let i_1 = %eval(&i-1);
660 16226 +            %let _name = %sysfunc(varname(&dsid, &i)
    );
661 16227 +            %let _newName = Clus&i_1;
662 16228 +                rename &_name = &_newName; ;
663 16229 +                *label &_name ="Cluster &i_1";
664 16230 +                label &_name = "%sysfunc(sasmsg(sash
    elp.dmine, rpt_varclus_label_clusternum, noquote,  &i_1))";
665 16231 +          %end;
666 16232 +          keep Cluster
667 16233 +          %do i = 2 %to &nclus2;
668 16234 +                %let _name = %sysfunc(varname(&dsid,
     &i));
669 16235 +                &_name
670 16236 +          %end;
671 16237 +          ;
672 16238 +    %let dsid= %sysfunc(close(&dsid));
673 16239 +     run;
```

```
674  16240 +     quit;
675  16241 +
676  16242 +    %if &RSquare eq Y %then %do;
677  16243 +
678  16244 +       data corr_tmp;
679  16245 +        set &outdata;
680  16246 +     run;
681  16247 +
682  16248 +     data &outdata(drop=i);
683  16249 +          set &outdata;
684  16250 +          %let dsid = %sysfunc(open(work.corr_tmp));
685  16251 +          %let nvar = %sysfunc(attrn(&dsid, NVAR));
686  16252 +          %do i = 2 %to &nvar;
687  16253 +            %let _name = %sysfunc(varname(&dsid, &i)
     );
688  16254 +            %let _name_md = &_name.**2;
689  16255 +                &_name = &_name_md;
690  16256 +          %end;
691  16257 +      %let dsid= %sysfunc(close(&dsid));
692  16258 +      run;
693  16259 +    %end;
694  16260 +
695  16261 +    %if &makeplotds eq Y %then %do;
696  16262 +     proc transpose data = &outdata
697  16263 +          out=&plotds(drop=_LABEL_ rename=(_NAME_ =
     Y Cluster=X Col1= Correlation));
698  16264 +          by cluster;
699  16265 +     run;
700  16266 +     data &plotds;
701  16267 +          set  &plotds;
702  16268 +          label x="%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_label_cluster, noquote))";
703  16269 +          label Y="%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_label_cluster, noquote))";
704  16270 +     run;
705  16271 +     %end;
```

```
706 16272 +     proc datasets lib = work nolist;
707 16273 +           delete corr_tmp;
708 16274 +     run;
709 16275 +     quit;
710 16276 +%mend MakeInterClusCorrData;
711 16277 +
712 16278 +
713 16279 +%macro MakeClusConstellData(indata=, outlink=, outno
    de=);
714 16280 +
715 16281 +data &outlink(drop = Selected);
716 16282 +     set &indata;
717 16283 +     LINKID = _N_;
718 16284 +     label LINKID = "%sysfunc(sasmsg(sashelp.dmine,
    rpt_varclus_label_linkid,  noquote))";
719 16285 +     if strip(upcase(Cluster)) eq strip(upcase(Varia
    ble)) then Variable = ClosestCluster;
720 16286 +run;
721 16287 +data &outnode(keep=NODEID TYPE LABEL);
722 16288 +    set &indata;
723 16289 +    length TYPE $16;
724 16290 +    rename Variable = NODEID;
725 16291 +    label  Variable= "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_nodeidvar, noquote))";
726 16292 +    if strip(upcase(Cluster)) eq strip(upcase(Variab
    le))
727 16293 +    then  TYPE = "CLUSTER";
728 16294 +    else  TYPE="VARIABLE";
729 16295 +    label TYPE = "%sysfunc(sasmsg(sashelp.dmine, rpt
    _varclus_label_nodetype, noquote))";
730 16296 +run;
731 16297 +quit;
732 16298 +%mend MakeClusConstellData;
733 16299 +
734 16300 +
735 16301 +
```

```
736 16302 +%macro MakeClusConstellData(indata=, outlink=, outno
    de=);
737 16303 +
738 16304 +data &outlink(drop = Selected);
739 16305 +     set &indata;
740 16306 +     LINKID = _N_;
741 16307 +     label LINKID = "%sysfunc(sasmsg(sashelp.dmine,
    rpt_varclus_label_linkid,  noquote))";
742 16308 +     if strip(upcase(Cluster)) eq strip(upcase(Varia
    ble)) then Variable = ClosestCluster;
743 16309 +run;
744 16310 +data &outnode(keep=NODEID TYPE LABEL);
745 16311 +    set &indata;
746 16312 +    length TYPE $16;
747 16313 +    rename Variable = NODEID;
748 16314 +    label  Variable= "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_nodeidvar, noquote))";
749 16315 +    if strip(upcase(Cluster)) eq strip(upcase(Variab
    le))
750 16316 +    then  TYPE = "CLUSTER";
751 16317 +    else  TYPE="VARIABLE";
752 16318 +    label TYPE = "%sysfunc(sasmsg(sashelp.dmine, rpt
    _varclus_label_nodetype, noquote))";
753 16319 +run;
754 16320 +quit;
755 16321 +%mend MakeClusConstellData;
756 16322 +
757 16323 +
758 16324 +/*--- This will work only when inds is not a view da
    ta -------
759 16325 +
760 16326 +%macro getNVarNObs(inds=, nvar=, nobs=);
761 16327 +
762 16328 +    %global &nvar;
763 16329 +    %global &nobs;
764 16330 +    data _null_;
```

```
765  16331 +         dsid = open("&inds");
766  16332 +         nv = attrn(dsid, 'NVAR');
767  16333 +         no = attrn(dsid, 'NOBS');
768  16334 +         dsid = close(dsid);
769  16335 +         call symput("&nvar", nv);
770  16336 +         call symput("&nobs", no);
771  16337 +    run;
772  16338 +    quit;
773  16339 +%mend  getNVarNObs;
774  16340 +
775  16341 ++------------------------------------------------
     ------------*/
776  16342 +
777  16343 +
778  16344 +%macro getNVar(inds=, nvar=);
779  16345 +    %global &nvar;
780  16346 +    data _null_;
781  16347 +         dsid = open("&inds");
782  16348 +         nv = attrn(dsid, 'NVAR');
783  16349 +         dsid = close(dsid);
784  16350 +         call symput("&nvar", nv);
785  16351 +    run;
786  16352 +    quit;
787  16353 +%mend  getNVar;
788  16354 +
789  16355 +
790  16356 +
791  16357 +%macro getNObs(inds=, nobs=);
792  16358 +    %global &nobs;
793  16359 +    data _null_;
794  16360 +        set &inds end=eof;
795  16361 +        if eof then call symput("&nobs", _N_);
796  16362 +    run;
797  16363 +    quit;
798  16364 +%mend  getNObs;
799  16365 +
```

```
800  16366 +%Macro CreateVarclusMeta(trainnum=);
801  16367 +   %EM_GETNAME(KEY=VARCLUSMETA, TYPE=DATA);
802  16368 +    data &EM_USER_VARCLUSMETA;
803  16369 +         length TrainNum 8.;
804  16370 +         length NewTrain $8;
805  16371 +         length NGCluster 8.;
806  16372 +         length ExportedComp $16;
807  16373 +         length HideVariable $8;
808  16374 +         TrainNum = &trainnum;
809  16375 +         NewTrain = "Y";
810  16376 +         ExportedComp = "&EM_PROPERTY_EXPORTEDCOMP";
811  16377 +         HideVariable = "&EM_PROPERTY_HIDEVARIABLE";
812  16378 +         NGCluster = 0; /* zero means no twostage */
813  16379 +   run;
814  16380 +   quit;
815  16381 +%mend CreateVarclusMeta;
816  NOTE: %INCLUDE (level 1) ending.
817  NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMEXPL.V
     ARIABLECLUSTERING_MACROS2.SOURCE.
818  16382 +
819  16383 +%macro MakeInterClusCorrData(indata=, outdata=, nclu
     ster=, globalclusid=, RSquare=N, makeplotds=N, plotds=);
820  16384 +    data &outdata(drop= _NCL_  _TYPE_);
821  16385 +       set &indata;
822  16386 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
     E_) eq 'CCORR') then delete;
823  16387 +       %if &globalclusid ne %then %do;
824  16388 +       _NAME_ = "GC&globalclusid._"||upcase(_NAME_);
825  16389 +       rename _NAME_ = Cluster;
826  16390 +       %end;
827  16391 +       %else %do;
828  16392 +        _NAME_ = upcase(_NAME_);
829  16393 +       rename _NAME_ = Cluster;
830  16394 +       %end;
831  16395 +       label _NAME_ = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_clustername,  noquote))";
```

```
832 16396 +    run;
833 16397 +    data corr_tmp;
834 16398 +        set &outdata;
835 16399 +    run;
836 16400 +
837 16401 +    %let dsid = %sysfunc(open(work.corr_tmp));
838 16402 +    %let nclus2= %eval(&ncluster+1);
839 16403 +    data &outdata;
840 16404 +         set &outdata;
841 16405 +          %do i = 2 %to &nclus2;
842 16406 +          %let i_1 = %eval(&i-1);
843 16407 +            %let _name = %sysfunc(varname(&dsid, &i)
    );
844 16408 +            %if &globalclusid ne %then
845 16409 +                %do; %let _newName = GC&globalclusid
    ._CLUS&i_1;
846 16410 +                     rename &_name = &_newName;
847 16411 +                     *label &_name ="GC &globalclusi
    d : Cluster &i_1";
848 16412 +                     label &_name = "%sysfunc(sasmsg
    (sashelp.dmine, rpt_varclus_label_gc_clusternum, noquote,
    &globalclusid, &i_1))";
849 16413 +                %end;
850 16414 +            %else
851 16415 +                %do; %let _newName = CLUS&i_1;
852 16416 +                     rename &_name = &_newName;
853 16417 +                     *label &_name ="Cluster &i_1";
854 16418 +                     label &_name = "%sysfunc(sasmsg
    (sashelp.dmine, rpt_varclus_label_clusternum, noquote,  &i_
    1))";
855 16419 +                %end;
856 16420 +          %end;
857 16421 +          keep Cluster
858 16422 +          %do i = 2 %to &nclus2;
859 16423 +                %let _name = %sysfunc(varname(&dsid,
     &i));
```

```
860  16424 +                &_name
861  16425 +          %end;
862  16426 +          ;
863  16427 +    %let dsid= %sysfunc(close(&dsid));
864  16428 +     run;
865  16429 +     quit;
866  16430 +
867  16431 +    %if &RSquare eq Y %then %do;
868  16432 +
869  16433 +       data corr_tmp;
870  16434 +        set &outdata;
871  16435 +     run;
872  16436 +
873  16437 +     data &outdata;
874  16438 +          set &outdata;
875  16439 +          %let dsid = %sysfunc(open(work.corr_tmp));
876  16440 +          %let nvar = %sysfunc(attrn(&dsid, NVAR));
877  16441 +          %do i = 2 %to &nvar;
878  16442 +            %let _name = %sysfunc(varname(&dsid, &i)
     );
879  16443 +            %let _name_md = &_name.**2;
880  16444 +                &_name = &_name_md;
881  16445 +          %end;
882  16446 +      %let dsid= %sysfunc(close(&dsid));
883  16447 +      run;
884  16448 +    %end;
885  16449 +
886  16450 +    %if &makeplotds eq Y %then %do;
887  16451 +     proc transpose data = &outdata
888  16452 +          out=&plotds(drop=_LABEL_ rename=(_NAME_ =
     Y Cluster=X Col1= Correlation));
889  16453 +          by cluster;
890  16454 +     run;
891  16455 +     data &plotds;
892  16456 +          set  &plotds;
893  16457 +          label x="%sysfunc(sasmsg(sashelp.dmine, rp
```

```
    t_varclus_label_cluster, noquote))";
894 16458 +          label Y="%sysfunc(sasmsg(sashelp.dmine, rp
    t_varclus_label_cluster, noquote))";
895 16459 +          label  Correlation="%sysfunc(sasmsg(sashel
    p.dmine, rpt_correlation_vlabel, noquote))";
896 16460 +     run;
897 16461 +     %end;
898 16462 +     proc datasets lib = work nolist;
899 16463 +           delete corr_tmp;
900 16464 +     run;
901 16465 +     quit;
902 16466 +%mend MakeInterClusCorrData;
903 16467 +
904 16468 +%macro MakeOwnRSquare(indata=, outdata=, ncluster=,
    globalclusid=);
905 16469 +    data _tmpds(drop= _NCL_);
906 16470 +       set &indata;
907 16471 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
    E_) in ('GROUP','RSQUARED')) then delete;
908 16472 +       %if &globalclusid ne %then %do;
909 16473 +           _NAME_ = "GC&globalclusid.";
910 16474 +       rename _NAME_ = Cluster;
911 16475 +       %end;
912 16476 +       %else %do;
913 16477 +        _NAME_ = "CLUS";
914 16478 +        rename _NAME_ = Cluster;
915 16479 +       %end;
916 16480 +       label _NAME_ = "%sysfunc(sasmsg(sashelp.dmine
    , rpt_varclus_label_clustername,  noquote))";
917 16481 +    run;
918 16482 +    proc transpose data =_tmpds out =&outdata;
919 16483 +    run;
920 16484 +
921 16485 +    data &outdata(drop=COL1);
922 16486 +        %if &globalclusid ne %then %do;
923 16487 +        length GCluster $16;
```

```
924  16488 +        %end;
925  16489 +        length Cluster $32;
926  16490 +        length _NAME_ $32;
927  16491 +         set &outdata;
928  16492 +         _NAME_ = upcase(_NAME_);
929  16493 +         rename _NAME_=Variable;
930  16494 +         *label _NAME_="Variable";
931  16495 +         label _NAME_ = "%sysfunc(sasmsg(sashelp.dmi
     ne, rpt_varclus_label_variable, noquote))" ;
932  16496 +         label Cluster = "%sysfunc(sasmsg(sashelp.dm
     ine, rpt_varclus_label_cluster, noquote))" ;
933  16497 +         label GCluster = "%sysfunc(sasmsg(sashelp.d
     mine, rpt_varclus_label_gcluster, noquote))" ;
934  16498 +
935  16499 +         %if &globalclusid ne %then %do;
936  16500 +          GCluster = "GC&globalclusid";
937  16501 +          Cluster = "GC&globalclusid._CLUS"||strip(C
     OL1);
938  16502 +         %end;
939  16503 +         %else %do;
940  16504 +         Cluster = "CLUS"||strip(COL1);
941  16505 +        %end;
942  16506 +         rename COL2 = RSqWithOwnClusComp;
943  16507 +         *label COL2 = "R-Square With Own Cluster Co
     mponent";
944  16508 +         label COL2 = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_ownrsq, noquote))" ;
945  16509 +
946  16510 +   run;
947  16511 +   proc sort data =&outdata;
948  16512 +        by Cluster RSqWithOwnClusComp;
949  16513 +   run;
950  16514 +   proc datasets lib = work nolist;
951  16515 +           delete _tmpds;
952  16516 +   run;
953  16517 +   quit;
```

```
954 16518 +%mend MakeOwnRSquare;
955 16519 +
956 16520 +%macro MakeClusStructCorrData(indata=, outdata=, glo
    balclusid=, ncluster=, Rsquare=N);
957 16521 +    data &outdata(drop= _NCL_  _TYPE_);
958 16522 +       %if &globalclusid ne %then %do;
959 16523 +        length GCluster $16;
960 16524 +        %end;
961 16525 +       set &indata;
962 16526 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
    E_) eq 'STRUCTUR') then delete;
963 16527 +       %if &globalclusid ne %then %do;
964 16528 +         GCluster = "GC&globalclusid";
965 16529 +         _NAME_ = "GC&globalclusid._"||upcase(_NAME_
    );
966 16530 +         rename _NAME_ = Cluster;
967 16531 +         label _NAME_ = "%sysfunc(sasmsg(sashelp.dmi
    ne, rpt_varclus_label_cluster, noquote))" ;
968 16532 +         label GCluster = "%sysfunc(sasmsg(sashelp.d
    mine, rpt_varclus_label_gcluster, noquote))" ;
969 16533 +
970 16534 +       %end;
971 16535 +       %else %do;
972 16536 +         _NAME_ = upcase(_NAME_);
973 16537 +         rename _NAME_ = Cluster;
974 16538 +         label _NAME_ = "%sysfunc(sasmsg(sashelp.dmi
    ne, rpt_varclus_label_cluster, noquote))" ;
975 16539 +         label GCluster = "%sysfunc(sasmsg(sashelp.d
    mine, rpt_varclus_label_gcluster, noquote))" ;
976 16540 +
977 16541 +       %end;
978 16542 +     run;
979 16543 +    %if &RSquare eq Y %then %do;
980 16544 +     data corr_tmp;
981 16545 +          set &outdata;
982 16546 +     run;
```

```
 983  16547 +     %let istart = 2;
 984  16548 +     %if &globalclusid ne %then %let istart = 3;
 985  16549 +     data &outdata;
 986  16550 +          set &outdata;
 987  16551 +          %let dsid = %sysfunc(open(work.corr_tmp));
 988  16552 +          %let nvar = %sysfunc(attrn(&dsid, NVAR));
 989  16553 +          %do i =&istart %to &nvar;
 990  16554 +            %let _name = %sysfunc(varname(&dsid, &i)
      );
 991  16555 +            %let _name_md = &_name.**2;
 992  16556 +                &_name = &_name_md;
 993  16557 +          %end;
 994  16558 +      %let dsid= %sysfunc(close(&dsid));
 995  16559 +      run;
 996  16560 +      proc datasets lib = work nolist;
 997  16561 +           delete corr_tmp;
 998  16562 +      run;
 999  16563 +    %end;
1000  16564 +     quit;
1001  16565 +%mend MakeClusStructCorrData;
1002  16566 +
1003  16567 +/*
1004  16568 +%MakeClusStructCorrData(indata=playpen._outstat, out
      data=_structrsq , ncluster=7, Rsquare=Y);
1005  16569 +*/
1006  16570 +
1007  16571 +%macro FindNextClosestClusByVar(indata=, outdata=, g
      lobalclusid=, ncluster=);
1008  16572 +
1009  16573 +     /* The indata should be the outdata
1010  16574 +        from %MakeClusStructCorrData(indata=, outdat
      a=, ); */
1011  16575 +
1012  16576 +     proc sort data =&indata out=_tmpclusRsq;
1013  16577 +     by cluster;
1014  16578 +     run;
```

```
1015  16579 +
1016  16580 +     proc transpose data =_tmpclusRsq out=_tmpclusRs
      q;
1017  16581 +      by cluster;
1018  16582 +     run;
1019  16583 +
1020  16584 +     proc sort data=_tmpclusRsq;
1021  16585 +        by _NAME_ COL1;
1022  16586 +     run;
1023  16587 +
1024  16588 +     data _tmpclusRsq;
1025  16589 +         length _NAME_ $32;
1026  16590 +         set _tmpclusRsq; by _NAME_;
1027  16591 +          _NAME_ = upcase(_NAME_);
1028  16592 +          %if &ncluster ne 1 %then %do;
1029  16593 +             if last._NAME_ then delete;
1030  16594 +          %end;
1031  16595 +          %else %do;
1032  16596 +             COL1 = 0;
1033  16597 +          %end;
1034  16598 +     run;
1035  16599 +     /* need to sort again */
1036  16600 +     proc sort data=_tmpclusRsq;
1037  16601 +        by _NAME_ COL1;
1038  16602 +     run;
1039  16603 +
1040  16604 +     data &outdata;
1041  16605 +         set _tmpclusRsq; by _NAME_;
1042  16606 +         Cluster = upcase(Cluster);
1043  16607 +         if last._NAME_ then output;
1044  16608 +         *label  COL1 = 'R-Sqaure with Next Cluster
      Component';
1045  16609 +         label COL1 = "%sysfunc(sasmsg(sashelp.dmine
      , rpt_varclus_label_nextrsq, noquote))" ;
1046  16610 +         rename COL1 = RSqWithNextClusComp;
1047  16611 +         Cluster = upcase(Cluster);
```

```
1048 16612 +         rename Cluster = ClosestCluster;
1049 16613 +         *label  Cluster = "Next Closest Cluster";
1050 16614 +         label Cluster = "%sysfunc(sasmsg(sashelp.dm
     ine, rpt_varclus_label_nextclus, noquote))" ;
1051 16615 +         rename _NAME_ = Variable;
1052 16616 +         label  _NAME_ = "%sysfunc(sasmsg(sashelp.dm
     ine, rpt_varclus_label_variable, noquote))" ;
1053 16617 +    run;
1054 16618 +
1055 16619 +
1056 16620 +    %if &globalclusid ne %then %do;
1057 16621 +    data &outdata;
1058 16622 +       length GCluster $16;
1059 16623 +       set &outdata;
1060 16624 +       GCluster = "GC&globalclusid";
1061 16625 +       run;
1062 16626 +    %end;
1063 16627 +    proc datasets lib = work nolist;
1064 16628 +           delete _tmpclusRsq;
1065 16629 +    run;
1066 16630 +    quit;
1067 16631 +%mend FindNextClosestClusByVar;
1068 16632 +
1069 16633 +
1070 16634 +%macro FindNextClosestClusByCluster(indata=, outdata
     =, globalclusid=, ncluster=);
1071 16635 +     /* The indata should be the outdata from %MakeI
     nterClusCorrData(indata=, outdata=, ); */
1072 16636 +     proc sort data =&indata out=_tmpclusRsq;
1073 16637 +     by cluster;
1074 16638 +     run;
1075 16639 +     proc transpose data =_tmpclusRsq out=_tmpclusRs
     q;
1076 16640 +     by cluster;
1077 16641 +     run;
1078 16642 +     proc sort data=_tmpclusRsq;
```

```
1079 16643 +        by _NAME_ col1;
1080 16644 +     run;
1081 16645 +     data _tmpclusRsq;
1082 16646 +         length _NAME_ $32;
1083 16647 +         set _tmpclusRsq; by _NAME_;
1084 16648 +          _NAME_ = upcase(_NAME_);
1085 16649 +          %if &ncluster ne 1 %then %do;
1086 16650 +             if last._NAME_ then delete;
1087 16651 +          %end;
1088 16652 +          %else %do;
1089 16653 +             COL1 = 0;
1090 16654 +          %end;
1091 16655 +     run;
1092 16656 +     data &outdata;
1093 16657 +         set _tmpclusRsq; by _NAME_;
1094 16658 +         Cluster = upcase(Cluster);
1095 16659 +         if last._NAME_ then output;
1096 16660 +         *label  COL1 = 'R-Sqaure with Next Cluster
     Component';
1097 16661 +         label COL1 = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_nextrsq, noquote))" ;
1098 16662 +         rename COL1 = RSqWithNextClusComp;
1099 16663 +         Cluster = upcase(Cluster);
1100 16664 +         rename Cluster = ClosestCluster;
1101 16665 +         *label  Cluster = "Next Closest Cluster";
1102 16666 +         label Cluster = "%sysfunc(sasmsg(sashelp.dm
     ine, rpt_varclus_label_nextclus, noquote))" ;
1103 16667 +         rename _NAME_ = Variable;
1104 16668 +         *label  _NAME_ = "Variable";
1105 16669 +         label  _NAME_ = "%sysfunc(sasmsg(sashelp.dm
     ine, rpt_varclus_label_variable, noquote))" ;
1106 16670 +
1107 16671 +    run;
1108 16672 +    %if &globalclusid ne %then %do;
1109 16673 +    data &outdata;
1110 16674 +       length GCluster $16;
```

```
1111 16675 +       set &outdata;
1112 16676 +       GCluster = "GC&globalclusid";
1113 16677 +       run;
1114 16678 +    %end;
1115 16679 +
1116 16680 +    proc datasets lib = work nolist;
1117 16681 +           delete _tmpclusRsq;
1118 16682 +    run;
1119 16683 +
1120 16684 +    quit;
1121 16685 +%mend FindNextClosestClusByCluster;
1122 16686 +
1123 16687 +%macro MakeVarClusResultTable(indata1=, indata2=, in
     data3=, outdata=, globalclusid=, ncluster=, selectedcomp=cl
     ustercomp);
1124 16688 +/*----
1125 16689 +  indata1=_ownRsq, indata2=_nextVarRsq, indata3=_nex
     tClusRSq,
1126 16690 +-----------*/
1127 16691 +
1128 16692 +proc sort data =&indata1;
1129 16693 +    by Variable;
1130 16694 +run;
1131 16695 +proc sort data =&indata2;
1132 16696 +    by Variable;
1133 16697 +run;
1134 16698 +data &outdata;
1135 16699 +    merge &indata1 &indata2;
1136 16700 +    by Variable;
1137 16701 +    length Type $16;
1138 16702 +    Type = 'Variable';
1139 16703 +    *label Type ='Type';
1140 16704 +    label Type = "%sysfunc(sasmsg(sashelp.dmine, rpt
     _varclus_label_type, noquote))" ;
1141 16705 +run;
1142 16706 +
```

```
1143  16707 +
1144  16708 +data &indata3;
1145  16709 +    set  &indata3 ;
1146  16710 +    length RSqWithOwnClusComp 8.;
1147  16711 +    Cluster = Variable;
1148  16712 +    RSqWithOwnClusComp = 1;
1149  16713 +    *label RSqWithOwnClusComp = "R-Square With Own C
      luster Component";
1150  16714 +    label RSqWithOwnClusComp = "%sysfunc(sasmsg(sash
      elp.dmine, rpt_varclus_label_ownrsq, noquote))" ;
1151  16715 +    length Type $16;
1152  16716 +    Type = 'ClusterComp';
1153  16717 +    label Type = "%sysfunc(sasmsg(sashelp.dmine, rpt
      _varclus_label_type, noquote))" ;
1154  16718 +
1155  16719 +;
1156  16720 +run;
1157  16721 +
1158  16722 +proc sort data=&outdata;
1159  16723 +    by Cluster;
1160  16724 +run;
1161  16725 +proc sort data =&indata3;
1162  16726 +    by Cluster;
1163  16727 +run;
1164  16728 +
1165  16729 +data &outdata;
1166  16730 +     set &outdata &indata3;
1167  16731 +     by Cluster;
1168  16732 +run;
1169  16733 +
1170  16734 +
1171  16735 +/* Create the Selected variable with all YES */
1172  16736 +
1173  16737 +data &outdata;
1174  16738 +     set &outdata;
1175  16739 +     length RsqRatio 8.;
```

```
1176 16740 +     length Selected $8;
1177 16741 +     *label RSqRatio = "1-R**2 Ratio";
1178 16742 +     label RSqRatio =  "%sysfunc(sasmsg(sashelp.dmin
     e, rpt_varclus_label_oneminusrsq, noquote))" ;
1179 16743 +     *label Selected = "Variable Selected";
1180 16744 +     label Selected = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_varselected, noquote))" ;
1181 16745 +     RsqRatio = (1-RSqWithOwnClusComp)/(1-RSqWithNex
     tClusComp);
1182 16746 +     Selected ='YES';
1183 16747 +     rename _LABEL_ = Label;
1184 16748 +     label _LABEL_ =  "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_label, noquote))" ;
1185 16749 +run;
1186 16750 +
1187 16751 +
1188 16752 +/*----  Selected = Y/N will be assigened at the %sco
     re                  -------+
1189 16753 +        Just create the Selected variable with all Y
     ES at the step above
1190 16754 + +-----------------------------------------------
     ----------------------------+
1191 16755 +
1192 16756 +proc sort data=&outdata;
1193 16757 +    by Cluster RsqRatio;
1194 16758 +run;
1195 16759 +
1196 16760 +%if &selectedcomp eq CLUSTERCOMP %then %do;
1197 16761 +data &outdata;
1198 16762 +     set &outdata; by Cluster;
1199 16763 +     length Selected $8;
1200 16764 +     label Selected = "Variable Selected";
1201 16765 +     if  first.Cluster then Selected ='Yes';
1202 16766 +     else Selected = 'No';
1203 16767 +    run;
1204 16768 +%end;
```

```
1205 16769 +%else %do;
1206 16770 +data &outdata(drop = _var _varchange);
1207 16771 +     set &outdata; retain _var 0; by Cluster;
1208 16772 +     length Selected $8;
1209 16773 +     label Selected = "Variable Selected";
1210 16774 +     if first.Cluster then  _varchange = 0;
1211 16775 +     else _varchange =1;
1212 16776 +     if _var ne _varchange then  Selected  = 'Yes';
1213 16777 +     else Selected = 'No';
1214 16778 +     if last.cluster then  _var = 0;
1215 16779 +     else _var = _varchange;
1216 16780 +run;
1217 16781 +%end;
1218 16782 +
1219 16783 +-------------------------------------------------
     ----------*/
1220 16784 +
1221 16785 +quit;
1222 16786 +%mend MakeVarClusResultTable;
1223 16787 +
1224 16788 +%Macro MakePlotDataFromCorrTable(indata=, outdata=,
     globalclusid=);
1225 16789 +     proc sort data =&indata;
1226 16790 +        by cluster;
1227 16791 +     run;
1228 16792 +     proc transpose data =&indata
1229 16793 +          out=&outdata(drop=_LABEL_ rename=(_NAME_ =
      Y Cluster=X Col1= Correlation));
1230 16794 +          by cluster;
1231 16795 +     run;
1232 16796 +     data &outdata;
1233 16797 +          set  &outdata;
1234 16798 +          label x= "%sysfunc(sasmsg(sashelp.dmine, r
     pt_varclus_label_cluster, noquote))" ;
1235 16799 +          label Y= "%sysfunc(sasmsg(sashelp.dmine, r
     pt_varclus_label_cluster, noquote))" ;
```

```
1236 16800 +          label Correlation = "%sysfunc(sasmsg(sashe
     lp.dmine, rpt_correlation_vlabel, noquote))" ;
1237 16801 +     run;
1238 16802 +     %if &globalclusid ne %then %do;
1239 16803 +     data &outdata;
1240 16804 +          Length GCluster $16;
1241 16805 +          label GCluster = "%sysfunc(sasmsg(sashelp.
     dmine, rpt_varclus_label_gcluster, noquote))" ;
1242 16806 +          set  &outdata;
1243 16807 +          GCluster ="GC&globalclusid.";
1244 16808 +          run;
1245 16809 +     %end;
1246 16810 +
1247 16811 +%Mend MakePlotDataFromCorrTable;
1248 16812 +
1249 16813 +
1250 16814 +%macro MakeCorrelation(indata=,
1251 16815 +                       outstat=_tmpoutstat,
1252 16816 +                       corrmatrix=N,
1253 16817 +                       outcorr=_tmpoutcorr,
1254 16818 +                       includeclassvar=N,
1255 16819 +                       target=,
1256 16820 +                       freq=,
1257 16821 +                       weight=);
1258 16822 +   %if &target eq  %then %do;
1259 16823 +
1260 16824 +       proc varclus data=&indata outstat=&outstat hi
      maxclusters=1 noprint;
1261 16825 +            var %EM_INTERVAL_INPUT %EM_INTERVAL_REJE
     CTED
1262 16826 +       %if &includeclassvar eq Y %then %do;
1263 16827 +          %let dsid = %sysfunc(open(&EM_USER_OUTDUMM
     Y));
1264 16828 +          %let nvar = %sysfunc(attrn(&dsid, NVAR));
1265 16829 +          %do i = 2 %to &nvar;
1266 16830 +          %let varname = %sysfunc(varname(&dsid, &i)
```

```
     );
1267 16831 +          &varname
1268 16832 +          %end;
1269 16833 +       %end;
1270 16834 +       ;
1271 16835 +      %if &freq ne  %then %do;
1272 16836 +          freq &freq;
1273 16837 +      %end;
1274 16838 +      %if &weight ne  %then %do;
1275 16839 +          weight &weight;
1276 16840 +      %end;
1277 16841 +
1278 16842 +      run;
1279 16843 +      %if &corrmatrix eq Y %then %do;
1280 16844 +      data  &outcorr (drop = _NCL_ _TYPE_);
1281 16845 +           set  &outstat ;
1282 16846 +           if _TYPE_ ='CORR' then output;
1283 16847 +      run;
1284 16848 +      %end;
1285 16849 +    %end;
1286 16850 +    %else %do;
1287 16851 +       proc corr data=&indata outp=&outstat noprint;
1288 16852 +            var
1289 16853 +       %let dsid = %sysfunc(open(&indata));
1290 16854 +       %let nvar = %sysfunc(attrn(&dsid, NVAR));
1291 16855 +          %do i = 1 %to &nvar;
1292 16856 +              %let _name = %sysfunc(varname(&dsid, &
     i));
1293 16857 +              %if &_name ne &target %then;
1294 16858 +              &_name
1295 16859 +           %end;
1296 16860 +        %let dsid= %sysfunc(close(&dsid));
1297 16861 +        ;
1298 16862 +        with &target;
1299 16863 +        run;
1300 16864 +    %end;
```

```
1301  16865 +    quit;
1302  16866 +%mend MakeCorrelation;
1303  16867 +
1304  16868 +
1305  16869 +%macro MakeCorrelationDistance(indata=,
1306  16870 +                                outdata=,
1307  16871 +                                rsquare = N
1308  16872 +                                );
1309  16873 +    data corr_tmp;
1310  16874 +        set &indata;
1311  16875 +        if _N_ = 1 then do;
1312  16876 +           output;
1313  16877 +           stop;
1314  16878 +         end;
1315  16879 +    run;
1316  16880 +    %if &outdata ne  %then %let  _outdata = &outdata
     ;
1317  16881 +    %else %let _outdata = &indata;
1318  16882 +
1319  16883 +    data &_outdata;
1320  16884 +         set &indata;
1321  16885 +
1322  16886 +         %let dsid = %sysfunc(open(work.corr_tmp));
1323  16887 +         %let nvar = %sysfunc(attrn(&dsid, NVAR));
1324  16888 +          %do i = 2 %to &nvar;
1325  16889 +              %let _name = %sysfunc(varname(&dsid, &
     i));
1326  16890 +              %if &rsquare eq Y %then %let _name_md
     = &_name.**2;
1327  16891 +              %else  %let _name_md = &_name;
1328  16892 +              &_name = 1- &_name_md;
1329  16893 +          %end;
1330  16894 +      %let dsid= %sysfunc(close(&dsid));
1331  16895 +      run;
1332  16896 +      proc datasets lib = work nolist;
1333  16897 +           delete corr_tmp;
```

```
1334  16898 +      run;
1335  16899 +      quit;
1336  16900 +  %mend MakeCorrelationDistance;
1337  16901 +
1338  16902 +
1339  16903 +%macro UpdateOutStatCorrToDistance(indata=, /* indat
      a should be a outstat from proc varclus */
1340  16904 +                                     rsquare = N
1341  16905 +                                     );
1342  16906 +    data corr_tmp;
1343  16907 +        set &indata;
1344  16908 +    run;
1345  16909 +    proc sql noprint;
1346  16910 +           update &indata
1347  16911 +           set
1348  16912 +     %let dsid = %sysfunc(open(work.corr_tmp));
1349  16913 +     %let nvar = %sysfunc(attrn(&dsid, NVAR));
1350  16914 +          %do i = 4 %to &nvar;
1351  16915 +          %let _name = %sysfunc(varname(&dsid, &i));
1352  16916 +             %if &rsquare eq Y %then %let _name_md =
       &_name.**2;
1353  16917 +            %else  %let _name_md = &_name;
1354  16918 +            %if &i < &nvar %then %do;
1355  16919 +               &_name = 1- &_name_md ,
1356  16920 +            %end;
1357  16921 +            %else %do;
1358  16922 +               &_name = &_name_md where _TYPE_ eq 'C
      ORR' ;
1359  16923 +            %end;
1360  16924 +          %end;
1361  16925 +      %let dsid= %sysfunc(close(&dsid));
1362  16926 +
1363  16927 +      select * from &indata;
1364  16928 +      run;
1365  16929 +     data &indata( drop = _NCL_);
1366  16930 +           set &indata;
```

```
1367  16931 +           if _TYPE_ not in ('CORR', 'STD', 'N', 'ME
      AN') then delete;
1368  16932 +           if _TYPE_ ='CORR' then _TYPE_ ='DISTANCE'
      ;
1369  16933 +     run;
1370  16934 +     data &indata(DROP = _NCL_);
1371  16935 +           set &indata;
1372  16936 +           if _TYPE_ = 'CORR' then _TYPE_ ='DISTANCE
      ';
1373  16937 +           if _TYPE_ not in ('DISTANCE', 'N', 'STD',
       'MEAN') then delete;
1374  16938 +           rename _NAME_ = _VAR_;
1375  16939 +     run;
1376  16940 +     proc datasets lib = work nolist;
1377  16941 +           delete corr_tmp;
1378  16942 +     run;
1379  16943 +     quit;
1380  16944 + %mend UpdateOutStatCorrToDistance;
1381  16945 +
1382  16946 +
1383  16947 +%macro HierClusWithCorr(indata= ,
1384  16948 +                         ncluster=,
1385  16949 +                         method = Ward,
1386  16950 +                         outtree = _outtree,
1387  16951 +                         idvar =_VAR_,
1388  16952 +                         outdata=,
1389  16953 +                         rescore = N,
1390  16954 +                         newncluster=
1391  16955 +                         );
1392  16956 +      %global &newncluster;
1393  16957 +      %if &rescore ne Y %then %do;
1394  16958 +      proc cluster data=&indata(type=Distance where=
      (upcase(strip(_TYPE_)) = "DISTANCE"))
1395  16959 +                   method=&method outtree=&outtree n
      oprint;
1396  16960 +           id &idvar;
```

```
1397 16961 +      run;
1398 16962 +      %end;
1399 16963 +      proc tree data=&outtree nclusters = &ncluster
     out=&outdata noprint;
1400 16964 +      run;
1401 16965 +      /* ----- Check some variables like CL1, CL5...
     , remove them ----*/
1402 16966 +      proc contents data =&indata out=_outcontent(ke
     ep=NAME) noprint;
1403 16967 +      run;
1404 16968 +      data _outcontent;
1405 16969 +          set _outcontent;
1406 16970 +          if NAME in ('_TYPE_' , '_VAR_') then delet
     e;
1407 16971 +          index = 1;
1408 16972 +          rename NAME = _NAME_;
1409 16973 +      run;
1410 16974 +      proc sort data=_outcontent;
1411 16975 +          by _NAME_;
1412 16976 +      run;
1413 16977 +      proc sort data =&outdata;
1414 16978 +           by _NAME_;
1415 16979 +      run;
1416 16980 +      data &outdata(drop=index);
1417 16981 +          merge &outdata _outcontent;
1418 16982 +          by _NAME_;
1419 16983 +          if index = . then delete;
1420 16984 +      run;
1421 16985 +      /*-----------------------------------------
     ---------------*/
1422 16986 +      data &outdata;
1423 16987 +           length CLUSNAME $16;
1424 16988 +           set &outdata;
1425 16989 +           if CLUSTER > &ncluster then delete;
1426 16990 +           CLUSNAME='GC'||strip(CLUSTER);
1427 16991 +           *label CLUSNAME = "Cluster Name";
```

```
1428 16992 +           label CLUSNAME  = "%sysfunc(sasmsg(sashel
     p.dmine, rpt_varclus_label_clustername, noquote))" ;
1429 16993 +           rename _NAME_ = VARIABLE ;
1430 16994 +           *label _NAME_ = "Variable";
1431 16995 +           *label CLUSTER = "Cluster";
1432 16996 +           label _NAME_ ="%sysfunc(sasmsg(sashelp.dm
     ine, rpt_varclus_label_variable, noquote))" ;
1433 16997 +           label CLUSTER ="%sysfunc(sasmsg(sashelp.d
     mine, rpt_varclus_label_cluster, noquote))" ;
1434 16998 +      run;
1435 16999 +      proc sort data=&outdata out=&outdata;
1436 17000 +           by CLUSTER;
1437 17001 +      run;
1438 17002 +      proc means data =&outdata noprint;
1439 17003 +           output out=_meanout;
1440 17004 +      run;
1441 17005 +      data _null_;
1442 17006 +            set _meanout;
1443 17007 +            if strip(_STAT_) eq 'MAX' then do;
1444 17008 +            call symput("&newncluster", CLUSTER);
1445 17009 +            stop;
1446 17010 +            end;
1447 17011 +      run;
1448 17012 +
1449 17013 +      proc datasets lib = work nolist;
1450 17014 +           delete _outcontent _meanout;
1451 17015 +      run;
1452 17016 +      quit;
1453 17017 +%mend  HierClusWithCorr;
1454 17018 +
1455 17019 +%macro CreateScoreCode2(indata=, ncluscomp=, globalc
     lusid=, fileref=);
1456 17020 +
1457 17021 +     data _tmpindata;
1458 17022 +          set &indata;
1459 17023 +          if (_TYPE_ in ('SCORE' 'MEAN' 'STD') and _
```

```
     NCL_ = &ncluscomp ) or (_TYPE_ in ('MEAN' 'STD'));
1460 17024 +          if _TYPE_ = 'MEAN' then _NAME_='MEAN';
1461 17025 +          if _TYPE_ = 'STD' then _NAME_='STD';
1462 17026 +          if _TYPE_ = 'SCORE' then _NAME_=upcase("GC
     &globalclusid._"||_NAME_);
1463 17027 +          DROP _TYPE_ _NCL_;
1464 17028 +     run;
1465 17029 +
1466 17030 +     filename _file_  "&fileRef";
1467 17031 +
1468 17032 +     data _null_;
1469 17033 +        FILE _file_ MOD;
1470 17034 +        put ' ';
1471 17035 +        put "/*-----------------------------------
     ------------*/";
1472 17036 +        put '/* ' "%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_score_title_gclus, noquote, &globalclusid))" '*/'
     ;
1473 17037 +        put "/*-----------------------------------
     ------------*/";
1474 17038 +        put ' ';
1475 17039 +        %let dsid = %sysfunc(open(work._tmpindata));
1476 17040 +
1477 17041 +        %let nvar = %sysfunc(attrn(&dsid, NVAR));
1478 17042 +        %let vn_name =%sysfunc(varnum(&dsid, _NAME_)
     );
1479 17043 +        %let k = 1;
1480 17044 +        %do %while(^%sysfunc(fetch(&dsid)));
1481 17045 +              %let _name = %sysfunc(getvarc(&dsid,
     &vn_name));
1482 17046 +              %if &k > 2 %then %do;
1483 17047 +               %let cn = %eval(&k-2);
1484 17048 +                put "&_name = 0 ; /*---" "%sysfunc(
     sasmsg(sashelp.dmine, rpt_varclus_score_gcluscompnum, noquo
     te, &globalclusid, &cn))" "------ */";
1485 17049 +               %end;
```

```
1486  17050 +                %let k = %eval(&k+1);
1487  17051 +
1488  17052 +        %end;
1489  17053 +        %let rc = %sysfunc(rewind(&dsid));
1490  17054 +
1491  17055 +        %do j= 2 %to &nvar;
1492  17056 +            %let _varname = %sysfunc(varname(&dsid,
      &j));
1493  17057 +            %do %while(^%sysfunc(fetch(&dsid)));
1494  17058 +                %let _name = %sysfunc(getvarc(&dsid,
       &vn_name));
1495  17059 +                %if &_name = MEAN %then
1496  17060 +                %let _mean = %sysfunc(getvarn(&dsid,
       &j));
1497  17061 +                %else %if &_name = STD %then
1498  17062 +                %let _std = %sysfunc(getvarn(&dsid,
      &j));
1499  17063 +                %else %do;
1500  17064 +                      %let coeff =  %sysfunc(getvarn
      (&dsid, &j));
1501  17065 +                      %let abscoeff = %sysfunc(abs(&
      coeff));
1502  17066 +                          %if &abscoeff >  0 %then %
      do;
1503  17067 +                       put "&_name = &_name+&coeff *
       (&_varname - &_mean)/&_std;";
1504  17068 +                           %end;
1505  17069 +                %end;
1506  17070 +             %end;
1507  17071 +             %let rc = %sysfunc(rewind(&dsid));
1508  17072 +         %end;
1509  17073 +
1510  17074 +        %let dsid= %sysfunc(close(&dsid));
1511  17075 +
1512  17076 +     run;
1513  17077 +
```

```
1514  17078 +     filename _file_;
1515  17079 +     proc datasets lib = work nolist;
1516  17080 +           delete _tmpindata;
1517  17081 +     run;
1518  17082 +     quit;
1519  17083 +%mend CreateScoreCode2;
1520  17084 +
1521  17085 +
1522  17086 +%macro MakeDeltaCode2(groupds=,deltacodefile=);
1523  17087 +
1524  17088 +     /*--- Build Code to Modify Metadata ---*/
1525  17089 +     filename X "&deltacodefile";
1526  17090 +     data _null_;
1527  17091 +        FILE X;
1528  17092 +        set &groupds end=eof;
1529  17093 +        if _N_=1 then do;
1530  17094 +           %if &EM_PROPERTY_INCLUDECLASSVAR eq Y %th
      en %do;
1531  17095 +            put "if upcase(strip(ROLE)) ='INPUT' and
       upcase(strip(LEVEL)) ^='INTERVAL' then ROLE ='REJECTED' ;"
      ;
1532  17096 +           %end;
1533  17097 +           put "if upcase(strip(ROLE))='INPUT' and u
      pcase(strip(LEVEL))='INTERVAL' then do;";
1534  17098 +           put "if upcase(strip(NAME)) in (";
1535  17099 +        end;
1536  17100 +        if Strip(upcase(Selected)) eq 'YES' then do;
1537  17101 +           string = '"'!!trim(left(VARIABLE))!!'"';
1538  17102 +           put string;
1539  17103 +        end;
1540  17104 +        if eof then do;
1541  17105 +           put ') then ROLE="INPUT";';
1542  17106 +           put 'else ROLE="REJECTED";';
1543  17107 +           put 'end;';
1544  17108 +
1545  17109 +           %if %upcase(&EM_PROPERTY_HIDEVARIABLE) eq
```

```
      Y %then %do;
1546 17110 +             put 'if upcase(strip(ROLE)) = "REJECTED
     " then delete ;';
1547 17111 +           %end;
1548 17112 +        end;
1549 17113 +     run;
1550 17114 +     quit;
1551 17115 +
1552 17116 +     filename X;
1553 17117 +     quit;
1554 17118 +%mend MakeDeltaCode2;
1555 17119 +
1556 17120 +%macro getInitialGClusterNumber(indata=, ninput=, nd
     ummy=0, div=100, ngc=);
1557 17121 +  %global &ngc;
1558 17122 +  data _null_;
1559 17123 +  %if &indata ne %then %do;
1560 17124 +      %let dsid = %sysfunc(open(&indata));
1561 17125 +           %let nvar = %sysfunc(attrn(&dsid, NVAR));
1562 17126 +      %let dsid = %sysfunc(close(&dsid));
1563 17127 +  %end;
1564 17128 +  %else %do;
1565 17129 +       %let nvar = %eval(&ninput+&ndummy); ;
1566 17130 +  %end;
1567 17131 +  %let numgc = %eval(&nvar/&div+2);
1568 17132 +  %let &ngc = &numgc;
1569 17133 +   run;
1570 17134 +  quit;
1571 17135 +%mend getInitialGClusterNumber;
1572 17136 +
1573 17137 +
1574 17138 +%macro MakeGobalConstellData(indata=, outlink=, outn
     ode=);
1575 17139 +data &outlink(drop = Selected);
1576 17140 +     set &indata;
1577 17141 +     LINKID = _N_;
```

```
1578 17142 +     label LINKID = "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_linkid, noquote))" ;
1579 17143 +run;
1580 17144 +data &outnode(keep=NODEID TYPE LABEL);
1581 17145 +    set &indata;
1582 17146 +    length TYPE $16;
1583 17147 +    rename VARIABLE = NODEID;
1584 17148 +    *label  CLUSNAME="Node ID";
1585 17149 +    label CLUSNAME= "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_nodeidvar, noquote))" ;
1586 17150 +    TYPE = "VARIABLE";
1587 17151 +    *label TYPE = "Node Type";
1588 17152 +    label TYPE =  "%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_label_nodetype, noquote))" ;
1589 17153 +    run;
1590 17154 +data _tmp(keep=NODEID TYPE LABEL);
1591 17155 +    set &indata;
1592 17156 +    length TYPE $16;
1593 17157 +    rename CLUSNAME = NODEID;
1594 17158 +    label  CLUSNAME= "%sysfunc(sasmsg(sashelp.dmine,
      rpt_varclus_label_nodeidvar, noquote))" ;
1595 17159 +    TYPE = "GCLUSTER";
1596 17160 +    label TYPE = "%sysfunc(sasmsg(sashelp.dmine, rpt
     _varclus_label_nodetype, noquote))" ;
1597 17161 +    run;
1598 17162 +proc sort data=_tmp;
1599 17163 +    by NODEID;
1600 17164 +run;
1601 17165 +data _tmp;
1602 17166 +    set _tmp; by NODEID;
1603 17167 +    if first.NODEID then output;
1604 17168 +run;
1605 17169 +proc sort data=&outnode;
1606 17170 +    by NODEID;
1607 17171 +run;
1608 17172 +data  &outnode;
```

```
1609 17173 +    set _tmp &outnode;
1610 17174 +run;
1611 17175 +proc datasets lib = work nolist;
1612 17176 +     delete _tmp;
1613 17177 +run;
1614 17178 +quit;
1615 17179 +%mend MakeGobalConstellData;
1616 17180 +
1617 17181 +/* Make contellation plot data among GCLUSTERS */
1618 17182 +
1619 17183 +%Macro MakeGClusterConstData(indata=, inoutrsq=, out
     node=, outlink=);
1620 17184 +
1621 17185 +data &outlink(keep = _NAME_ _PARENT_ _LABEL_ LINKID)
     ;
1622 17186 +     set &indata;
1623 17187 +     LINKID = _N_;
1624 17188 +     if upcase(substr(strip(_NAME_),1, 2))="CL" then
      do;
1625 17189 +        _NAME_ = "ROOT"||upcase(substr(strip(_NAME_)
     ,5));
1626 17190 +     end;
1627 17191 +     if _PARENT_ ne " " and upcase(substr(strip(_PAR
     ENT_),1, 2))="CL" then do;
1628 17192 +        _PARENT_ = "ROOT"||upcase(substr(strip(_PARE
     NT_),5));
1629 17193 +     end;
1630 17194 +     if upcase(substr(strip(_LABEL_),1, 2))="CL" the
     n do;
1631 17195 +        _LABEL_ = "ROOT"||upcase(substr(strip(_LABEL
     _),5));
1632 17196 +    end;
1633 17197 +run;
1634 17198 +
1635 17199 +data _tmp_outrsquare;
1636 17200 +        set &inoutrsq;
```

```
1637 17201 +        if upcase(strip(TYPE)) = 'CLUSTERCOMP' then
     delete;
1638 17202 +run;
1639 17203 +
1640 17204 +proc freq data =_tmp_outrsquare noprint;
1641 17205 +         tables GCluster/out=_tmp_GCLUSFREQ(rename=(
     GCLUSTER=_NAME_));
1642 17206 +run;
1643 17207 +
1644 17208 +data &outnode(keep=_NAME_ TYPE LABEL);
1645 17209 +    set &outlink;
1646 17210 +    length TYPE $16;
1647 17211 +    length LABEL $100;
1648 17212 +    /*label CLUSNAME= "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_nodeidvar, noquote))" ;*/
1649 17213 +
1650 17214 +    if upcase(substr(strip(_NAME_),1, 2))='GC' then
     do;
1651 17215 +    TYPE = "GCLUSTER";
1652 17216 +    LABEL = "%sysfunc(sasmsg(sashelp.dmine, rpt_varc
     lus_label_gcluster, noquote)):"||_NAME_;
1653 17217 +    end;
1654 17218 +    else do;
1655 17219 +    TYPE= "ROOT";
1656 17220 +    LABEL= _NAME_;
1657 17221 +    end;
1658 17222 +    label TYPE  =  "%sysfunc(sasmsg(sashelp.dmine, r
     pt_varclus_label_nodetype, noquote))" ;
1659 17223 +    label LABEL =  "%sysfunc(sasmsg(sashelp.dmine, r
     pt_varclus_label_label, noquote))" ;
1660 17224 +    label _NAME_ =  "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_nodeidvar, noquote))" ;
1661 17225 +    run;
1662 17226 +
1663 17227 +proc sort data=&outnode;
1664 17228 +     by _NAME_;
```

```
1665 17229 +proc sort data=_tmp_GCLUSFREQ;
1666 17230 +     by _NAME_;
1667 17231 +run;
1668 17232 +
1669 17233 +data  &outnode;
1670 17234 +     merge &outnode _tmp_GCLUSFREQ; by _NAME_;
1671 17235 +     if COUNT=. then COUNT=1;
1672 17236 +run;
1673 17237 +
1674 17238 +proc datasets lib = work nolist;
1675 17239 +     delete _tmp_outrsquare _tmp_GCLUSFREQ;
1676 17240 +run;
1677 17241 +
1678 17242 +quit;
1679 17243 +%Mend MakeGClusterConstData;
1680 17244 +
1681 17245 +
1682 17246 +%macro CreateGClusterScoreCode(indata=,  globalclusi
     d=, fileref=);
1683 17247 +
1684 17248 +    data _gscoretmpds;
1685 17249 +          set &indata;
1686 17250 +          if (_TYPE_ in ('SCORE' 'MEAN' 'STD') and _
     NCL_ = 1 ) or (_TYPE_ in ('MEAN' 'STD'));
1687 17251 +          if _TYPE_ = 'MEAN' then _NAME_='MEAN';
1688 17252 +          if _TYPE_ = 'STD' then _NAME_='STD';
1689 17253 +          if _TYPE_ = 'SCORE' then _NAME_ = "GC"||st
     rip(&globalclusid);
1690 17254 +          DROP _TYPE_ _NCL_;
1691 17255 +     run;
1692 17256 +
1693 17257 +    /*  %let gscorefile =  %bquote(&EM_NODEDIR)&EM_D
     SEP.gclusterscore.sas;
1694 17258 +        GCluster Component &globalclusid ------ */
1695 17259 +
1696 17260 +    filename _file_  "&fileref";
```

```
1697 17261 +
1698 17262 +    data _null_;
1699 17263 +        %if &globalclusid eq 1 %then %do;
1700 17264 +          FILE _file_;
1701 17265 +        %end;
1702 17266 +        %else %do;
1703 17267 +          FILE _file_ MOD;
1704 17268 +        %end;
1705 17269 +
1706 17270 +        %let dsid = %sysfunc(open(work._gscoretmpds)
     );
1707 17271 +        %let nvar = %sysfunc(attrn(&dsid, NVAR));
1708 17272 +        %let vn_name =%sysfunc(varnum(&dsid, _NAME_)
     );
1709 17273 +
1710 17274 +        %let k = 1;
1711 17275 +        %do %while(^%sysfunc(fetch(&dsid)));
1712 17276 +                %let _name = %sysfunc(getvarc(&dsid,
      &vn_name));
1713 17277 +                %if &k > 2 %then %do;
1714 17278 +                put "&_name = 0 ; ";
1715 17279 +                %end;
1716 17280 +                %let k = %eval(&k+1);
1717 17281 +        %end;
1718 17282 +
1719 17283 +        %let rc = %sysfunc(rewind(&dsid));
1720 17284 +        %do i= 2 %to &nvar;
1721 17285 +            %let _varname =  %sysfunc(varname(&dsid,
      &i));
1722 17286 +            %do %while(^%sysfunc(fetch(&dsid)));
1723 17287 +                %let _name = %sysfunc(getvarc(&dsid,
      &vn_name));
1724 17288 +                %if &_name = MEAN %then
1725 17289 +                %let _mean = %sysfunc(getvarn(&dsid,
      &i));
1726 17290 +                %else %if &_name = STD %then
```

```
1727 17291 +                %let _std = %sysfunc(getvarn(&dsid,
     &i));
1728 17292 +                %else %do;
1729 17293 +                      %let coeff =  %sysfunc(getvarn
     (&dsid, &i));
1730 17294 +                      %let abscoeff = %sysfunc(abs(&
     coeff));
1731 17295 +                          %if &abscoeff >  0 %then %
     do;
1732 17296 +                       put "&_name = &_name+&coeff *
      (&_varname - &_mean)/&_std;";
1733 17297 +                           %end;
1734 17298 +                 %end;
1735 17299 +             %end;
1736 17300 +             %let rc = %sysfunc(rewind(&dsid));
1737 17301 +
1738 17302 +         %end;
1739 17303 +
1740 17304 +        %let dsid= %sysfunc(close(&dsid));
1741 17305 +       run;
1742 17306 +
1743 17307 +
1744 17308 +       proc datasets lib=work nolist;
1745 17309 +            delete _gscoretmpds;
1746 17310 +       run;
1747 17311 +      quit;
1748 17312 +
1749 17313 +%mend CreateGClusterScoreCode;
1750 17314 +
1751 17315 +
1752 17316 +%macro MakeGClusterCorrelation(Indata=, ngcluster=,
     gscorecode=, outrsquare=);
1753 17317 +
1754 17318 +   %EM_REGISTER(KEY=GSCORE, TYPE=DATA);
1755 17319 +   %EM_GETNAME(KEY=GSCORE, TYPE=DATA);
1756 17320 +   %EM_REGISTER(KEY=GSCORESTAT, TYPE=DATA);
```

```
1757 17321 +   %EM_GETNAME(KEY=GSCORESTAT, TYPE=DATA);
1758 17322 +   %EM_REGISTER(KEY=GSCORETREE, TYPE=DATA);
1759 17323 +   %EM_GETNAME(KEY=GSCORETREE, TYPE=DATA);
1760 17324 +   %EM_REGISTER(KEY=GSCORECORR, TYPE=DATA);
1761 17325 +   %EM_GETNAME(KEY=GSCORECORR, TYPE=DATA);
1762 17326 +   %EM_REGISTER(KEY=GSCORECORRPLOT, TYPE=DATA);
1763 17327 +   %EM_GETNAME(KEY=GSCORECORRPLOT, TYPE=DATA);
1764 17328 +   %EM_REGISTER(KEY=GCLUSLINK, TYPE=DATA);
1765 17329 +   %EM_GETNAME(KEY=GCLUSLINK, TYPE=DATA);
1766 17330 +   %EM_REGISTER(KEY=GCLUSNODE, TYPE=DATA);
1767 17331 +   %EM_GETNAME(KEY=GCLUSNODE, TYPE=DATA);
1768 17332 +
1769 17333 +   filename gsfile "&gscorecode";
1770 17334 +
1771 17335 +   data &EM_USER_GSCORE;
1772 17336 +             set &indata;
1773 17337 +             %include  gsfile;
1774 17338 +        keep
1775 17339 +        %do i=1 %to &ngcluster;
1776 17340 +         %let gcvarname = GC&i;
1777 17341 +         &gcvarname
1778 17342 +        %end;
1779 17343 +     ;
1780 17344 +   run;
1781 17345 +
1782 17346 +   proc varclus data=&EM_USER_GSCORE outstat=&EM_USE
     R_GSCORESTAT outtree=&EM_USER_GSCORETREE
1783 17347 +    %if %upcase(&EM_PROPERTY_CLUSCOMP) eq CENTROID %
     then %do; centroid %end;
1784 17348 +    %if %upcase(&EM_PROPERTY_CLUSSOURCE) eq COV %the
     n %do; cov %end;
1785 17349 +    %if %upcase(&EM_PROPERTY_CLUSHIERACHY) eq Y %the
     n %do; hi %end;
1786 17350 +    noprint  ;
1787 17351 +     var
1788 17352 +       %do i=1 %to &ngcluster;
```

```
1789 17353 +         %let gcvarname = GC&i;
1790 17354 +         &gcvarname
1791 17355 +        %end;
1792 17356 +   ;
1793 17357 +   run;
1794 17358 +
1795 17359 +
1796 17360 +   %MakeVarClusCorrData(statds=&EM_USER_GSCORESTAT,
     corrds=&EM_USER_GSCORECORR, corrplotds=&EM_USER_GSCORECORRP
     LOT );
1797 17361 +   data &EM_USER_GSCORECORRPLOT ;
1798 17362 +        set &EM_USER_GSCORECORRPLOT;
1799 17363 +        rename _X_ = X;
1800 17364 +        rename _Y_ = Y;
1801 17365 +        label _X_ = "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_gcluster, noquote))" ;
1802 17366 +        label _Y_ = "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_gcluster, noquote))" ;
1803 17367 +   run;
1804 17368 +
1805 17369 +   %MakeGClusterConstData(indata=&EM_USER_GSCORETREE
     , inoutrsq=&outrsquare, outnode=&EM_USER_GCLUSNODE, outlink
     =&EM_USER_GCLUSLINK);
1806 17370 +
1807 17371 +   data &EM_USER_GSCORETREE;
1808 17372 +        length _NAME_ $32;
1809 17373 +        length _LABEL_ $100;
1810 17374 +        set &EM_USER_GSCORETREE(DROP=_LABEL_);
1811 17375 +        if upcase(substr(strip(_NAME_),1, 2))='GC' t
     hen do;
1812 17376 +         _LABEL_ = "%sysfunc(sasmsg(sashelp.dmine, r
     pt_varclus_label_gcluster, noquote)):"||_NAME_;
1813 17377 +        end;else do;
1814 17378 +         _LABEL_ = _NAME_;
1815 17379 +        end;
1816 17380 +       label _LABEL_ = "%sysfunc(sasmsg(sashelp.dmin
```

```
     e, rpt_varclus_label_variable, noquote))";
1817 17381 +
1818 17382 +   run;
1819 17383 +
1820 17384 +   quit;
1821 17385 +
1822 17386 +%mend MakeGClusterCorrelation;
1823 17387 +
1824 17388 +
1825 NOTE: %INCLUDE (level 1) ending.
1826 NOTE: Fileref TEMP has been deassigned.
1827 NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMEXPL.V
     ARIABLECLUSTERING_TRAIN.SOURCE.
1828 17391 +%macro train;
1829 17393 +   filename temp catalog 'sashelp.emexpl.variableclu
     stering_train1.source';
1830 17394 +   %include temp;
1831 17395 +   filename temp catalog 'sashelp.emexpl.variableclu
     stering_train2.source';
1832 17396 +   %include temp;
1833 17397 +   filename temp catalog 'sashelp.emutil.em_copyfile
     .source';
1834 17398 +   %include temp;
1835 17399 +   filename temp;
1836 17401 +   %let VARCLUS_MAXNUMOBS = 100000;
1837 17402 +   %let VARCLUS_MAXNUMVAR = 200;
1838 17403 +   %let trainnum = 0;
1839 17404 +   %let error = 0;
1840 17406 +   %if &EM_IMPORT_DATA eq %then %do;
1841 17407 +       %let  EMEXCEPTIONSTRING = exception.server.IM
     PORT.NOTRAIN,1;
1842 17408 +       %let  error = 1;
1843 17409 +       %goto endtrain;
1844 17410 +   %end;
1845 17412 +    %if (%sysfunc(exist(&EM_IMPORT_DATA)) or %sysfun
     c(exist(&EM_IMPORT_DATA, VIEW))) < 1 %then %do;
```

```
1846 17413 +        %let  EMEXCEPTIONSTRING = exception.server.I
     MPORT.NOTRAIN,1;
1847 17414 +        %let  error = 1;
1848 17415 +        %goto endline;
1849 17416 +    %end;
1850 17418 +   %let _num_input_interval = %eval(&EM_NUM_INTERVAL
     _INPUT+&EM_NUM_INTERVAL_REJECTED);
1851 17419 +   %let _num_input_binary = %eval(&EM_NUM_BINARY_INP
     UT+&EM_NUM_BINARY_REJECTED);
1852 17420 +   %let _num_input_nominal = %eval(&EM_NUM_NOMINAL_I
     NPUT+&EM_NUM_NOMINAL_REJECTED);
1853 17421 +   %let _num_input_ordinal = %eval(&EM_NUM_ORDINAL_I
     NPUT+&EM_NUM_ORDINAL_REJECTED);
1854 17422 +   %let _num_input_class = %eval(&_num_input_binary+
     &_num_input_nominal+&_num_input_ordinal);
1855 17423 +   %let _num_input_total = %eval(&_num_input_interva
     l+&_num_input_class);
1856 17425 +   %if (&EM_PROPERTY_INCLUDECLASSVAR eq Y) %then %do
     ;
1857 17426 +        %if  &_num_input_total < 2 %then %do;
1858 17427 +            %let  EMEXCEPTIONSTRING = exception.serv
     er.METADATA.USEATLEAST2INPUTREJECT;
1859 17428 +            %let error = 2;
1860 17429 +            %goto endtrain;
1861 17430 +        %end;
1862 17431 +   %end;
1863 17432 +   %else %do;
1864 17433 +         %if  &_num_input_interval < 2 %then %do;
1865 17434 +         %let  EMEXCEPTIONSTRING = exception.server.
     METADATA.USEATLEAST2INPUTREJECT;
1866 17435 +         %let error = 2;
1867 17436 +         %goto endtrain;
1868 17437 +         %end;
1869 17438 +   %end;
1870 17440 +   %em_checkerror();
1871 17442 +   %if &EMEXCEPTIONSTRING ne %then %do;
```

```
1872 17443 +       %goto endtrain;
1873 17444 +   %end;
1874 17446 +   %if (&EM_PROPERTY_INCLUDECLASSVAR eq Y) and (&_nu
     m_input_class > 0) %then %do;
1875 17448 +        %EM_GETNAME(key=OUTDUMMY, type=DATA);
1876 17449 +        %MakeDummyVariables( indata=&EM_IMPORT_DATA,
1877 17450 +                             outvar=&EM_USER_OUTDUMM
     Y,
1878 17451 +                             outdata=_newtrainds,
1879 17452 +                             fileref=&EM_FILE_EMFLOW
     SCORECODE);
1880 17454 +       %if &EM_PROPERTY_SUPPRESSSAMPWARN eq N %then
     %do;
1881 17455 +          %getNObs(inds=_newtrainds, nobs=_varclus_
     nobs);
1882 17456 +           %if &_varclus_nobs > &VARCLUS_MAXNUMOBS
      %then %do;
1883 17457 +               proc datasets lib=work nolist;
1884 17458 +               delete  _newtrainds;
1885 17459 +               run;
1886 17460 +               quit;
1887 17461 +               %let  EMEXCEPTIONSTRING = exception.
     server.varclus.sample.warning;
1888 17462 +               %let error = 3;
1889 17463 +              %goto endtrain;
1890 17464 +           %end;
1891 17465 +        %end;
1892 17467 +        %if &EM_PROPERTY_TWOSTAGECLUS eq AUTO %then
     %do;
1893 17468 +            %getNVar(inds=_newtrainds, nvar=_nvar);
1894 17470 +           %if &_nvar > &VARCLUS_MAXNUMVAR %then %d
     o;
1895 17471 +               %let trainnum = 2;
1896 17472 +           %end;
1897 17473 +           %else %do;
1898 17474 +               %let trainnum = 1;
```

```
1899 17475 +            %end;
1900 17476 +         %end;
1901 17477 +    %end;
1902 17478 +    %else %do;
1903 17480 +          %if &EM_PROPERTY_SUPPRESSSAMPWARN eq N %th
     en %do;
1904 17482 +            %getNObs(inds=&EM_IMPORT_DATA, nobs=_var
     clus_nobs);
1905 17484 +            %if &_varclus_nobs > &VARCLUS_MAXNUMOBS
      %then %do;
1906 17486 +                %let  EMEXCEPTIONSTRING = exception
     .server.varclus.sample.warning;
1907 17487 +                 %let error = 3;
1908 17488 +                %goto endtrain;
1909 17489 +            %end;
1910 17490 +         %end;
1911 17493 +          %if &EM_PROPERTY_TWOSTAGECLUS eq AUTO %then
      %do;
1912 17494 +            /*%getNVar(inds=&EM_IMPORT_DATA, nvar=_n
     var);
1913 17495 +              %if &_nvar > &VARCLUS_MAXNUMVAR %then
     %do;*/
1914 17496 +            %if &EM_NUM_INTERVAL_INPUT > &VARCLUS_MA
     XNUMVAR %then %do;
1915 17497 +                %let trainnum = 2;
1916 17498 +            %end;
1917 17499 +            %else %do;
1918 17500 +                %let trainnum = 1;
1919 17501 +            %end;
1920 17502 +         %end;
1921 17504 +   %end;
1922 17506 +    %em_checkerror();
1923 17508 +    %if &EMEXCEPTIONSTRING ne %then %do;
1924 17509 +       %let error = 4;
1925 17510 +       %goto endtrain;
1926 17511 +    %end;
```

```
1927 17513 +    %CreateVarclusMeta(trainnum=&trainnum);
1928 17515 +    %if &error > 0 %then  %goto endtrain;
1929 17517 +       /* when only EM_PROPERTY_TWOSTAGECLUS eq AUTO
      &_triannum will be 1 or 2  */
1930 17519 +       %if (&trainnum = 1 ) or %upcase(&EM_PROPERTY_
     TWOSTAGECLUS) = NO %then %do;
1931 17520 +          %train1;
1932 17521 +       %end;
1933 17522 +       %if (&trainnum = 2 ) or %upcase(&EM_PROPERTY_
     TWOSTAGECLUS) = YES %then %do;
1934 17523 +          %train2;
1935 17524 +       %end;
1936 17526 +      filename _in  "&EM_FILE_EMFLOWSCORECODE";
1937 17527 +      filename _out "&EM_FILE_EMPUBLISHSCORECODE";
1938 17528 +      data _null_;
1939 17529 +         length line $20000;
1940 17530 +         file _out lrecl=20000;
1941 17531 +         fid=fopen("_in",'i',20000,'v');
1942 17532 +         if fid > 0  then do;
1943 17533 +            do while(^fread(fid));
1944 17534 +               rlen = frlen(fid);
1945 17535 +               rc= fget(fid,line,20000);
1946 17536 +               start = length(line)-length(left(line
     ))+1;
1947 17537 +               line=strip(line);
1948 17538 +               if line ne 'delete;' then do;
1949 17539 +                  put @start line;
1950 17540 +               end;
1951 17541 +            end;
1952 17542 +            if fid > 0  then rc=fclose(fid);
1953 17543 +         end;
1954 17544 +      run;
1955 17545 +      filename _in;
1956 17546 +      filename _out;
1957 17548 +   %endtrain:
1958 17551 +%mend train;
```

```
1959 NOTE: %INCLUDE (level 1) ending.
1960 NOTE: Fileref TEMP has been deassigned.
1961 NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMEXPL.V
     ARIABLECLUSTERING_TRAIN1.SOURCE.
1962 17553 +%macro VarClus(indata=,
1963 17554 +                outstat=,
1964 17555 +                outtree=,
1965 17556 +                vars=,
1966 17557 +                freq=,
1967 17558 +                weight=,
1968 17559 +                includeclassvar=
1969 17560 +                );
1970 17562 +   %if &vars eq %then
1971 17563 +       %let vars = %EM_INTERVAL_INPUT %EM_INTERVAL_R
     EJECTED;
1972 17565 +        /*Add by ruzhan, May 28, 2013*/
1973 17566 +        ods graphics on;
1974 17567 +        %let odspath =;
1975 17568 +        data _null_;
1976 17569 +            path = pathname('WORK');
1977 17570 +            call symput('odspath', path);
1978 17571 +        run;
1979 17572 +        ods listing gpath="&odspath";
1980 17573 +        ods output DENDROGRAM=&outtree;
1981 17574 +       proc varclus data = &indata outstat= &outstat
1982 17575 +        %if %upcase(&EM_PROPERTY_CLUSCOMP) eq CENTRO
     ID %then %do; centroid %end;
1983 17576 +        %if %upcase(&EM_PROPERTY_CLUSSOURCE) eq COV
     %then %do; cov %end;
1984 17577 +        %if %upcase(&EM_PROPERTY_CLUSHIERACHY) eq Y
     %then %do; hi %end;
1985 17578 +        /*-----------------------
1986 17579 +        %if %upcase(&EM_PROPERTY_MAXCLUS) ne DEFAULT
      %then %do;
1987 17580 +            %let maxc = %sysevalf(&EM_PROPERTY_MAXCL
     US, int);
```

```
1988 17581 +            %if &maxc > 1 %then %do;     maxc=&maxc
     %end;
1989 17582 +       %end;
1990 17583 +       -------------------------*/
1991 17584 +       %if (&EM_PROPERTY_MAXCLUS ne . )  and (%upcas
     e(&EM_PROPERTY_MAXCLUS) ne DEFAULT) %then %do;
1992 17585 +            maxc = &EM_PROPERTY_MAXCLUS
1993 17586 +       %end;
1994 17588 +       /*-----------------------------------------
     ------------------------------------------------------
1995 17589 +        (%upcase(&EM_PROPERTY_MAXCLUS) ne DEFAULT) i
     s needed because of diagram conversion from Em12.1 to 12.3
1996 17590 +        -----------------------------------------
     ------------------------------------------------------*
     /
1997 17592 +       %if (%upcase(&EM_PROPERTY_CLUSCOMP) ne CENTRO
     ID) and
1998 17593 +           ( ( &EM_PROPERTY_MAXEIGEN ne . )  and (%u
     pcase(&EM_PROPERTY_MAXEIGEN) ne DEFAULT) ) %then %do;
1999 17594 +            maxeigen = &EM_PROPERTY_MAXEIGEN
2000 17595 +       %end;
2001 17597 +       %if &EM_PROPERTY_PROPORTION ne 0  and (%upcas
     e(&EM_PROPERTY_PROPORTION) ne DEFAULT) %then %do;
2002 17598 +            proportion = &EM_PROPERTY_PROPORTION
2003 17599 +       %end;
2004 17600 +       %if %upcase(&EM_PROPERTY_PRINTOPTION) eq SUMM
     ARY %then %do;
2005 17601 +              summary
2006 17602 +       %end;
2007 17604 +       %if %upcase(&EM_PROPERTY_PRINTOPTION) eq SHOR
     T %then %do;
2008 17605 +          short
2009 17606 +       %end;
2010 17607 +        %if %upcase(&EM_PROPERTY_PRINTOPTION) eq ALL
      %then %do;
2011 17608 +          corr trace simple
```

```
2012 17609 +       %end;
2013 17610 +       %if %upcase(&EM_PROPERTY_PRINTOPTION) eq NONE
      %then %do;
2014 17611 +          noprint
2015 17612 +       %end;
2016 17613 +        /*Comment out by ruzhan, May 28, 2013*/
2017 17614 +       /*       outtree=&outtree*/
2018 17615 +       ;
2019 17616 +        var %EM_INTERVAL_INPUT %EM_INTERVAL_REJECTED
2020 17618 +      %if &includeclassvar eq Y %then %do;
2021 17619 +          %let dsid = %sysfunc(open(&EM_USER_OUTDUMM
     Y));
2022 17620 +          %if &dsid > 0 %then %do;
2023 17621 +             %let nvar = %sysfunc(attrn(&dsid, NVAR)
     );
2024 17622 +              %do i = 2 %to &nvar;
2025 17623 +                  %let varname = %sysfunc(varname(&d
     sid, &i));
2026 17624 +                   &varname
2027 17625 +              %end;
2028 17627 +          %end;
2029 17628 +      %end;
2030 17630 +      ;
2031 17631 +      /*-------------------------
2032 17632 +      %if &freq ne  %then %do;
2033 17633 +          freq &freq;
2034 17634 +      %end;
2035 17635 +      %if &weight ne  %then %do;
2036 17636 +          weight &weight;
2037 17637 +      %end;
2038 17638 +      ----------------------------*/
2039 17639 +      %if %EM_FREQ ne %then %do;
2040 17640 +           Freq %EM_FREQ;
2041 17641 +      %end;
2042 17642 +      ;
2043 17643 +    run;
```

```
2044 17644 +    ods graphics off;
2045 17645 +    %endline:
2046 17646 +    quit;
2047 17647 +%mend VarClus;
2048 17649 +%macro Train1;
2049 17651 +   %EM_GETNAME(key=OUTCORRPLOT, type=DATA) ;
2050 17652 +   %EM_GETNAME(key=OUTCORR, type=DATA) ;
2051 17653 +   %EM_GETNAME(key=OUTSTAT, type=DATA) ;
2052 17654 +   %EM_GETNAME(key=OUTGROUP, type=DATA) ;
2053 17655 +   %EM_GETNAME(key=OUTSTATPLOT, type=DATA) ;
2054 17656 +   %EM_GETNAME(key=OUTTREE, type=DATA) ;
2055 17657 +   %EM_GETNAME(key=OUTRSQUARE, type=DATA) ;
2056 17658 +   %EM_GETNAME(key=OUTCLUSSTRUCT, type=DATA) ;
2057 17659 +   %EM_GETNAME(key=OUTCLUSCORR, type=DATA) ;
2058 17660 +   %EM_GETNAME(key=OUTCLUSCORRPLOT, type=DATA) ;
2059 17661 +   %EM_GETNAME(key=OUTCLUSRSQUARE, type=DATA) ;
2060 17662 +   %EM_GETNAME(key=OUTLINK, type=DATA) ;
2061 17663 +   %EM_GETNAME(key=OUTNODE, type=DATA) ;
2062 17665 +   /*
2063 17666 +     %let VARCLUS_MAXNUMOBS = 100;
2064 17667 +     %let VARCLUS_MAXNUMVAR = 200;
2065 17668 +   */
2066 17670 +   /* Train
2067 17672 +   %if &EM_IMPORT_DATA eq %then %do;
2068 17673 +       %let  EMEXCEPTIONSTRING = exception.server.IM
     PORT.NOTRAIN,1;
2069 17674 +       %goto endtrain1;
2070 17675 +   %end;
2071 17677 +   %if  &EM_NUM_INTERVAL_INPUT < 2 %then %do;
2072 17678 +        %let  EMEXCEPTIONSTRING = exception.server.M
     ETADATA.USEATLEAST2INPUTREJECT;
2073 17679 +        %goto endtrain1;
2074 17680 +   %end;
2075 17681 +   */
2076 17682 +   %if &EM_PROPERTY_INCLUDECLASSVAR eq Y %then %do;
2077 17683 +       /* moved this to pretrain ------------
```

```
2078 17685 +        %EM_GETNAME(key=OUTDUMMY, type=DATA);
2079 17686 +        %MakeDummyVariables( indata=&EM_IMPORT_DATA,
2080 17687 +                             outvar=&EM_USER_OUTDUMM
     Y,
2081 17688 +                             outdata=_newtrainds,
2082 17689 +                             fileref=&EM_FILE_EMFLOW
     SCORECODE);
2083 17691 +        %if &EM_PROPERTY_SUPPRESSSAMPWARN eq N %then
      %do;
2084 17692 +            %getNVarNObs(inds=_newtrainds, nvar=_nva
     r, nobs=_nobs);
2085 17693 +            %if &_nobs > &VARCLUS_MAXNUMOBS ne %then
      %do;
2086 17694 +                %let  EMEXCEPTIONSTRING = exception.
     server.varclus.sample.warning
2087 17695 +                %goto endtrain;
2088 17696 +            %end;
2089 17697 +        %end;
2090 17699 +        +-------------------------------------*/
2091 17701 +        %if (%sysfunc(exist(_newtrainds)) or %sysfun
     c(exist(_newtrainds, VIEW))) < 1 %then %do;
2092 17702 +        /*when there is no class var and &EM_PROPERT
     Y_INCLUDECLASSVAR eq Y
2093 17703 +          There is no _newtrainds */
2094 17704 +         %VarClus( indata=&EM_IMPORT_DATA,
2095 17705 +                  outstat=&EM_USER_OUTSTAT,
2096 17706 +                  outtree = &EM_USER_OUTTREE,
2097 17707 +                  includeclassvar=N
2098 17708 +                 );
2099 17709 +        %end;
2100 17710 +        %else %do;
2101 17711 +        %VarClus( indata=_newtrainds,
2102 17712 +                  outstat=&EM_USER_OUTSTAT,
2103 17713 +                  outtree = &EM_USER_OUTTREE,
2104 17714 +                  includeclassvar=Y
2105 17715 +                 );
```

```
2106 17716 +         proc datasets lib=work nolist;
2107 17717 +              delete  _newtrainds;
2108 17718 +         run;
2109 17719 +         %end;
2110 17720 +    %end;
2111 17721 +    %else %do;
2112 17723 +        /*----- moved this to pretrain ------------
2113 17724 +        %if &EM_PROPERTY_SUPPRESSSAMPWARN eq N %then
      %do;
2114 17725 +            %getNVarNObs(inds=&EM_IMPORT_DATA, nvar=
     _nvar, nobs=_nobs);
2115 17726 +            %if &_nobs > &VARCLUS_MAXNUMOBS ne %then
      %do;
2116 17727 +                %let  EMEXCEPTIONSTRING = exception
     .server.varclus.sample.warning;
2117 17728 +                %goto endtrain;
2118 17729 +            %end;
2119 17730 +        %end;
2120 17731 +        +-------------------------------------*/
2121 17733 +        %VarClus(indata=&EM_IMPORT_DATA,
2122 17734 +                 outstat=&EM_USER_OUTSTAT,
2123 17735 +                 outtree =&EM_USER_OUTTREE,
2124 17736 +                 includeclassvar=N
2125 17737 +               );
2126 17739 +   %end;
2127 17741 +   %em_checkerror();
2128 17743 +   %if &EMEXCEPTIONSTRING ne %then %do;
2129 17744 +       %goto endtrain1;
2130 17745 +   %end;
2131 17746 +    /*
2132 17747 +    %FindClusNum(statds=&EM_USER_OUTSTAT, groupds=&E
     M_USER_OUTGROUP, minvariation=&EM_PROPERTY_MINVARIATION);
2133 17748 +    */
2134 17750 +    %getNclusfromTrain(inoutstat=&EM_USER_OUTSTAT, n
     c=_nclus);
2135 17752 +    %let gid=;
```

```
2136 17754 +    /*--- To make the main result table ------------
     ----------*/
2137 17756 +    %MakeInterClusCorrData(indata=&EM_USER_OUTSTAT,
     globalclusid=&gid, outdata=&EM_USER_OUTCLUSCORR, ncluster=&
     _nclus, RSquare=Y);
2138 17757 +    %FindNextClosestClusByCluster(indata=&EM_USER_OU
     TCLUSCORR, outdata=_nextClusRSq, globalclusid=&gid, ncluste
     r=&_nclus);
2139 17758 +    %MakeOwnRSquare(indata=&EM_USER_OUTSTAT, globalc
     lusid=&gid, outdata=_ownRsq, ncluster=&_nclus);
2140 17759 +    %MakeClusStructCorrData(indata=&EM_USER_OUTSTAT,
      globalclusid=&gid, outdata=&EM_USER_OUTCLUSSTRUCT , nclust
     er=&_nclus, Rsquare=Y);
2141 17760 +    %FindNextClosestClusByVar(indata=&EM_USER_OUTCLU
     SSTRUCT, outdata=_nextVarRSq, globalclusid=&gid, ncluster=&
     _nclus);
2142 17761 +    %MakeVarClusResultTable(indata1=_ownRsq, indata2
     =_nextVarRsq, indata3=_nextClusRSq, outdata=&EM_USER_OUTRSQ
     UARE, globalclusid=&gid, ncluster=&_nclus, selectedcomp=&EM
     _PROPERTY_EXPORTEDCOMP);
2143 17763 +   /* Inter Cluster Correlation Plot -------------
     ---------*/
2144 17765 +    %MakeInterClusCorrData(indata=&EM_USER_OUTSTAT,
     globalclusid=&gid, outdata=&EM_USER_OUTCLUSCORRPLOT, nclust
     er=&_nclus, RSquare=N);
2145 17766 +    %MakePlotDataFromCorrTable(indata=&EM_USER_OUTCL
     USCORRPLOT, outdata=&EM_USER_OUTCLUSCORRPLOT, globalclusid=
     &gid);
2146 17768 +   /* Create some other data sets for results ----*/
2147 17770 +   %MakeStatPlotData(statds=&EM_USER_OUTSTAT , outst
     atplotds=&EM_USER_OUTSTATPLOT);
2148 17771 +   %MakeVarClusCorrData(statds=&EM_USER_OUTSTAT, cor
     rds=&EM_USER_OUTCORR, corrplotds=&EM_USER_OUTCORRPLOT);
2149 17772 +   %MakeClusConstellData(indata=&EM_USER_OUTRSQUARE,
      outlink=&EM_USER_OUTLINK, outnode=&EM_USER_OUTNODE);
2150 17774 +  /*---- move this to Score action ------------
```

```
2151 17775 +   %MakeDeltaCode(groupds=&EM_USER_OUTRSQUARE, Delta
     CodeFile=&EM_FILE_CDELTA_TRAIN);
2152 17776 +   */
2153 17778 +   /* create score code at training time */
2154 17779 +   %CreateScoreCode(indata=&EM_USER_OUTSTAT, nclusco
     mp=&_nclus, fileref=&EM_FILE_EMFLOWSCORECODE);
2155 17780 +   %endtrain1:
2156 17781 +%mend Train1;
2157 NOTE: %INCLUDE (level 1) ending.
2158 NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMEXPL.V
     ARIABLECLUSTERING_TRAIN2.SOURCE.
2159 17784 +%macro VarClus2( indata=,
2160 17785 +                 outdata=,
2161 17786 +                 outstat =outstat,
2162 17787 +                 outrsquare=outrsquare,
2163 17788 +                 outtree = outtree,
2164 17789 +                 outclusstruct = outclusstruct,
2165 17790 +                 outcluscorr =outcluscorr,
2166 17791 +                 outcluscorrplot =outcluscorrplot,
2167 17792 +                 outclusrsquare =outclusrsquare,
2168 17793 +                 outstatplot = outstatplot,
2169 17794 +                 outcorr = outcorr,
2170 17795 +                 outcorrplot = outcorrplot,
2171 17796 +                 clusdata=,
2172 17797 +                 outnode=outnode,
2173 17798 +                 outlink=outlink,
2174 17799 +                 ngroup =1  );
2175 17800 +         %if (%sysfunc(exist(&indata)) or %sysfunc(e
     xist(&indata, VIEW))) < 1 %then %do;
2176 17801 +              %let  EMEXCEPTIONSTRING = exception.se
     rver.IMPORT.NOTRAIN,1;
2177 17802 +              %goto endline;
2178 17803 +         %end;
2179 17805 +        %global numglobalcluster;
2180 17806 +        %let  numglobalcluster = &ngroup;
2181 17807 +        %let  gscorefile =  %bquote(&EM_NODEDIR)&EM_
```

```
     DSEP.gclusterscore.sas;
2182 17808 +        %do vci =1 %to &ngroup;
2183 17809 +           data tmp_clusdata;
2184 17810 +                set &clusdata;
2185 17811 +                if CLUSTER ne &vci then delete;
2186 17812 +           run;
2187 17813 +           %let outstatds = &outstat.&vci;
2188 17814 +           %let outrsquareds = &outrsquare.&vci;
2189 17815 +           %let outtreeds = &outtree.&vci;
2190 17816 +           %let outclusstuctds = &outclusstruct.&vci
     ;
2191 17817 +           %let outcluscorrds = &outcluscorr.&vci;
2192 17818 +           %let outcluscorrplotds = &outcluscorrplot
     .&vci;
2193 17819 +           %let outclusrsquareds = &outclussquare.&v
     ci;
2194 17820 +           %let outstatplotds = &outstatplot.&vci;
2195 17821 +           %let outcorrds = &outcorr.&vci;
2196 17822 +           %let outcorrplotds = &outcorrplot.&vci;
2197 17823 +           %let outnodeds = &outnode.&vci;
2198 17824 +           %let outlinkds = &outlink.&vci;
2199 17826 +           %EM_REGISTER(KEY=&outstatds, TYPE=DATA);
2200 17827 +           %EM_GETNAME(KEY=&outstatds, TYPE=DATA);
2201 17828 +           %let emuseroutstat = &EM_USER_&outstatds;
2202 17830 +           %EM_REGISTER(KEY=&outtreeds, TYPE=DATA);
2203 17831 +           %EM_GETNAME(KEY=&outtreeds, TYPE=DATA);
2204 17832 +           %let emuserouttree = &EM_USER_&outtreeds;
2205 17834 +           %EM_REGISTER(KEY=&outcluscorrds, TYPE=DAT
     A);
2206 17835 +           %EM_GETNAME(KEY=&outcluscorrds, TYPE=DATA
     );
2207 17836 +           %let emuseroutcluscorr = &EM_USER_&outclu
     scorrds;
2208 17838 +           %EM_REGISTER(KEY=&outclusstuctds, TYPE=DA
     TA);
2209 17839 +           %EM_GETNAME(KEY=&outclusstuctds, TYPE=DAT
```

```
A);
2210 17840 +          %let emuseroutclusstruct = &EM_USER_&outc
lusstuctds;
2211 17842 +          %EM_REGISTER(KEY=&outrsquareds, TYPE=DATA
);
2212 17843 +          %EM_GETNAME(KEY=&outrsquareds, TYPE=DATA)
;
2213 17844 +          %let emuseroutrsquare = &EM_USER_&outrsqu
areds;
2214 17846 +          %EM_REGISTER(KEY=&outcluscorrplotds, TYPE
=DATA);
2215 17847 +          %EM_GETNAME(KEY=&outcluscorrplotds, TYPE=
DATA);
2216 17848 +          %let emuseroutcluscorrplot = &EM_USER_&ou
tcluscorrplotds;
2217 17850 +          %EM_REGISTER(KEY=&outstatplotds, TYPE=DAT
A);
2218 17851 +          %EM_GETNAME(KEY=&outstatplotds, TYPE=DATA
);
2219 17852 +          %let emuseroutstatplot = &EM_USER_&outsta
tplotds;
2220 17854 +          %EM_REGISTER(KEY=&outcorrds, TYPE=DATA);
2221 17855 +          %EM_GETNAME(KEY=&outcorrds, TYPE=DATA);
2222 17856 +          %let emuseroutcorr = &EM_USER_&outcorrds;
2223 17858 +          %EM_REGISTER(KEY=&outcorrplotds, TYPE=DAT
A);
2224 17859 +          %EM_GETNAME(KEY=&outcorrplotds, TYPE=DATA
);
2225 17860 +          %let emuseroutcorrplot = &EM_USER_&outcor
rplotds;
2226 17862 +          %EM_REGISTER(KEY=&outnodeds, TYPE=DATA);
2227 17863 +          %EM_GETNAME(KEY=&outnodeds, TYPE=DATA);
2228 17864 +          %let emuseroutnode = &EM_USER_&outnodeds;
2229 17866 +          %EM_REGISTER(KEY=&outlinkds, TYPE=DATA);
2230 17867 +          %EM_GETNAME(KEY=&outlinkds, TYPE=DATA);
2231 17868 +          %let emuseroutlink = &EM_USER_&outlinkds;
```

```
2232 17870 +           /*Add by ruzhan, May 28, 2013*/
2233 17871 +           ods graphics on;
2234 17872 +            %let odspath =;
2235 17873 +            data _null_;
2236 17874 +                path = pathname('WORK');
2237 17875 +                call symput('odspath', path);
2238 17876 +            run;
2239 17877 +            ods listing gpath="&odspath";
2240 17879 +           ods output DENDROGRAM=&emuserouttree;
2241 17881 +           proc varclus data=&indata outstat=&emuser
     outstat
2242 17882 +            %if %upcase(&EM_PROPERTY_CLUSCOMP) eq CE
     NTROID %then %do; centroid %end;
2243 17883 +            %if %upcase(&EM_PROPERTY_CLUSSOURCE) eq
     COV %then %do; cov %end;
2244 17884 +            %if %upcase(&EM_PROPERTY_CLUSHIERACHY) e
     q Y %then %do; hi %end;
2245 17885 +            %if (&EM_PROPERTY_MAXCLUS ne .)  and (%u
     pcase(&EM_PROPERTY_MAXCLUS) ne DEFAULT) %then %do;
2246 17886 +                  maxc=&EM_PROPERTY_MAXCLUS
2247 17887 +           %end;
2248 17889 +            %if (%upcase(&EM_PROPERTY_CLUSCOMP) ne C
     ENTROID) and
2249 17890 +                ( (&EM_PROPERTY_MAXEIGEN ne .)  and
     (%upcase(&EM_PROPERTY_MAXEIGEN) ne DEFAULT)) %then %do;
2250 17891 +            maxeigen = &EM_PROPERTY_MAXEIGEN
2251 17892 +            %end;
2252 17894 +           %if (&EM_PROPERTY_PROPORTION ne 0)  and (
     %upcase(&EM_PROPERTY_PROPORTION) ne DEFAULT) %then %do;
2253 17895 +                proportion = &EM_PROPERTY_PROPORTION
2254 17896 +           %end;
2255 17897 +           %if %upcase(&EM_PROPERTY_PRINTOPTION) eq
     SUMMARY %then %do;
2256 17898 +              summary
2257 17899 +           %end;
2258 17900 +           %if %upcase(&EM_PROPERTY_PRINTOPTION) eq
```

```
     SHORT %then %do;
2259 17901 +              short
2260 17902 +           %end;
2261 17903 +            %if %upcase(&EM_PROPERTY_PRINTOPTION) eq
      ALL %then %do;
2262 17904 +              corr trace simple
2263 17905 +           %end;
2264 17906 +           %if %upcase(&EM_PROPERTY_PRINTOPTION) eq
     NONE %then %do;
2265 17907 +              noprint
2266 17908 +           %end;
2267 17909 +            /*Comment out by ruzhan, May 28, 2013*/
2268 17910 +              /*outtree=&emuserouttree*/
2269 17911 +           ;
2270 17912 +           var
2271 17913 +           %let dsid=%sysfunc(open(work.tmp_clusdata
     ));
2272 17914 +           %let vn_name =%sysfunc(varnum(&dsid, VARI
     ABLE));
2273 17915 +               %do %while(^%sysfunc(fetch(&dsid)));
2274 17916 +                   %let _varname= %sysfunc(getvarc(
     &dsid, &vn_name));
2275 17917 +                   &_varname
2276 17918 +               %end;
2277 17919 +            %let dsid = %sysfunc(close(&dsid));
2278 17920 +           ;
2279 17921 +           %if %EM_FREQ ne %then %do;
2280 17922 +           Freq %EM_FREQ;
2281 17923 +           %end;
2282 17924 +           run;
2283 17925 +            ods graphics off;
2284 17926 +            %getNclusfromTrain(inoutstat=&emuserouts
     tat, nc=_nclus);
2285 17928 +            %let gid=&vci;
2286 17929 +            /*--- To make the main result table ----
     ------------------*/
```

```
2287 17930 +           %MakeInterClusCorrData(indata=&emuserout
     stat, globalclusid=&gid, outdata=&emuseroutcluscorr, nclust
     er=&_nclus, RSquare=Y);
2288 17931 +           %FindNextClosestClusByCluster(indata=&em
     useroutcluscorr, outdata=_nextClusRSq, globalclusid=&gid, n
     cluster=&_nclus);
2289 17932 +           %MakeOwnRSquare(indata=&emuseroutstat, g
     lobalclusid=&gid, outdata=_ownRsq, ncluster=&_nclus);
2290 17933 +           %MakeClusStructCorrData(indata=&emuserou
     tstat, globalclusid=&gid, outdata=&emuseroutclusstruct , nc
     luster=&_nclus, Rsquare=Y);
2291 17934 +           %FindNextClosestClusByVar(indata=&emuser
     outclusstruct, outdata=_nextVarRSq, globalclusid=&gid, nclu
     ster=&_nclus);
2292 17935 +           %MakeVarClusResultTable(indata1=_ownRsq,
      indata2=_nextVarRsq, indata3=_nextClusRSq, outdata=&emuser
     outrsquare, globalclusid=&gid, ncluster=&_nclus, selectedco
     mp=&EM_PROPERTY_EXPORTEDCOMP);
2293 17937 +           /* Inter Cluster Correlation Plot ------
     ------------------*/
2294 17939 +           %MakeInterClusCorrData(indata=&emuserout
     stat, globalclusid=&gid, outdata=&emuseroutcluscorrplot, nc
     luster=&_nclus, RSquare=N);
2295 17940 +           %MakePlotDataFromCorrTable(indata=&emuse
     routcluscorrplot, outdata=&emuseroutcluscorrplot, globalclu
     sid=&gid);
2296 17942 +           /* Create some other data sets for result
     s ---------------*/
2297 17944 +           %MakeStatPlotData(statds=&emuseroutstat ,
      outstatplotds=&emuseroutstatplot);
2298 17945 +           %MakeVarClusCorrData(statds=&emuseroutsta
     t, corrds=&emuseroutcorr, corrplotds=&emuseroutcorrplot);
2299 17946 +           %MakeClusConstellData(indata=&emuseroutrs
     quare, outlink=&emuseroutlink, outnode=&emuseroutnode);
2300 17948 +           %CreateScoreCode2(indata=&emuseroutstat,
     ncluscomp=&_nclus, globalclusid=&gid, fileref=&EM_FILE_EMFL
```

```
     OWSCORECODE);
2301 17950 +           %CreateGClusterScoreCode(indata=&emuserou
     tstat,  globalclusid=&gid, fileref=&gscorefile);
2302 17952 +           %if &vci = 1 %then %do;
2303 17953 +           data &EM_USER_OUTRSQUARE;
2304 17954 +                set &emuseroutrsquare;
2305 17955 +           run;
2306 17956 +           %end;
2307 17957 +           %else %do;
2308 17958 +           data &EM_USER_OUTRSQUARE;
2309 17959 +                set &EM_USER_OUTRSQUARE &emuseroutrs
     quare;
2310 17960 +           run;
2311 17961 +           %end;
2312 17962 +           quit;
2313 17963 +       %end;
2314 17965 +       /*---- move this to Score action ------------
2315 17966 +       %MakeDeltaCode2(groupds=&EM_USER_OUTRSQUARE,
     DeltaCodeFile=&EM_FILE_CDELTA_TRAIN);
2316 17967 +       */
2317 17969 +       /* Make Score Gcluster component & correlatio
     n  */
2318 17970 +       %MakeGClusterCorrelation(Indata=&indata, ngcl
     uster=&ngroup, gscorecode=&gscorefile, outrsquare=&EM_USER_
     OUTRSQUARE);
2319 17972 +       quit;
2320 17973 +       %endline:
2321 17974 +%mend VarClus2;
2322 17977 +%macro Train2;
2323 17979 +   %EM_GETNAME(key=OUTCORRPLOT, type=DATA) ;
2324 17980 +   %EM_GETNAME(key=OUTCORR, type=DATA) ;
2325 17981 +   %EM_GETNAME(key=OUTSTAT, type=DATA) ;
2326 17982 +   %EM_GETNAME(key=OUTGROUP, type=DATA) ;
2327 17983 +   %EM_GETNAME(key=OUTSTATPLOT, type=DATA) ;
2328 17984 +   %EM_GETNAME(key=OUTTREE, type=DATA) ;
2329 17985 +   %EM_GETNAME(key=OUTRSQUARE, type=DATA) ;
```

```
2330 17986 +   %EM_GETNAME(key=OUTCLUSSTRUCT, type=DATA) ;
2331 17987 +   %EM_GETNAME(key=OUTCLUSCORR, type=DATA) ;
2332 17988 +   %EM_GETNAME(key=OUTCLUSCORRPLOT, type=DATA) ;
2333 17989 +   %EM_GETNAME(key=OUTCLUSRSQUARE, type=DATA) ;
2334 17990 +   %EM_GETNAME(key=GOUTSTAT, type=DATA) ;
2335 17991 +   %EM_GETNAME(key=GOUTTREE, type=DATA) ;
2336 17992 +   %EM_GETNAME(key=GOUTCORR, type=DATA) ;
2337 17993 +   %EM_GETNAME(key=GOUTGROUP, type=DATA) ;
2338 17994 +   %EM_GETNAME(key=GOUTNODE, type=DATA) ;
2339 17995 +   %EM_GETNAME(key=GOUTLINK, type=DATA) ;
2340 17996 +   %EM_GETNAME(key=VARCLUSMETA, type=DATA) ;
2341 18000 +    /* Train */
2342 18002 +   /* moved to pretrain ------------
2343 18004 +   %if &EM_IMPORT_DATA eq %then %do;
2344 18005 +       %let  EMEXCEPTIONSTRING = exception.server.IM
     PORT.NOTRAIN,1;
2345 18006 +       %goto endtrain2;
2346 18007 +   %end;
2347 18009 +   %if  &EM_NUM_INTERVAL_INPUT < 2 %then %do;
2348 18010 +       %let  EMEXCEPTIONSTRING = exception.server.M
     ETADATA.USEATLEAST2INPUTREJECT;
2349 18011 +       %goto endtrain2;
2350 18012 +   %end;
2351 18013 +   +-----------------------------------------*/
2352 18015 +    %let _num_input_interval = %eval(&EM_NUM_INTERVA
     L_INPUT+&EM_NUM_INTERVAL_REJECTED);
2353 18018 +    %if &EM_PROPERTY_INCLUDECLASSVAR eq Y %then %do;
2354 18020 +       /*----- moved to pretrain ------------
2355 18022 +       %EM_GETNAME(key=OUTDUMMY, type=DATA);
2356 18023 +       %MakeDummyVariables( indata=&EM_IMPORT_DATA,
2357 18024 +                            outvar=&EM_USER_OUTDUMM
     Y,
2358 18025 +                            outdata=_newtrainds,
2359 18026 +                            fileref=&EM_FILE_EMFLOW
     SCORECODE,
2360 18027 +                             ndummyvars=_ndummyvars
```

```
2361 18028 +                              );
2362 18030 +        %if &EM_PROPERTY_SUPPRESSSAMPWARN eq N %then
      %do;
2363 18031 +            %getNVarNObs(inds=_newtrainds, nvar=_nva
     r, nobs=_nobs);
2364 18032 +            %if &_nobs > &VARCLUS_MAXNUMOBS  %then %
     do;
2365 18033 +                %let  EMEXCEPTIONSTRING = exception.
     server.varclus.sample.warning
2366 18034 +                %goto endtrain;
2367 18035 +            %end;
2368 18036 +        %end;
2369 18038 +        +---------------------------------------*/
2370 18039 +        %if (%sysfunc(exist(_newtrainds)) or %sysfun
     c(exist(_newtrainds, VIEW))) < 1 %then %do;
2371 18040 +        /*when there is no class var and &EM_PROPERT
     Y_INCLUDECLASSVAR eq Y
2372 18041 +          There is no _newtrainds */
2373 18042 +        %MakeCorrelation(    indata=&EM_IMPORT_DATA,
2374 18043 +                             outstat=&EM_USER_GOUTST
     AT,
2375 18044 +                             corrmatrix=Y,
2376 18045 +                             outcorr=&EM_USER_GOUTCO
     RR,
2377 18046 +                             includeclassvar=N,
2378 18047 +                             freq =%EM_FREQ
2379 18048 +                           );
2380 18049 +        %getInitialGClusterNumber(ninput=&_num_input
     _interval, ngc=_ngc);
2381 18051 +        %end;
2382 18052 +        %else %do;
2383 18053 +         %MakeCorrelation(   indata=_newtrainds,
2384 18054 +                             outstat=&EM_USER_GOUTST
     AT,
2385 18055 +                             corrmatrix=Y,
2386 18056 +                             outcorr=&EM_USER_GOUTCO
```

```
     RR,
2387 18057 +                              includeclassvar=Y,
2388 18058 +                              freq =%EM_FREQ
2389 18059 +                            );
2390 18060 +        %getInitialGClusterNumber(ninput=&_num_input
     _interval, ndummy=&_ndummyvars, ngc=_ngc);
2391 18062 +        %end;
2392 18064 +    %end;
2393 18065 +    %else %do;
2394 18067 +        /* moved to pretrain -------------------
2395 18069 +         %if &EM_PROPERTY_SUPPRESSSAMPWARN eq N %the
     n %do;
2396 18070 +            %getNVarNObs(inds=&EM_IMPORT_DATA, nvar=
     _nvar, nobs=_nobs);
2397 18071 +            %if &_nobs > &VARCLUS_MAXNUMOBS  %then %
     do;
2398 18072 +                %let  EMEXCEPTIONSTRING = exception.
     server.varclus.sample.warning
2399 18073 +                %goto endtrain;
2400 18074 +            %end;
2401 18075 +        %end;
2402 18076 +         +----------------------------------------
     */
2403 18078 +        %MakeCorrelation( indata=&EM_IMPORT_DATA,
2404 18079 +                          outstat=&EM_USER_GOUTSTAT,
2405 18080 +                          corrmatrix=Y,
2406 18081 +                          outcorr=&EM_USER_GOUTCORR,
2407 18082 +                          includeclassvar=N,
2408 18083 +                          freq =%EM_FREQ
2409 18084 +                          );
2410 18085 +        %getInitialGClusterNumber(ninput=&_num_input
     _interval, ngc=_ngc);
2411 18086 +    %end;
2412 18088 +   %UpdateOutStatCorrToDistance(indata=&EM_USER_GOUT
     STAT, /* indata should be a outstat from proc varclus */
2413 18089 +                                rsquare = N
```

```
2414 18090 +                                  );
2415 18091 +   %HierClusWithCorr( indata=&EM_USER_GOUTSTAT ,
2416 18092 +                     ncluster=&_ngc,
2417 18093 +                     method = Ward,
2418 18094 +                     outtree =&EM_USER_GOUTTREE,
2419 18095 +                     idvar = _VAR_,
2420 18096 +                     outdata=&EM_USER_GOUTGROUP,
2421 18097 +                     rescore = N,
2422 18098 +                     newncluster=_newnclus
2423 18099 +                       );
2424 18101 +   /* store the number of global cluster */
2425 18103 +   data &EM_USER_VARCLUSMETA;
2426 18104 +        set &EM_USER_VARCLUSMETA;
2427 18105 +        NGCluster = &_newnclus;
2428 18106 +   run;
2429 18108 +   %MakeGobalConstellData(indata=&EM_USER_GOUTGROUP,
      outlink=&EM_USER_GOUTLINK, outnode=&EM_USER_GOUTNODE);
2430 18110 +   %if &EM_PROPERTY_INCLUDECLASSVAR eq Y %then %do;
2431 18111 +        %if (%sysfunc(exist(_newtrainds)) or %sysfun
     c(exist(_newtrainds, VIEW))) < 1 %then %do;
2432 18112 +           %VarClus2(indata=&EM_IMPORT_DATA,
2433 18113 +                clusdata=&EM_USER_GOUTGROUP,
2434 18114 +                ngroup=&_newnclus);
2435 18115 +         %end;
2436 18116 +         %else %do;
2437 18117 +            %VarClus2(indata=_newtrainds,
2438 18118 +                clusdata=&EM_USER_GOUTGROUP,
2439 18119 +                ngroup=&_newnclus);
2440 18121 +         %end;
2441 18123 +   %end;
2442 18124 +   %else %do;
2443 18125 +       %VarClus2(indata=&EM_IMPORT_DATA,
2444 18126 +                clusdata=&EM_USER_GOUTGROUP,
2445 18127 +                ngroup=&_newnclus);
2446 18128 +   %end;
2447 18130 +   %em_checkerror();
```

```
2448  18131 +   %if &EMEXCEPTIONSTRING ne %then %do;
2449  18132 +       %put &em_codebar;
2450  18133 +       %put Error: Variable clustering failed.;
2451  18134 +       %put &em_codebar;
2452  18135 +       %goto endtrain2;
2453  18136 +   %end;
2454  18137 +   /*
2455  18138 +    %FindClusNum(statds=&EM_USER_OUTSTAT, groupds=&E
      M_USER_OUTGROUP, minvariation=&EM_PROPERTY_MINVARIATION);
2456  18139 +   */
2457  18140 +   %endtrain2:
2458  18141 +%mend Train2;
2459  NOTE: %INCLUDE (level 1) ending.
2460  NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMUTIL.E
      M_COPYFILE.SOURCE.
2461  18143 +%macro em_copyfile(infref=, outfref=, append=N);
2462  18144 +
2463  18145 +   %if %sysfunc(fileref(&infref))=0 and %sysfunc(fil
      eref(&outfref))<=0 %then %do;
2464  18146 +
2465  18147 +       %let omode =;
2466  18148 +       %if &append=Y %then
2467  18149 +           %let omode = MOD;
2468  18150 +       data _null_;
2469  18151 +          length line $20000;
2470  18152 +          file &outfref &omode lrecl=20000;
2471  18153 +          fid=fopen("&infref",'i',20000,'v');
2472  18154 +          if fid > 0  then do;
2473  18155 +             do while(^fread(fid));
2474  18156 +                rlen = frlen(fid);
2475  18157 +                rc= fget(fid,line,20000);
2476  18158 +                start = length(line)-length(left(lin
      e))+1;
2477  18159 +                line=strip(line);
2478  18160 +                put @start line;
2479  18161 +             end;
```

```
2480  18162 +             if fid > 0  then rc=fclose(fid);
2481  18163 +          end;
2482  18164 +       run;
2483  18165 +
2484  18166 +   %end;
2485  18167 +%mend em_copyfile;
2486  NOTE: %INCLUDE (level 1) ending.
2487  NOTE: Fileref TEMP has been deassigned.
2488
2489  18168  %let SYSCC = 0;
2490  NOTE: PROCEDURE DISPLAY used (Total process time):
2491        real time           0.00 seconds
2492        user cpu time       0.00 seconds
2493        system cpu time     0.00 seconds
2494        memory              24708.75k
2495        OS Memory           34936.00k
2496        Timestamp           07/01/2024 05:54:26 AM
2497        Step Count                        1  Switch Count  1
2498        Page Faults                       0
2499        Page Reclaims                     95
2500        Page Swaps                        0
2501        Voluntary Context Switches        1
2502        Involuntary Context Switches      0
2503        Block Input Operations            0
2504        Block Output Operations           0
2505
2506
2507
2508  NOTE: Records processed = 24999   Memory used = 511K.
2509  NOTE: There were 24999 observations read from the data set
      CHURN.EM_SAVE_TRAIN.
2510  NOTE: View EMWS4.IMPT_TRAIN.VIEW used (Total process time):
2511        real time           0.09 seconds
2512        user cpu time       0.03 seconds
2513        system cpu time     0.06 seconds
2514        memory              158482.00k
```

```
2515       OS Memory           167800.00k
2516       Timestamp           07/01/2024 05:54:26 AM
2517       Step Count                       1  Switch Count  11
2518       Page Faults                      0
2519       Page Reclaims                    33498
2520       Page Swaps                       0
2521       Voluntary Context Switches       29
2522       Involuntary Context Switches     0
2523       Block Input Operations           0
2524       Block Output Operations          72
2525
2526 NOTE: There were 24999 observations read from the data set
     EMWS4.IDS_DATA.
2527 NOTE: There were 24999 observations read from the data set
     EMWS4.IMPT_TRAIN.
2528 NOTE: The data set WORK._DMDBDAT has 24999 observations and
      1 variables.
2529 NOTE: The data set WORK._CLASSOUT has 2 observations and 9
     variables.
2530 NOTE: PROCEDURE DMDB used (Total process time):
2531       real time           0.11 seconds
2532       user cpu time       0.04 seconds
2533       system cpu time     0.07 seconds
2534       memory              158482.00k
2535       OS Memory           167800.00k
2536       Timestamp           07/01/2024 05:54:26 AM
2537       Step Count                       1  Switch Count  6
2538       Page Faults                      0
2539       Page Reclaims                    33664
2540       Page Swaps                       0
2541       Voluntary Context Switches       33
2542       Involuntary Context Switches     0
2543       Block Input Operations           0
2544       Block Output Operations          584
2545
2546
```

```
2547
2548 NOTE: DATA statement used (Total process time):
2549       real time           0.00 seconds
2550       user cpu time       0.00 seconds
2551       system cpu time     0.00 seconds
2552       memory              158482.00k
2553       OS Memory           167800.00k
2554       Timestamp           07/01/2024 05:54:26 AM
2555       Step Count                        1  Switch Count  0
2556       Page Faults                       0
2557       Page Reclaims                     64
2558       Page Swaps                        0
2559       Voluntary Context Switches        0
2560       Involuntary Context Switches      0
2561       Block Input Operations            0
2562       Block Output Operations           0
2563
2564
2565 NOTE: This is the PROC statement.
2566 NOTE: The training set WORK._DMDBDAT.DATA has 1 variable(s)
     .
2567 NOTE: The PROC statement has finished with return code 0.
2568 NOTE: This is the INPUT statement.
2569 NOTE: 1 input variable(s) defined for ID=I1.
2570 NOTE: The INPUT statement has finished with return code 0.
2571 NOTE: This is the MAKE statement.
2572
2573 NOTE: 1 input variable(s).
2574 NOTE: Number of cases=24999
2575 NOTE: Sum of frequencies=24999
2576 NOTE: Sum of weights=24999
2577 NOTE: VARDEF=DF
2578 NOTE: Maximum number of categories=2
2579 NOTE: The total number of variables is 1 with dimensionalit
     y 2.
2580 NOTE: The data set EMWS4.VARCLUS_OUTDUMMY has 3 observation
```

```
     s and 3 variables.
2581 NOTE: The MAKE statement has finished with return code 0.
2582 NOTE: View EMWS4.IMPT_TRAIN.VIEW used (Total process time):
2583       real time           0.09 seconds
2584       user cpu time       0.04 seconds
2585       system cpu time     0.05 seconds
2586       memory              159319.37k
2587       OS Memory           169600.00k
2588       Timestamp           07/01/2024 05:54:27 AM
2589       Step Count                        1  Switch Count  13
2590       Page Faults                       0
2591       Page Reclaims                     33481
2592       Page Swaps                        0
2593       Voluntary Context Switches        30
2594       Involuntary Context Switches      0
2595       Block Input Operations            0
2596       Block Output Operations           8456
2597
2598 NOTE: This is the SCORE statement.
2599 NOTE: The data set WORK._NEWTRAINDS has 24999 observations
     and 17 variables.
2600 NOTE: There were 24999 observations read from the data set
     CHURN.EM_SAVE_TRAIN.
2601 NOTE: There were 24999 observations read from the data set
     EMWS4.IDS_DATA.
2602 NOTE: The SCORE statement has finished with return code 0.
2603 NOTE: This is the CODE statement.
2604 NOTE: External file /home/u63452984/case-study-s2192852/Wor
     kspaces/EMWS4/VarClus/EMFLOWSCORE.sas opened.
2605 NOTE: The CODE statement has finished with return code 0.
2606
2607
2608 NOTE: PROCEDURE DMZIP used (Total process time):
2609       real time           0.12 seconds
2610       user cpu time       0.04 seconds
2611       system cpu time     0.07 seconds
```

```
2612        memory                  159319.37k
2613        OS Memory               169600.00k
2614        Timestamp               07/01/2024 05:54:27 AM
2615        Step Count                          1  Switch Count  8
2616        Page Faults                         0
2617        Page Reclaims                       33873
2618        Page Swaps                          0
2619        Voluntary Context Switches          71
2620        Involuntary Context Switches        0
2621        Block Input Operations              0
2622        Block Output Operations             8728
2623
2624
2625
2626 NOTE: The file WORK._DMDBCAT (memtype=DATA) was not found,
     but appears on a DELETE statement.
2627 NOTE: Deleting WORK._DMDBDAT (memtype=DATA).
2628 NOTE: Deleting WORK._CLASSOUT (memtype=DATA).
2629 NOTE: PROCEDURE DATASETS used (Total process time):
2630        real time               0.00 seconds
2631        user cpu time           0.00 seconds
2632        system cpu time         0.00 seconds
2633        memory                  159319.37k
2634        OS Memory               169600.00k
2635        Timestamp               07/01/2024 05:54:27 AM
2636        Step Count                          1  Switch Count  0
2637        Page Faults                         0
2638        Page Reclaims                       111
2639        Page Swaps                          0
2640        Voluntary Context Switches          0
2641        Involuntary Context Switches        0
2642        Block Input Operations              0
2643        Block Output Operations             8
2644
2645
2646
```

```
2647 NOTE: Numeric values have been converted to character value
     s at the places given by: (Line):(Column).
2648       2:167
2649 NOTE: There were 24999 observations read from the data set
     WORK._NEWTRAINDS.
2650 NOTE: DATA statement used (Total process time):
2651       real time           0.00 seconds
2652       user cpu time       0.00 seconds
2653       system cpu time     0.00 seconds
2654       memory              159319.37k
2655       OS Memory           169600.00k
2656       Timestamp           07/01/2024 05:54:27 AM
2657       Step Count                        1  Switch Count  0
2658       Page Faults                       0
2659       Page Reclaims                     279
2660       Page Swaps                        0
2661       Voluntary Context Switches        0
2662       Involuntary Context Switches      0
2663       Block Input Operations            0
2664       Block Output Operations           0
2665
2666
2667
2668 NOTE: Numeric values have been converted to character value
     s at the places given by: (Line):(Column).
2669       2:226
2670 NOTE: DATA statement used (Total process time):
2671       real time           0.00 seconds
2672       user cpu time       0.00 seconds
2673       system cpu time     0.00 seconds
2674       memory              159319.37k
2675       OS Memory           169600.00k
2676       Timestamp           07/01/2024 05:54:27 AM
2677       Step Count                        1  Switch Count  0
2678       Page Faults                       0
2679       Page Reclaims                     252
```

```
2680       Page Swaps                          0
2681       Voluntary Context Switches          0
2682       Involuntary Context Switches        0
2683       Block Input Operations              0
2684       Block Output Operations             0
2685
2686
2687
2688 18169  %let SYSCC = 0;
2689 NOTE: PROCEDURE DISPLAY used (Total process time):
2690       real time           0.00 seconds
2691       user cpu time       0.01 seconds
2692       system cpu time     0.00 seconds
2693       memory              159319.37k
2694       OS Memory           169600.00k
2695       Timestamp           07/01/2024 05:54:27 AM
2696       Step Count                          1  Switch Count  1
2697       Page Faults                         0
2698       Page Reclaims                       55
2699       Page Swaps                          0
2700       Voluntary Context Switches          1
2701       Involuntary Context Switches        0
2702       Block Input Operations              0
2703       Block Output Operations             8
2704
2705
2706
2707 NOTE: The data set EMWS4.VARCLUS_VARCLUSMETA has 1 observat
     ions and 5 variables.
2708 NOTE: DATA statement used (Total process time):
2709       real time           0.00 seconds
2710       user cpu time       0.00 seconds
2711       system cpu time     0.00 seconds
2712       memory              159319.37k
2713       OS Memory           169600.00k
2714       Timestamp           07/01/2024 05:54:27 AM
```

```
2715       Step Count                          1  Switch Count  0
2716       Page Faults                         0
2717       Page Reclaims                       90
2718       Page Swaps                          0
2719       Voluntary Context Switches          13
2720       Involuntary Context Switches        0
2721       Block Input Operations              0
2722       Block Output Operations             264
2723
2724
2725
2726 NOTE: DATA statement used (Total process time):
2727       real time           0.00 seconds
2728       user cpu time       0.00 seconds
2729       system cpu time     0.00 seconds
2730       memory              159319.37k
2731       OS Memory           169600.00k
2732       Timestamp           07/01/2024 05:54:27 AM
2733       Step Count                          1  Switch Count  0
2734       Page Faults                         0
2735       Page Reclaims                       28
2736       Page Swaps                          0
2737       Voluntary Context Switches          0
2738       Involuntary Context Switches        0
2739       Block Input Operations              0
2740       Block Output Operations             0
2741
2742
2743
2744 NOTE: Clustering algorithm converged.
2745 NOTE: Clustering algorithm converged.
2746 NOTE: The data set EMWS4.VARCLUS_OUTTREE has 8 observations
      and 3 variables.
2747 NOTE: The data set EMWS4.VARCLUS_OUTSTAT has 27 observation
     s and 8 variables.
2748 NOTE: The PROCEDURE VARCLUS printed page 2.
```

```
2749 NOTE: PROCEDURE VARCLUS used (Total process time):
2750       real time           2.09 seconds
2751       user cpu time       0.05 seconds
2752       system cpu time     0.01 seconds
2753       memory              159319.37k
2754       OS Memory           169600.00k
2755       Timestamp           07/01/2024 05:54:29 AM
2756       Step Count                        1  Switch Count  0
2757       Page Faults                       0
2758       Page Reclaims                     3934
2759       Page Swaps                        0
2760       Voluntary Context Switches        893
2761       Involuntary Context Switches      0
2762       Block Input Operations            288
2763       Block Output Operations           944
2764
2765
2766
2767 NOTE: Deleting WORK._NEWTRAINDS (memtype=DATA).
2768
2769 NOTE: PROCEDURE DATASETS used (Total process time):
2770       real time           0.00 seconds
2771       user cpu time       0.00 seconds
2772       system cpu time     0.00 seconds
2773       memory              159319.37k
2774       OS Memory           169600.00k
2775       Timestamp           07/01/2024 05:54:29 AM
2776       Step Count                        1  Switch Count  0
2777       Page Faults                       0
2778       Page Reclaims                     50
2779       Page Swaps                        0
2780       Voluntary Context Switches        0
2781       Involuntary Context Switches      0
2782       Block Input Operations            0
2783       Block Output Operations           8
2784
```

```
2785
2786
2787 18170  %let SYSCC = 0;
2788 NOTE: PROCEDURE DISPLAY used (Total process time):
2789       real time           0.00 seconds
2790       user cpu time       0.01 seconds
2791       system cpu time     0.00 seconds
2792       memory              159319.37k
2793       OS Memory           169600.00k
2794       Timestamp           07/01/2024 05:54:29 AM
2795       Step Count                        1  Switch Count  1
2796       Page Faults                       0
2797       Page Reclaims                     63
2798       Page Swaps                        0
2799       Voluntary Context Switches        1
2800       Involuntary Context Switches      0
2801       Block Input Operations            0
2802       Block Output Operations           0
2803
2804
2805
2806 NOTE: Numeric values have been converted to character value
     s at the places given by: (Line):(Column).
2807       5:214
2808 NOTE: There were 27 observations read from the data set EMW
     S4.VARCLUS_OUTSTAT.
2809 NOTE: DATA statement used (Total process time):
2810       real time           0.00 seconds
2811       user cpu time       0.00 seconds
2812       system cpu time     0.00 seconds
2813       memory              159319.37k
2814       OS Memory           169600.00k
2815       Timestamp           07/01/2024 05:54:29 AM
2816       Step Count                        1  Switch Count  0
2817       Page Faults                       0
2818       Page Reclaims                     63
```

```
2819       Page Swaps                          0
2820       Voluntary Context Switches          10
2821       Involuntary Context Switches        0
2822       Block Input Operations              288
2823       Block Output Operations             0
2824
2825
2826
2827 NOTE: Numeric values have been converted to character value
     s at the places given by: (Line):(Column).
2828       7:15
2829 NOTE: Character values have been converted to numeric value
     s at the places given by: (Line):(Column).
2830       7:9
2831 NOTE: There were 27 observations read from the data set EMW
     S4.VARCLUS_OUTSTAT.
2832 NOTE: The data set EMWS4.VARCLUS_OUTCLUSCORR has 2 observat
     ions and 6 variables.
2833 NOTE: DATA statement used (Total process time):
2834       real time           0.00 seconds
2835       user cpu time       0.00 seconds
2836       system cpu time     0.00 seconds
2837       memory              159319.37k
2838       OS Memory           169600.00k
2839       Timestamp           07/01/2024 05:54:29 AM
2840       Step Count                          1  Switch Count  0
2841       Page Faults                         0
2842       Page Reclaims                       269
2843       Page Swaps                          0
2844       Voluntary Context Switches          17
2845       Involuntary Context Switches        0
2846       Block Input Operations              0
2847       Block Output Operations             264
2848
2849
2850
```

```
2851 NOTE: There were 2 observations read from the data set EMWS
     4.VARCLUS_OUTCLUSCORR.
2852 NOTE: The data set WORK.CORR_TMP has 2 observations and 6 v
     ariables.
2853 NOTE: DATA statement used (Total process time):
2854       real time           0.00 seconds
2855       user cpu time       0.00 seconds
2856       system cpu time     0.01 seconds
2857       memory              159319.37k
2858       OS Memory           169600.00k
2859       Timestamp           07/01/2024 05:54:29 AM
2860       Step Count                        1  Switch Count  0
2861       Page Faults                       0
2862       Page Reclaims                     126
2863       Page Swaps                        0
2864       Voluntary Context Switches        9
2865       Involuntary Context Switches      0
2866       Block Input Operations            288
2867       Block Output Operations           264
2868
2869
2870
2871 NOTE: There were 2 observations read from the data set EMWS
     4.VARCLUS_OUTCLUSCORR.
2872 NOTE: The data set EMWS4.VARCLUS_OUTCLUSCORR has 2 observat
     ions and 3 variables.
2873 NOTE: DATA statement used (Total process time):
2874       real time           0.01 seconds
2875       user cpu time       0.00 seconds
2876       system cpu time     0.00 seconds
2877       memory              159319.37k
2878       OS Memory           169600.00k
2879       Timestamp           07/01/2024 05:54:29 AM
2880       Step Count                        1  Switch Count  0
2881       Page Faults                       0
2882       Page Reclaims                     368
```

```
2883       Page Swaps                          0
2884       Voluntary Context Switches          28
2885       Involuntary Context Switches        0
2886       Block Input Operations              0
2887       Block Output Operations             264
2888
2889
2890
2891 NOTE: There were 2 observations read from the data set EMWS
     4.VARCLUS_OUTCLUSCORR.
2892 NOTE: The data set WORK.CORR_TMP has 2 observations and 3 v
     ariables.
2893 NOTE: DATA statement used (Total process time):
2894       real time           0.00 seconds
2895       user cpu time       0.00 seconds
2896       system cpu time     0.00 seconds
2897       memory              159319.37k
2898       OS Memory           169600.00k
2899       Timestamp           07/01/2024 05:54:29 AM
2900       Step Count                          1  Switch Count  0
2901       Page Faults                         0
2902       Page Reclaims                       129
2903       Page Swaps                          0
2904       Voluntary Context Switches          12
2905       Involuntary Context Switches        0
2906       Block Input Operations              288
2907       Block Output Operations             264
2908
2909
2910
2911 NOTE: There were 2 observations read from the data set EMWS
     4.VARCLUS_OUTCLUSCORR.
2912 NOTE: The data set EMWS4.VARCLUS_OUTCLUSCORR has 2 observat
     ions and 3 variables.
2913 NOTE: DATA statement used (Total process time):
2914       real time           0.01 seconds
```

```
2915       user cpu time       0.00 seconds
2916       system cpu time     0.00 seconds
2917       memory              159319.37k
2918       OS Memory           169600.00k
2919       Timestamp           07/01/2024 05:54:29 AM
2920       Step Count                        1  Switch Count  0
2921       Page Faults                       0
2922       Page Reclaims                     164
2923       Page Swaps                        0
2924       Voluntary Context Switches        30
2925       Involuntary Context Switches      0
2926       Block Input Operations            0
2927       Block Output Operations           264
2928
2929
2930
2931  NOTE: Deleting WORK.CORR_TMP (memtype=DATA).
2932
2933  NOTE: PROCEDURE DATASETS used (Total process time):
2934       real time           0.00 seconds
2935       user cpu time       0.00 seconds
2936       system cpu time     0.00 seconds
2937       memory              159319.37k
2938       OS Memory           169600.00k
2939       Timestamp           07/01/2024 05:54:29 AM
2940       Step Count                        1  Switch Count  0
2941       Page Faults                       0
2942       Page Reclaims                     50
2943       Page Swaps                        0
2944       Voluntary Context Switches        0
2945       Involuntary Context Switches      0
2946       Block Input Operations            0
2947       Block Output Operations           8
2948
2949
2950
```

```
2951 NOTE: There were 2 observations read from the data set EMWS
     4.VARCLUS_OUTCLUSCORR.
2952 NOTE: The data set WORK._TMPCLUSRSQ has 2 observations and
     3 variables.
2953 NOTE: PROCEDURE SORT used (Total process time):
2954       real time           0.00 seconds
2955       user cpu time       0.01 seconds
2956       system cpu time     0.00 seconds
2957       memory              159319.37k
2958       OS Memory           169600.00k
2959       Timestamp           07/01/2024 05:54:29 AM
2960       Step Count                        1  Switch Count  0
2961       Page Faults                       0
2962       Page Reclaims                     173
2963       Page Swaps                        0
2964       Voluntary Context Switches        10
2965       Involuntary Context Switches      0
2966       Block Input Operations            288
2967       Block Output Operations           272
2968
2969
2970
2971 NOTE: There were 2 observations read from the data set WORK
     ._TMPCLUSRSQ.
2972 NOTE: The data set WORK._TMPCLUSRSQ has 4 observations and
     4 variables.
2973 NOTE: PROCEDURE TRANSPOSE used (Total process time):
2974       real time           0.00 seconds
2975       user cpu time       0.00 seconds
2976       system cpu time     0.00 seconds
2977       memory              159319.37k
2978       OS Memory           169600.00k
2979       Timestamp           07/01/2024 05:54:29 AM
2980       Step Count                        1  Switch Count  0
2981       Page Faults                       0
2982       Page Reclaims                     257
```

```
2983       Page Swaps                          0
2984       Voluntary Context Switches          0
2985       Involuntary Context Switches        0
2986       Block Input Operations              0
2987       Block Output Operations             528
2988
2989
2990
2991  NOTE: There were 4 observations read from the data set WORK
      ._TMPCLUSRSQ.
2992  NOTE: The data set WORK._TMPCLUSRSQ has 4 observations and
      4 variables.
2993  NOTE: PROCEDURE SORT used (Total process time):
2994       real time           0.00 seconds
2995       user cpu time       0.00 seconds
2996       system cpu time     0.01 seconds
2997       memory              159319.37k
2998       OS Memory           169600.00k
2999       Timestamp           07/01/2024 05:54:29 AM
3000       Step Count                          1  Switch Count  0
3001       Page Faults                         0
3002       Page Reclaims                       117
3003       Page Swaps                          0
3004       Voluntary Context Switches          0
3005       Involuntary Context Switches        0
3006       Block Input Operations              0
3007       Block Output Operations             264
3008
3009
3010
3011  NOTE: There were 4 observations read from the data set WORK
      ._TMPCLUSRSQ.
3012  NOTE: The data set WORK._TMPCLUSRSQ has 2 observations and
      4 variables.
3013  NOTE: DATA statement used (Total process time):
3014       real time           0.00 seconds
```

```
3015       user cpu time       0.00 seconds
3016       system cpu time     0.00 seconds
3017       memory              159319.37k
3018       OS Memory           169600.00k
3019       Timestamp           07/01/2024 05:54:29 AM
3020       Step Count                        1  Switch Count  0
3021       Page Faults                       0
3022       Page Reclaims                     139
3023       Page Swaps                        0
3024       Voluntary Context Switches        0
3025       Involuntary Context Switches      0
3026       Block Input Operations            0
3027       Block Output Operations           264
3028
3029
3030
3031 NOTE: There were 2 observations read from the data set WORK
     ._TMPCLUSRSQ.
3032 NOTE: The data set WORK._NEXTCLUSRSQ has 2 observations and
      4 variables.
3033 NOTE: DATA statement used (Total process time):
3034       real time           0.00 seconds
3035       user cpu time       0.00 seconds
3036       system cpu time     0.00 seconds
3037       memory              159319.37k
3038       OS Memory           169600.00k
3039       Timestamp           07/01/2024 05:54:29 AM
3040       Step Count                        1  Switch Count  0
3041       Page Faults                       0
3042       Page Reclaims                     474
3043       Page Swaps                        0
3044       Voluntary Context Switches        0
3045       Involuntary Context Switches      0
3046       Block Input Operations            0
3047       Block Output Operations           264
3048
```

```
3049
3050
3051 NOTE: Deleting WORK._TMPCLUSRSQ (memtype=DATA).
3052
3053 NOTE: PROCEDURE DATASETS used (Total process time):
3054       real time           0.00 seconds
3055       user cpu time       0.00 seconds
3056       system cpu time     0.00 seconds
3057       memory              159319.37k
3058       OS Memory           169600.00k
3059       Timestamp           07/01/2024 05:54:29 AM
3060       Step Count                        1  Switch Count  0
3061       Page Faults                       0
3062       Page Reclaims                     48
3063       Page Swaps                        0
3064       Voluntary Context Switches        0
3065       Involuntary Context Switches      0
3066       Block Input Operations            0
3067       Block Output Operations           8
3068
3069
3070
3071 NOTE: Numeric values have been converted to character value
     s at the places given by: (Line):(Column).
3072       2:77
3073 NOTE: Character values have been converted to numeric value
     s at the places given by: (Line):(Column).
3074       2:71
3075 NOTE: There were 27 observations read from the data set EMW
     S4.VARCLUS_OUTSTAT.
3076 NOTE: The data set WORK._TMPDS has 2 observations and 7 var
     iables.
3077 NOTE: DATA statement used (Total process time):
3078       real time           0.00 seconds
3079       user cpu time       0.00 seconds
3080       system cpu time     0.00 seconds
```

```
3081        memory                  159319.37k
3082        OS Memory               169600.00k
3083        Timestamp               07/01/2024 05:54:29 AM
3084        Step Count                          1  Switch Count  0
3085        Page Faults                         0
3086        Page Reclaims                       248
3087        Page Swaps                          0
3088        Voluntary Context Switches          4
3089        Involuntary Context Switches        0
3090        Block Input Operations              0
3091        Block Output Operations             264
3092
3093
3094
3095 NOTE: There were 2 observations read from the data set WORK
     ._TMPDS.
3096 NOTE: The data set WORK._OWNRSQ has 5 observations and 4 va
     riables.
3097 NOTE: PROCEDURE TRANSPOSE used (Total process time):
3098        real time               0.00 seconds
3099        user cpu time           0.00 seconds
3100        system cpu time         0.00 seconds
3101        memory                  159319.37k
3102        OS Memory               169600.00k
3103        Timestamp               07/01/2024 05:54:29 AM
3104        Step Count                          1  Switch Count  0
3105        Page Faults                         0
3106        Page Reclaims                       225
3107        Page Swaps                          0
3108        Voluntary Context Switches          0
3109        Involuntary Context Switches        0
3110        Block Input Operations              0
3111        Block Output Operations             528
3112
3113
3114
```

```
3115 NOTE: Numeric values have been converted to character value
     s at the places given by: (Line):(Column).
3116       12:35
3117 NOTE: Variable GCluster is uninitialized.
3118 NOTE: There were 5 observations read from the data set WORK
     ._OWNRSQ.
3119 NOTE: The data set WORK._OWNRSQ has 5 observations and 4 va
     riables.
3120 NOTE: DATA statement used (Total process time):
3121       real time           0.00 seconds
3122       user cpu time       0.00 seconds
3123       system cpu time     0.00 seconds
3124       memory              159319.37k
3125       OS Memory           169600.00k
3126       Timestamp           07/01/2024 05:54:29 AM
3127       Step Count                        1  Switch Count  0
3128       Page Faults                       0
3129       Page Reclaims                     591
3130       Page Swaps                        0
3131       Voluntary Context Switches        0
3132       Involuntary Context Switches      0
3133       Block Input Operations            0
3134       Block Output Operations           264
3135
3136
3137
3138 NOTE: There were 5 observations read from the data set WORK
     ._OWNRSQ.
3139 NOTE: The data set WORK._OWNRSQ has 5 observations and 4 va
     riables.
3140 NOTE: PROCEDURE SORT used (Total process time):
3141       real time           0.00 seconds
3142       user cpu time       0.00 seconds
3143       system cpu time     0.00 seconds
3144       memory              159319.37k
3145       OS Memory           169600.00k
```

```
3146        Timestamp           07/01/2024 05:54:29 AM
3147        Step Count                        1  Switch Count  0
3148        Page Faults                       0
3149        Page Reclaims                     116
3150        Page Swaps                        0
3151        Voluntary Context Switches        0
3152        Involuntary Context Switches      0
3153        Block Input Operations            0
3154        Block Output Operations           264
3155
3156
3157
3158  NOTE: Deleting WORK._TMPDS (memtype=DATA).
3159
3160  NOTE: PROCEDURE DATASETS used (Total process time):
3161        real time           0.00 seconds
3162        user cpu time       0.00 seconds
3163        system cpu time     0.00 seconds
3164        memory              159319.37k
3165        OS Memory           169600.00k
3166        Timestamp           07/01/2024 05:54:29 AM
3167        Step Count                        1  Switch Count  0
3168        Page Faults                       0
3169        Page Reclaims                     48
3170        Page Swaps                        0
3171        Voluntary Context Switches        0
3172        Involuntary Context Switches      0
3173        Block Input Operations            0
3174        Block Output Operations           8
3175
3176
3177
3178  NOTE: Numeric values have been converted to character value
      s at the places given by: (Line):(Column).
3179        4:40
3180  NOTE: Character values have been converted to numeric value
```

```
     s at the places given by: (Line):(Column).
3181       4:34
3182 NOTE: Variable GCluster is uninitialized.
3183 NOTE: There were 27 observations read from the data set EMW
     S4.VARCLUS_OUTSTAT.
3184 NOTE: The data set EMWS4.VARCLUS_OUTCLUSSTRUCT has 2 observ
     ations and 6 variables.
3185 NOTE: DATA statement used (Total process time):
3186       real time           0.00 seconds
3187       user cpu time       0.00 seconds
3188       system cpu time     0.00 seconds
3189       memory              159319.37k
3190       OS Memory           169600.00k
3191       Timestamp           07/01/2024 05:54:29 AM
3192       Step Count                        1  Switch Count  0
3193       Page Faults                       0
3194       Page Reclaims                     358
3195       Page Swaps                        0
3196       Voluntary Context Switches        16
3197       Involuntary Context Switches      0
3198       Block Input Operations            0
3199       Block Output Operations           264
3200
3201
3202
3203 NOTE: There were 2 observations read from the data set EMWS
     4.VARCLUS_OUTCLUSSTRUCT.
3204 NOTE: The data set WORK.CORR_TMP has 2 observations and 6 v
     ariables.
3205 NOTE: DATA statement used (Total process time):
3206       real time           0.00 seconds
3207       user cpu time       0.00 seconds
3208       system cpu time     0.00 seconds
3209       memory              159319.37k
3210       OS Memory           169600.00k
3211       Timestamp           07/01/2024 05:54:29 AM
```

```
3212       Step Count                         1  Switch Count  0
3213       Page Faults                        0
3214       Page Reclaims                      127
3215       Page Swaps                         0
3216       Voluntary Context Switches         10
3217       Involuntary Context Switches       0
3218       Block Input Operations             288
3219       Block Output Operations            264
3220
3221
3222
3223 NOTE: There were 2 observations read from the data set EMWS
     4.VARCLUS_OUTCLUSSTRUCT.
3224 NOTE: The data set EMWS4.VARCLUS_OUTCLUSSTRUCT has 2 observ
     ations and 6 variables.
3225 NOTE: DATA statement used (Total process time):
3226       real time           0.01 seconds
3227       user cpu time       0.01 seconds
3228       system cpu time     0.01 seconds
3229       memory              159319.37k
3230       OS Memory           169600.00k
3231       Timestamp           07/01/2024 05:54:29 AM
3232       Step Count                         1  Switch Count  0
3233       Page Faults                        0
3234       Page Reclaims                      164
3235       Page Swaps                         0
3236       Voluntary Context Switches         30
3237       Involuntary Context Switches       0
3238       Block Input Operations             0
3239       Block Output Operations            264
3240
3241
3242
3243 NOTE: Deleting WORK.CORR_TMP (memtype=DATA).
3244
3245 NOTE: PROCEDURE DATASETS used (Total process time):
```

```
3246        real time           0.00 seconds
3247        user cpu time       0.00 seconds
3248        system cpu time     0.00 seconds
3249        memory              159319.37k
3250        OS Memory           169600.00k
3251        Timestamp           07/01/2024 05:54:29 AM
3252        Step Count                        1  Switch Count  0
3253        Page Faults                       0
3254        Page Reclaims                     49
3255        Page Swaps                        0
3256        Voluntary Context Switches        0
3257        Involuntary Context Switches      0
3258        Block Input Operations            0
3259        Block Output Operations           8
3260
3261
3262
3263 NOTE: There were 2 observations read from the data set EMWS
     4.VARCLUS_OUTCLUSSTRUCT.
3264 NOTE: The data set WORK._TMPCLUSRSQ has 2 observations and
     6 variables.
3265 NOTE: PROCEDURE SORT used (Total process time):
3266        real time           0.00 seconds
3267        user cpu time       0.00 seconds
3268        system cpu time     0.00 seconds
3269        memory              159319.37k
3270        OS Memory           169600.00k
3271        Timestamp           07/01/2024 05:54:29 AM
3272        Step Count                        1  Switch Count  0
3273        Page Faults                       0
3274        Page Reclaims                     150
3275        Page Swaps                        0
3276        Voluntary Context Switches        10
3277        Involuntary Context Switches      0
3278        Block Input Operations            288
3279        Block Output Operations           272
```

```
3280
3281
3282
3283 NOTE: There were 2 observations read from the data set WORK
     ._TMPCLUSRSQ.
3284 NOTE: The data set WORK._TMPCLUSRSQ has 10 observations and
      4 variables.
3285 NOTE: PROCEDURE TRANSPOSE used (Total process time):
3286       real time           0.00 seconds
3287       user cpu time       0.00 seconds
3288       system cpu time     0.00 seconds
3289       memory              159319.37k
3290       OS Memory           169600.00k
3291       Timestamp           07/01/2024 05:54:29 AM
3292       Step Count                        1  Switch Count  0
3293       Page Faults                       0
3294       Page Reclaims                     227
3295       Page Swaps                        0
3296       Voluntary Context Switches        0
3297       Involuntary Context Switches      0
3298       Block Input Operations            0
3299       Block Output Operations           528
3300
3301
3302
3303 NOTE: There were 10 observations read from the data set WOR
     K._TMPCLUSRSQ.
3304 NOTE: The data set WORK._TMPCLUSRSQ has 10 observations and
      4 variables.
3305 NOTE: PROCEDURE SORT used (Total process time):
3306       real time           0.00 seconds
3307       user cpu time       0.00 seconds
3308       system cpu time     0.00 seconds
3309       memory              159319.37k
3310       OS Memory           169600.00k
3311       Timestamp           07/01/2024 05:54:29 AM
```

```
3312        Step Count                         1  Switch Count  0
3313        Page Faults                        0
3314        Page Reclaims                      117
3315        Page Swaps                         0
3316        Voluntary Context Switches         0
3317        Involuntary Context Switches       0
3318        Block Input Operations             0
3319        Block Output Operations            264
3320
3321
3322
3323 NOTE: There were 10 observations read from the data set WOR
     K._TMPCLUSRSQ.
3324 NOTE: The data set WORK._TMPCLUSRSQ has 5 observations and
     4 variables.
3325 NOTE: DATA statement used (Total process time):
3326        real time           0.00 seconds
3327        user cpu time       0.00 seconds
3328        system cpu time     0.00 seconds
3329        memory              159319.37k
3330        OS Memory           169600.00k
3331        Timestamp           07/01/2024 05:54:29 AM
3332        Step Count                         1  Switch Count  0
3333        Page Faults                        0
3334        Page Reclaims                      130
3335        Page Swaps                         0
3336        Voluntary Context Switches         0
3337        Involuntary Context Switches       0
3338        Block Input Operations             0
3339        Block Output Operations            272
3340
3341
3342
3343 NOTE: There were 5 observations read from the data set WORK
     ._TMPCLUSRSQ.
3344 NOTE: The data set WORK._TMPCLUSRSQ has 5 observations and
```

```
     4 variables.
3345 NOTE: PROCEDURE SORT used (Total process time):
3346       real time           0.00 seconds
3347       user cpu time       0.00 seconds
3348       system cpu time     0.00 seconds
3349       memory              159319.37k
3350       OS Memory           169600.00k
3351       Timestamp           07/01/2024 05:54:29 AM
3352       Step Count                        1  Switch Count  0
3353       Page Faults                       0
3354       Page Reclaims                     118
3355       Page Swaps                        0
3356       Voluntary Context Switches        0
3357       Involuntary Context Switches      0
3358       Block Input Operations            0
3359       Block Output Operations           264
3360
3361
3362
3363 NOTE: There were 5 observations read from the data set WORK
     ._TMPCLUSRSQ.
3364 NOTE: The data set WORK._NEXTVARRSQ has 5 observations and
     4 variables.
3365 NOTE: DATA statement used (Total process time):
3366       real time           0.00 seconds
3367       user cpu time       0.01 seconds
3368       system cpu time     0.00 seconds
3369       memory              159319.37k
3370       OS Memory           169600.00k
3371       Timestamp           07/01/2024 05:54:29 AM
3372       Step Count                        1  Switch Count  0
3373       Page Faults                       0
3374       Page Reclaims                     471
3375       Page Swaps                        0
3376       Voluntary Context Switches        0
3377       Involuntary Context Switches      0
```

```
3378       Block Input Operations            0
3379       Block Output Operations           264
3380
3381
3382
3383  NOTE: Deleting WORK._TMPCLUSRSQ (memtype=DATA).
3384
3385  NOTE: PROCEDURE DATASETS used (Total process time):
3386        real time           0.00 seconds
3387        user cpu time       0.00 seconds
3388        system cpu time     0.00 seconds
3389        memory              159319.37k
3390        OS Memory           169600.00k
3391        Timestamp           07/01/2024 05:54:29 AM
3392        Step Count                        1  Switch Count  0
3393        Page Faults                       0
3394        Page Reclaims                     48
3395        Page Swaps                        0
3396        Voluntary Context Switches        0
3397        Involuntary Context Switches      0
3398        Block Input Operations            0
3399        Block Output Operations           8
3400
3401
3402
3403  NOTE: There were 5 observations read from the data set WORK
      ._OWNRSQ.
3404  NOTE: The data set WORK._OWNRSQ has 5 observations and 4 va
      riables.
3405  NOTE: PROCEDURE SORT used (Total process time):
3406        real time           0.00 seconds
3407        user cpu time       0.00 seconds
3408        system cpu time     0.00 seconds
3409        memory              159319.37k
3410        OS Memory           169600.00k
3411        Timestamp           07/01/2024 05:54:29 AM
```

```
3412       Step Count                         1  Switch Count  0
3413       Page Faults                        0
3414       Page Reclaims                      117
3415       Page Swaps                         0
3416       Voluntary Context Switches         0
3417       Involuntary Context Switches       0
3418       Block Input Operations             0
3419       Block Output Operations            264
3420
3421
3422
3423 NOTE: There were 5 observations read from the data set WORK
     ._NEXTVARRSQ.
3424 NOTE: The data set WORK._NEXTVARRSQ has 5 observations and
     4 variables.
3425 NOTE: PROCEDURE SORT used (Total process time):
3426       real time           0.00 seconds
3427       user cpu time       0.00 seconds
3428       system cpu time     0.00 seconds
3429       memory              159319.37k
3430       OS Memory           169600.00k
3431       Timestamp           07/01/2024 05:54:29 AM
3432       Step Count                         1  Switch Count  0
3433       Page Faults                        0
3434       Page Reclaims                      117
3435       Page Swaps                         0
3436       Voluntary Context Switches         0
3437       Involuntary Context Switches       0
3438       Block Input Operations             0
3439       Block Output Operations            264
3440
3441
3442
3443 NOTE: There were 5 observations read from the data set WORK
     ._OWNRSQ.
3444 NOTE: There were 5 observations read from the data set WORK
```

```
     ._NEXTVARRSQ.
3445 NOTE: The data set EMWS4.VARCLUS_OUTRSQUARE has 5 observati
     ons and 7 variables.
3446 NOTE: DATA statement used (Total process time):
3447       real time           0.01 seconds
3448       user cpu time       0.00 seconds
3449       system cpu time     0.00 seconds
3450       memory              159319.37k
3451       OS Memory           169600.00k
3452       Timestamp           07/01/2024 05:54:29 AM
3453       Step Count                        1  Switch Count  0
3454       Page Faults                       0
3455       Page Reclaims                     287
3456       Page Swaps                        0
3457       Voluntary Context Switches        16
3458       Involuntary Context Switches      0
3459       Block Input Operations            0
3460       Block Output Operations           264
3461
3462
3463
3464 NOTE: There were 2 observations read from the data set WORK
     ._NEXTCLUSRSQ.
3465 NOTE: The data set WORK._NEXTCLUSRSQ has 2 observations and
      7 variables.
3466 NOTE: DATA statement used (Total process time):
3467       real time           0.00 seconds
3468       user cpu time       0.00 seconds
3469       system cpu time     0.01 seconds
3470       memory              159319.37k
3471       OS Memory           169600.00k
3472       Timestamp           07/01/2024 05:54:29 AM
3473       Step Count                        1  Switch Count  0
3474       Page Faults                       0
3475       Page Reclaims                     358
3476       Page Swaps                        0
```

```
3477       Voluntary Context Switches        0
3478       Involuntary Context Switches      0
3479       Block Input Operations            0
3480       Block Output Operations           264
3481
3482
3483
3484 NOTE: There were 5 observations read from the data set EMWS
     4.VARCLUS_OUTRSQUARE.
3485 NOTE: The data set EMWS4.VARCLUS_OUTRSQUARE has 5 observati
     ons and 7 variables.
3486 NOTE: PROCEDURE SORT used (Total process time):
3487       real time           0.01 seconds
3488       user cpu time       0.01 seconds
3489       system cpu time     0.00 seconds
3490       memory              159319.37k
3491       OS Memory           169600.00k
3492       Timestamp           07/01/2024 05:54:29 AM
3493       Step Count                        1  Switch Count  0
3494       Page Faults                       0
3495       Page Reclaims                     116
3496       Page Swaps                        0
3497       Voluntary Context Switches        38
3498       Involuntary Context Switches      0
3499       Block Input Operations            288
3500       Block Output Operations           264
3501
3502
3503
3504 NOTE: There were 2 observations read from the data set WORK
     ._NEXTCLUSRSQ.
3505 NOTE: The data set WORK._NEXTCLUSRSQ has 2 observations and
      7 variables.
3506 NOTE: PROCEDURE SORT used (Total process time):
3507       real time           0.00 seconds
3508       user cpu time       0.00 seconds
```

```
3509        system cpu time     0.00 seconds
3510        memory              159319.37k
3511        OS Memory           169600.00k
3512        Timestamp           07/01/2024 05:54:29 AM
3513        Step Count                        1  Switch Count  0
3514        Page Faults                       0
3515        Page Reclaims                     116
3516        Page Swaps                        0
3517        Voluntary Context Switches        0
3518        Involuntary Context Switches      0
3519        Block Input Operations            0
3520        Block Output Operations           264
3521
3522
3523
3524 NOTE: There were 5 observations read from the data set EMWS
     4.VARCLUS_OUTRSQUARE.
3525 NOTE: There were 2 observations read from the data set WORK
     ._NEXTCLUSRSQ.
3526 NOTE: The data set EMWS4.VARCLUS_OUTRSQUARE has 7 observati
     ons and 7 variables.
3527 NOTE: DATA statement used (Total process time):
3528        real time           0.01 seconds
3529        user cpu time       0.00 seconds
3530        system cpu time     0.00 seconds
3531        memory              159319.37k
3532        OS Memory           169600.00k
3533        Timestamp           07/01/2024 05:54:29 AM
3534        Step Count                        1  Switch Count  0
3535        Page Faults                       0
3536        Page Reclaims                     170
3537        Page Swaps                        0
3538        Voluntary Context Switches        38
3539        Involuntary Context Switches      0
3540        Block Input Operations            288
3541        Block Output Operations           264
```

```
3542
3543
3544
3545 NOTE: There were 7 observations read from the data set EMWS
     4.VARCLUS_OUTRSQUARE.
3546 NOTE: The data set EMWS4.VARCLUS_OUTRSQUARE has 7 observati
     ons and 9 variables.
3547 NOTE: DATA statement used (Total process time):
3548       real time           0.01 seconds
3549       user cpu time       0.00 seconds
3550       system cpu time     0.00 seconds
3551       memory              159319.37k
3552       OS Memory           169600.00k
3553       Timestamp           07/01/2024 05:54:29 AM
3554       Step Count                        1  Switch Count  0
3555       Page Faults                       0
3556       Page Reclaims                     471
3557       Page Swaps                        0
3558       Voluntary Context Switches        40
3559       Involuntary Context Switches      0
3560       Block Input Operations            288
3561       Block Output Operations           264
3562
3563
3564
3565 NOTE: Numeric values have been converted to character value
     s at the places given by: (Line):(Column).
3566       48667:90
3567 NOTE: Character values have been converted to numeric value
     s at the places given by: (Line):(Column).
3568       48667:84
3569 NOTE: There were 27 observations read from the data set EMW
     S4.VARCLUS_OUTSTAT.
3570 NOTE: The data set EMWS4.VARCLUS_OUTCLUSCORRPLOT has 2 obse
     rvations and 6 variables.
3571 NOTE: DATA statement used (Total process time):
```

```
3572        real time           0.00 seconds
3573        user cpu time       0.00 seconds
3574        system cpu time     0.00 seconds
3575        memory              159319.37k
3576        OS Memory           169600.00k
3577        Timestamp           07/01/2024 05:54:29 AM
3578        Step Count                        1  Switch Count  0
3579        Page Faults                       0
3580        Page Reclaims                     244
3581        Page Swaps                        0
3582        Voluntary Context Switches        20
3583        Involuntary Context Switches      0
3584        Block Input Operations            0
3585        Block Output Operations           264
3586
3587
3588
3589 NOTE: There were 2 observations read from the data set EMWS
     4.VARCLUS_OUTCLUSCORRPLOT.
3590 NOTE: The data set WORK.CORR_TMP has 2 observations and 6 v
     ariables.
3591 NOTE: DATA statement used (Total process time):
3592        real time           0.00 seconds
3593        user cpu time       0.00 seconds
3594        system cpu time     0.01 seconds
3595        memory              159319.37k
3596        OS Memory           169600.00k
3597        Timestamp           07/01/2024 05:54:29 AM
3598        Step Count                        1  Switch Count  0
3599        Page Faults                       0
3600        Page Reclaims                     128
3601        Page Swaps                        0
3602        Voluntary Context Switches        9
3603        Involuntary Context Switches      0
3604        Block Input Operations            288
3605        Block Output Operations           264
```

```
3606
3607
3608
3609 NOTE: There were 2 observations read from the data set EMWS
     4.VARCLUS_OUTCLUSCORRPLOT.
3610 NOTE: The data set EMWS4.VARCLUS_OUTCLUSCORRPLOT has 2 obse
     rvations and 3 variables.
3611 NOTE: DATA statement used (Total process time):
3612       real time           0.01 seconds
3613       user cpu time       0.00 seconds
3614       system cpu time     0.00 seconds
3615       memory              159319.37k
3616       OS Memory           169600.00k
3617       Timestamp           07/01/2024 05:54:29 AM
3618       Step Count                        1  Switch Count  0
3619       Page Faults                       0
3620       Page Reclaims                     357
3621       Page Swaps                        0
3622       Voluntary Context Switches        30
3623       Involuntary Context Switches      0
3624       Block Input Operations            0
3625       Block Output Operations           264
3626
3627
3628
3629 NOTE: Deleting WORK.CORR_TMP (memtype=DATA).
3630
3631 NOTE: PROCEDURE DATASETS used (Total process time):
3632       real time           0.00 seconds
3633       user cpu time       0.00 seconds
3634       system cpu time     0.00 seconds
3635       memory              159319.37k
3636       OS Memory           169600.00k
3637       Timestamp           07/01/2024 05:54:29 AM
3638       Step Count                        1  Switch Count  0
3639       Page Faults                       0
```

```
3640       Page Reclaims                     48
3641       Page Swaps                        0
3642       Voluntary Context Switches        0
3643       Involuntary Context Switches      0
3644       Block Input Operations            0
3645       Block Output Operations           8
3646
3647
3648
3649 NOTE: There were 2 observations read from the data set EMWS
     4.VARCLUS_OUTCLUSCORRPLOT.
3650 NOTE: The data set EMWS4.VARCLUS_OUTCLUSCORRPLOT has 2 obse
     rvations and 3 variables.
3651 NOTE: PROCEDURE SORT used (Total process time):
3652       real time           0.01 seconds
3653       user cpu time       0.00 seconds
3654       system cpu time     0.00 seconds
3655       memory              159319.37k
3656       OS Memory           169600.00k
3657       Timestamp           07/01/2024 05:54:29 AM
3658       Step Count                        1  Switch Count  0
3659       Page Faults                       0
3660       Page Reclaims                     115
3661       Page Swaps                        0
3662       Voluntary Context Switches        38
3663       Involuntary Context Switches      0
3664       Block Input Operations            288
3665       Block Output Operations           264
3666
3667
3668
3669 NOTE: There were 2 observations read from the data set EMWS
     4.VARCLUS_OUTCLUSCORRPLOT.
3670 NOTE: The data set EMWS4.VARCLUS_OUTCLUSCORRPLOT has 4 obse
     rvations and 3 variables.
3671 NOTE: PROCEDURE TRANSPOSE used (Total process time):
```

```
3672        real time           0.01 seconds
3673        user cpu time       0.00 seconds
3674        system cpu time     0.00 seconds
3675        memory              159319.37k
3676        OS Memory           169600.00k
3677        Timestamp           07/01/2024 05:54:29 AM
3678        Step Count                        1  Switch Count  0
3679        Page Faults                       0
3680        Page Reclaims                     228
3681        Page Swaps                        0
3682        Voluntary Context Switches        41
3683        Involuntary Context Switches      0
3684        Block Input Operations            288
3685        Block Output Operations           528
3686
3687
3688
3689 NOTE: There were 4 observations read from the data set EMWS
     4.VARCLUS_OUTCLUSCORRPLOT.
3690 NOTE: The data set EMWS4.VARCLUS_OUTCLUSCORRPLOT has 4 obse
     rvations and 3 variables.
3691 NOTE: DATA statement used (Total process time):
3692        real time           0.01 seconds
3693        user cpu time       0.00 seconds
3694        system cpu time     0.00 seconds
3695        memory              159319.37k
3696        OS Memory           169600.00k
3697        Timestamp           07/01/2024 05:54:29 AM
3698        Step Count                        1  Switch Count  0
3699        Page Faults                       0
3700        Page Reclaims                     470
3701        Page Swaps                        0
3702        Voluntary Context Switches        39
3703        Involuntary Context Switches      0
3704        Block Input Operations            288
3705        Block Output Operations           264
```

```
3706
3707
3708
3709 NOTE: There were 3 observations read from the data set EMWS
     4.VARCLUS_OUTSTAT.
3710       WHERE _type_ in ('MEAN', 'N', 'STD');
3711 NOTE: The data set WORK.VARCLUS_TMP has 3 observations and
     6 variables.
3712 NOTE: DATA statement used (Total process time):
3713       real time           0.00 seconds
3714       user cpu time       0.00 seconds
3715       system cpu time     0.00 seconds
3716       memory              159319.37k
3717       OS Memory           169600.00k
3718       Timestamp           07/01/2024 05:54:29 AM
3719       Step Count                        1  Switch Count  0
3720       Page Faults                       0
3721       Page Reclaims                     132
3722       Page Swaps                        0
3723       Voluntary Context Switches        4
3724       Involuntary Context Switches      0
3725       Block Input Operations            0
3726       Block Output Operations           264
3727
3728
3729
3730 NOTE: There were 3 observations read from the data set WORK
     .VARCLUS_TMP.
3731 NOTE: The data set EMWS4.VARCLUS_OUTSTATPLOT has 5 observat
     ions and 5 variables.
3732 NOTE: PROCEDURE TRANSPOSE used (Total process time):
3733       real time           0.00 seconds
3734       user cpu time       0.00 seconds
3735       system cpu time     0.00 seconds
3736       memory              159319.37k
3737       OS Memory           169600.00k
```

```
3738        Timestamp           07/01/2024 05:54:29 AM
3739        Step Count                        1  Switch Count  0
3740        Page Faults                       0
3741        Page Reclaims                     186
3742        Page Swaps                        0
3743        Voluntary Context Switches        15
3744        Involuntary Context Switches      0
3745        Block Input Operations            0
3746        Block Output Operations           528
3747
3748
3749
3750 NOTE: There were 5 observations read from the data set EMWS
     4.VARCLUS_OUTSTATPLOT.
3751 NOTE: The data set EMWS4.VARCLUS_OUTSTATPLOT has 5 observat
     ions and 6 variables.
3752 NOTE: DATA statement used (Total process time):
3753        real time           0.01 seconds
3754        user cpu time       0.00 seconds
3755        system cpu time     0.01 seconds
3756        memory              159319.37k
3757        OS Memory           169600.00k
3758        Timestamp           07/01/2024 05:54:29 AM
3759        Step Count                        1  Switch Count  0
3760        Page Faults                       0
3761        Page Reclaims                     470
3762        Page Swaps                        0
3763        Voluntary Context Switches        38
3764        Involuntary Context Switches      0
3765        Block Input Operations            288
3766        Block Output Operations           264
3767
3768
3769
3770 NOTE: There were 5 observations read from the data set EMWS
     4.VARCLUS_OUTSTATPLOT.
```

```
3771 NOTE: The data set EMWS4.VARCLUS_OUTSTATPLOT has 5 observat
     ions and 6 variables.
3772 NOTE: PROCEDURE SORT used (Total process time):
3773       real time           0.01 seconds
3774       user cpu time       0.01 seconds
3775       system cpu time     0.00 seconds
3776       memory              159319.37k
3777       OS Memory           169600.00k
3778       Timestamp           07/01/2024 05:54:29 AM
3779       Step Count                        1  Switch Count  0
3780       Page Faults                       0
3781       Page Reclaims                     116
3782       Page Swaps                        0
3783       Voluntary Context Switches        40
3784       Involuntary Context Switches      0
3785       Block Input Operations            288
3786       Block Output Operations           264
3787
3788
3789
3790 NOTE: Deleting WORK.VARCLUS_TMP (memtype=DATA).
3791
3792 NOTE: PROCEDURE DATASETS used (Total process time):
3793       real time           0.00 seconds
3794       user cpu time       0.00 seconds
3795       system cpu time     0.00 seconds
3796       memory              159319.37k
3797       OS Memory           169600.00k
3798       Timestamp           07/01/2024 05:54:29 AM
3799       Step Count                        1  Switch Count  0
3800       Page Faults                       0
3801       Page Reclaims                     49
3802       Page Swaps                        0
3803       Voluntary Context Switches        0
3804       Involuntary Context Switches      0
3805       Block Input Operations            0
```

```
3806       Block Output Operations           8
3807
3808
3809
3810 NOTE: There were 5 observations read from the data set EMWS
     4.VARCLUS_OUTSTAT.
3811       WHERE _type_='CORR';
3812 NOTE: The data set EMWS4.VARCLUS_OUTCORR has 5 observations
      and 6 variables.
3813 NOTE: DATA statement used (Total process time):
3814       real time           0.00 seconds
3815       user cpu time       0.00 seconds
3816       system cpu time     0.00 seconds
3817       memory              159319.37k
3818       OS Memory           169600.00k
3819       Timestamp           07/01/2024 05:54:29 AM
3820       Step Count                        1  Switch Count  0
3821       Page Faults                       0
3822       Page Reclaims                     129
3823       Page Swaps                        0
3824       Voluntary Context Switches        17
3825       Involuntary Context Switches      0
3826       Block Input Operations            0
3827       Block Output Operations           264
3828
3829
3830
3831 NOTE: There were 5 observations read from the data set EMWS
     4.VARCLUS_OUTCORR.
3832 NOTE: The data set EMWS4.VARCLUS_OUTCORR has 5 observations
      and 6 variables.
3833 NOTE: PROCEDURE SORT used (Total process time):
3834       real time           0.01 seconds
3835       user cpu time       0.00 seconds
3836       system cpu time     0.00 seconds
3837       memory              159319.37k
```

```
3838        OS Memory           169600.00k
3839        Timestamp           07/01/2024 05:54:29 AM
3840        Step Count                      1  Switch Count  0
3841        Page Faults                     0
3842        Page Reclaims                   117
3843        Page Swaps                      0
3844        Voluntary Context Switches      38
3845        Involuntary Context Switches    0
3846        Block Input Operations          288
3847        Block Output Operations         264
3848
3849
3850
3851 NOTE: There were 5 observations read from the data set EMWS
     4.VARCLUS_OUTCORR.
3852 NOTE: The data set EMWS4.VARCLUS_OUTCORRPLOT has 25 observa
     tions and 4 variables.
3853 NOTE: PROCEDURE TRANSPOSE used (Total process time):
3854        real time           0.00 seconds
3855        user cpu time       0.00 seconds
3856        system cpu time     0.00 seconds
3857        memory              159319.37k
3858        OS Memory           169600.00k
3859        Timestamp           07/01/2024 05:54:29 AM
3860        Step Count                      1  Switch Count  0
3861        Page Faults                     0
3862        Page Reclaims                   189
3863        Page Swaps                      0
3864        Voluntary Context Switches      26
3865        Involuntary Context Switches    0
3866        Block Input Operations          288
3867        Block Output Operations         528
3868
3869
3870
3871 NOTE: There were 25 observations read from the data set EMW
```

```
     S4.VARCLUS_OUTCORRPLOT.
3872 NOTE: The data set EMWS4.VARCLUS_OUTCORRPLOT has 25 observa
     tions and 5 variables.
3873 NOTE: DATA statement used (Total process time):
3874       real time           0.01 seconds
3875       user cpu time       0.00 seconds
3876       system cpu time     0.00 seconds
3877       memory              159319.37k
3878       OS Memory           169600.00k
3879       Timestamp           07/01/2024 05:54:29 AM
3880       Step Count                        1  Switch Count  0
3881       Page Faults                       0
3882       Page Reclaims                     126
3883       Page Swaps                        0
3884       Voluntary Context Switches        39
3885       Involuntary Context Switches      0
3886       Block Input Operations            288
3887       Block Output Operations           264
3888
3889
3890
3891 NOTE: There were 15 observations read from the data set EMW
     S4.VARCLUS_OUTCORRPLOT.
3892       WHERE _LABEL_ not = ' ';
3893 NOTE: The data set WORK.VARCLUS_MATCH has 15 observations a
     nd 2 variables.
3894 NOTE: DATA statement used (Total process time):
3895       real time           0.00 seconds
3896       user cpu time       0.00 seconds
3897       system cpu time     0.00 seconds
3898       memory              159319.37k
3899       OS Memory           169600.00k
3900       Timestamp           07/01/2024 05:54:29 AM
3901       Step Count                        1  Switch Count  0
3902       Page Faults                       0
3903       Page Reclaims                     132
```

```
3904        Page Swaps                          0
3905        Voluntary Context Switches          11
3906        Involuntary Context Switches        0
3907        Block Input Operations              288
3908        Block Output Operations             264
3909
3910
3911
3912  NOTE: Numeric values have been converted to character value
      s at the places given by: (Line):(Column).
3913        7:246
3914  NOTE: DATA statement used (Total process time):
3915        real time           0.00 seconds
3916        user cpu time       0.00 seconds
3917        system cpu time     0.00 seconds
3918        memory              159319.37k
3919        OS Memory           169600.00k
3920        Timestamp           07/01/2024 05:54:29 AM
3921        Step Count                          1  Switch Count  0
3922        Page Faults                         0
3923        Page Reclaims                       61
3924        Page Swaps                          0
3925        Voluntary Context Switches          0
3926        Involuntary Context Switches        0
3927        Block Input Operations              0
3928        Block Output Operations             0
3929
3930
3931
3932  NOTE: There were 15 observations read from the data set WOR
      K.VARCLUS_MATCH.
3933  NOTE: The data set WORK.VARCLUS_MATCH has 15 observations a
      nd 2 variables.
3934  NOTE: PROCEDURE SORT used (Total process time):
3935        real time           0.00 seconds
3936        user cpu time       0.00 seconds
```

```
3937        system cpu time     0.00 seconds
3938        memory              159319.37k
3939        OS Memory           169600.00k
3940        Timestamp           07/01/2024 05:54:29 AM
3941        Step Count                        1  Switch Count  0
3942        Page Faults                       0
3943        Page Reclaims                     115
3944        Page Swaps                        0
3945        Voluntary Context Switches        0
3946        Involuntary Context Switches      0
3947        Block Input Operations            0
3948        Block Output Operations           264
3949
3950
3951
3952 NOTE: There were 25 observations read from the data set EMW
      S4.VARCLUS_OUTCORRPLOT.
3953 NOTE: The data set EMWS4.VARCLUS_OUTCORRPLOT has 25 observa
      tions and 5 variables.
3954 NOTE: PROCEDURE SORT used (Total process time):
3955        real time           0.01 seconds
3956        user cpu time       0.00 seconds
3957        system cpu time     0.00 seconds
3958        memory              159319.37k
3959        OS Memory           169600.00k
3960        Timestamp           07/01/2024 05:54:29 AM
3961        Step Count                        1  Switch Count  0
3962        Page Faults                       0
3963        Page Reclaims                     118
3964        Page Swaps                        0
3965        Voluntary Context Switches        30
3966        Involuntary Context Switches      0
3967        Block Input Operations            0
3968        Block Output Operations           272
3969
3970
```

```
3971
3972 WARNING: Multiple lengths were specified for the BY variabl
     e _NAME_ by input data sets. This might cause unexpected re
     sults.
3973 NOTE: MERGE statement has more than one data set with repea
     ts of BY values.
3974 NOTE: There were 15 observations read from the data set WOR
     K.VARCLUS_MATCH.
3975 NOTE: There were 25 observations read from the data set EMW
     S4.VARCLUS_OUTCORRPLOT.
3976 NOTE: The data set EMWS4.VARCLUS_OUTCORRPLOT has 25 observa
     tions and 3 variables.
3977 NOTE: DATA statement used (Total process time):
3978       real time           0.01 seconds
3979       user cpu time       0.00 seconds
3980       system cpu time     0.00 seconds
3981       memory              159319.37k
3982       OS Memory           169600.00k
3983       Timestamp           07/01/2024 05:54:29 AM
3984       Step Count                        1  Switch Count  0
3985       Page Faults                       0
3986       Page Reclaims                     513
3987       Page Swaps                        0
3988       Voluntary Context Switches        37
3989       Involuntary Context Switches      0
3990       Block Input Operations            288
3991       Block Output Operations           264
3992
3993
3994
3995 NOTE: There were 25 observations read from the data set EMW
     S4.VARCLUS_OUTCORRPLOT.
3996 NOTE: The data set EMWS4.VARCLUS_OUTCORRPLOT has 25 observa
     tions and 3 variables.
3997 NOTE: PROCEDURE SORT used (Total process time):
3998       real time           0.01 seconds
```

```
3999        user cpu time       0.00 seconds
4000        system cpu time     0.01 seconds
4001        memory              159319.37k
4002        OS Memory           169600.00k
4003        Timestamp           07/01/2024 05:54:29 AM
4004        Step Count                        1  Switch Count  0
4005        Page Faults                       0
4006        Page Reclaims                     117
4007        Page Swaps                        0
4008        Voluntary Context Switches        40
4009        Involuntary Context Switches      0
4010        Block Input Operations            288
4011        Block Output Operations           264
4012
4013
4014
4015  NOTE: Deleting WORK.VARCLUS_MATCH (memtype=DATA).
4016
4017  NOTE: PROCEDURE DATASETS used (Total process time):
4018        real time           0.00 seconds
4019        user cpu time       0.00 seconds
4020        system cpu time     0.00 seconds
4021        memory              159319.37k
4022        OS Memory           169600.00k
4023        Timestamp           07/01/2024 05:54:29 AM
4024        Step Count                        1  Switch Count  0
4025        Page Faults                       0
4026        Page Reclaims                     49
4027        Page Swaps                        0
4028        Voluntary Context Switches        0
4029        Involuntary Context Switches      0
4030        Block Input Operations            0
4031        Block Output Operations           8
4032
4033
4034
```

```
4035 NOTE: There were 7 observations read from the data set EMWS
     4.VARCLUS_OUTRSQUARE.
4036 NOTE: The data set EMWS4.VARCLUS_OUTLINK has 7 observations
      and 9 variables.
4037 NOTE: DATA statement used (Total process time):
4038       real time           0.00 seconds
4039       user cpu time       0.01 seconds
4040       system cpu time     0.00 seconds
4041       memory              159319.37k
4042       OS Memory           169600.00k
4043       Timestamp           07/01/2024 05:54:29 AM
4044       Step Count                        1  Switch Count  0
4045       Page Faults                       0
4046       Page Reclaims                     246
4047       Page Swaps                        0
4048       Voluntary Context Switches        26
4049       Involuntary Context Switches      0
4050       Block Input Operations            288
4051       Block Output Operations           264
4052
4053
4054
4055 NOTE: There were 7 observations read from the data set EMWS
     4.VARCLUS_OUTRSQUARE.
4056 NOTE: The data set EMWS4.VARCLUS_OUTNODE has 7 observations
      and 3 variables.
4057 NOTE: DATA statement used (Total process time):
4058       real time           0.00 seconds
4059       user cpu time       0.00 seconds
4060       system cpu time     0.00 seconds
4061       memory              159319.37k
4062       OS Memory           169600.00k
4063       Timestamp           07/01/2024 05:54:29 AM
4064       Step Count                        1  Switch Count  0
4065       Page Faults                       0
4066       Page Reclaims                     357
```

```
4067        Page Swaps                          0
4068        Voluntary Context Switches          19
4069        Involuntary Context Switches        0
4070        Block Input Operations              0
4071        Block Output Operations             264
4072
4073
4074
4075  NOTE: There were 27 observations read from the data set EMW
      S4.VARCLUS_OUTSTAT.
4076  NOTE: The data set EMWS4.VARCLUS_OUTSTATSCORE has 4 observa
      tions and 6 variables.
4077  NOTE: DATA statement used (Total process time):
4078        real time           0.00 seconds
4079        user cpu time       0.00 seconds
4080        system cpu time     0.00 seconds
4081        memory              159319.37k
4082        OS Memory           169600.00k
4083        Timestamp           07/01/2024 05:54:29 AM
4084        Step Count                          1  Switch Count  0
4085        Page Faults                         0
4086        Page Reclaims                       127
4087        Page Swaps                          0
4088        Voluntary Context Switches          18
4089        Involuntary Context Switches        0
4090        Block Input Operations              0
4091        Block Output Operations             264
4092
4093
4094
4095  NOTE: The file _FILE_ is:
4096        Filename=/home/u63452984/case-study-s2192852/Workspac
      es/EMWS4/VarClus/EMFLOWSCORE.sas,
4097        Owner Name=u63452984,Group Name=oda,
4098        Access Permission=-rw-r--r--,
4099        Last Modified=07 January 2024 05:54:27,
```

```
4100       File Size (bytes)=613
4101
4102  NOTE: 12 records were written to the file _FILE_.
4103       The minimum record length was 1.
4104       The maximum record length was 86.
4105  NOTE: DATA statement used (Total process time):
4106       real time           0.00 seconds
4107       user cpu time       0.01 seconds
4108       system cpu time     0.00 seconds
4109       memory              159319.37k
4110       OS Memory           169600.00k
4111       Timestamp           07/01/2024 05:54:29 AM
4112       Step Count                        1  Switch Count  0
4113       Page Faults                       0
4114       Page Reclaims                     415
4115       Page Swaps                        0
4116       Voluntary Context Switches        15
4117       Involuntary Context Switches      0
4118       Block Input Operations            288
4119       Block Output Operations           8
4120
4121
4122
4123  NOTE: The file _OUT is:
4124       Filename=/home/u63452984/case-study-s2192852/Workspac
     es/EMWS4/VarClus/EMPUBLISHSCORE.sas,
4125       Owner Name=u63452984,Group Name=oda,
4126       Access Permission=-rw-r--r--,
4127       Last Modified=07 January 2024 05:54:29
4128
4129  NOTE: 39 records were written to the file _OUT.
4130       The minimum record length was 1.
4131       The maximum record length was 86.
4132  NOTE: DATA statement used (Total process time):
4133       real time           0.00 seconds
4134       user cpu time       0.00 seconds
```

```
4135        system cpu time     0.00 seconds
4136        memory              159319.37k
4137        OS Memory           169600.00k
4138        Timestamp           07/01/2024 05:54:29 AM
4139        Step Count                        1  Switch Count  0
4140        Page Faults                       0
4141        Page Reclaims                     29
4142        Page Swaps                        0
4143        Voluntary Context Switches        8
4144        Involuntary Context Switches      0
4145        Block Input Operations            0
4146        Block Output Operations           8
4147
4148
4149 NOTE: Fileref _IN has been deassigned.
4150 NOTE: Fileref _OUT has been deassigned.
4151 18171  *-------------------------------------------------
     ---------*;
4152 18172  * End TRAIN: VarClus;
4153 18173  *-------------------------------------------------
     ---------*;
4154 18174
4155
4156 18175  *-------------------------------------------------
     ---------*;
4157 18176  * Close any missing semi colons;
4158 18177  *-------------------------------------------------
     ---------*;
4159 18178  ;
4160 18179  ;
4161 18180  ;
4162 18181  ;
4163 18182  quit;
4164 18183  *-------------------------------------------------
     ---------*;
4165 18184  * Close any unbalanced quotes;
```

```
4166 18185  *-------------------------------------------------
     ---------*;
4167 18186  /*; *"; *'; */
4168 18187  ;
4169 18188  run;
4170 18189  quit;
4171 18190  /* Reset EM Options */
4172 18191  options formchar="|----|+|---+=|-/\<>*";
4173 18192  options nocenter ls=256 ps=10000;
4174 18193  goptions reset=all device=GIF NODISPLAY;
4175
4176 *-------------------------------------------------------
     --*
4177 * Score Log
4178 Date:                 07 January 2024
4179 Time:                 05:54:30
4180 *-------------------------------------------------------
     --*
4181 18295  %let EMEXCEPTIONSTRING=;
4182 18296  *-------------------------------------------------
     ---------*;
4183 18297  * SCORE: VarClus;
4184 18298  *-------------------------------------------------
     ---------*;
4185 18299  %let EM_ACTION = SCORE;
4186 18300  %let syscc = 0;
4187 18301
4188 18302  %macro main;
4189 18303
4190 18304     filename temp catalog 'sashelp.emexpl.variableclu
     stering_macros.source';
4191 18305     %include temp;
4192 18306     filename temp catalog 'sashelp.emexpl.variableclu
     stering_macros2.source';
4193 18307     %include temp;
4194 18308     filename temp;
```

```
4195 18309
4196 18310    %SetProperties;
4197 18311
4198 18312     %if %upcase(&EM_ACTION) = CREATE %then %do;
4199 18313         filename temp catalog 'sashelp.emexpl.variabl
     eclustering_create.source';
4200 18314         %include temp;
4201 18315         filename temp;
4202 18316         %create;
4203 18317     %end;
4204 18318     %else
4205 18319     %if %upcase(&EM_ACTION) = TRAIN %then %do;
4206 18320          filename temp catalog 'sashelp.emexpl.variab
     leclustering_train.source';
4207 18321             %include temp;
4208 18322             filename temp;
4209 18323             %train;
4210 18324     %end;
4211 18325     %else
4212 18326     %if %upcase(&EM_ACTION) = SCORE %then %do;
4213 18327             filename temp catalog 'sashelp.emexpl.var
     iableclustering_score.source';
4214 18328             %include temp;
4215 18329             filename temp;
4216 18330             %score;
4217 18331     %end;
4218 18332     %else
4219 18333     %if %upcase(&EM_ACTION) = REPORT %then %do;
4220 18334             filename temp catalog 'sashelp.emexpl.var
     iableclustering_report.source';
4221 18335             %include temp;
4222 18336             filename temp;
4223 18337             %report;
4224 18338     %end;
4225 18339     /*
4226 18340     %if %upcase(&EM_ACTION) = OPENTESTTABLE %then %do
```

```
     ;
4227 18341         %put 'OPENING TABLE';
4228 18342     %end;
4229 18343     %if %upcase(&EM_ACTION) = CLOSETESTTABLE %then %d
     o;
4230 18344         %put 'CLOSE TABLE';
4231 18345     %end;
4232 18346     */
4233 18347  %mend main;
4234 18348  %main;
4235 NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMEXPL.V
     ARIABLECLUSTERING_MACROS.SOURCE.
4236 18349 +
4237 18350 +/* Initialize property macro variables */
4238 18351 +%macro SetProperties;
4239 18352 +   %em_checkmacro(name=EM_PROPERTY_MAXCLUS,       gl
     obal=Y, value=DEFAULT);
4240 18353 +   %em_checkmacro(name=EM_PROPERTY_HIDEVARIABLE,  gl
     obal=Y, value=Y);
4241 18354 +   %em_checkmacro(name=EM_PROPERTY_PRINTOPTION,   gl
     obal=Y, value=SHORT);
4242 18355 +   %em_checkmacro(name=EM_PROPERTY_CLUSSOURCE,    gl
     obal=Y, value=CORR);
4243 18356 +   %em_checkmacro(name=EM_PROPERTY_CLUSCOMP,      gl
     obal=Y, value=PRINCIPAL);
4244 18357 +   %em_checkmacro(name=EM_PROPERTY_CLUSHIERACHY,
       global=Y, value=Y);
4245 18358 +   %em_checkmacro(name=EM_PROPERTY_INCLUDECLASSVAR,
          global=Y, value=N);
4246 18359 +   %em_checkmacro(name=EM_PROPERTY_EXPORTEDCOMP,
       global=Y, value=CLUSTERCOMP);
4247 18360 +   %em_checkmacro(name=EM_PROPERTY_MAXEIGEN,
      global=Y, value=DEFAULT);
4248 18361 +   %em_checkmacro(name=EM_PROPERTY_PROPORTION,
     global=Y, value=DEFAULT);
4249 18362 +   %em_checkmacro(name=EM_PROPERTY_PRINTOPTION,
```

```
      global=Y, value=SHORT);
4250 18363 +   %em_checkmacro(name=EM_PROPERTY_TWOSTAGECLUS,
       global=Y, value=AUTO);
4251 18364 +   %em_checkmacro(name=EM_PROPERTY_SUPPRESSSAMPWARN,
           global=Y, value=N);
4252 18365 +
4253 18366 +%mend SetProperties;
4254 18367 +
4255 18368 +%Macro MakeDummyVariables(indata=,
4256 18369 +                          outvar=,
4257 18370 +                          outdata=,
4258 18371 +                          fileref=,
4259 18372 +                          recreatecmeta=N, /* option
     al */
4260 18373 +                          incmeta=,     /* optional
     */
4261 18374 +                          outcmeta=,    /* optional
     */
4262 18375 +                          ndummyvars=_ndummyvars
4263 18376 +                          );
4264 18377 +    %global &ndummyvars;
4265 18378 +
4266 18379 +    proc dmdb batch data=&indata out=_dmdbdat dmdbca
     t=_dmdbcat classout=_classout;;
4267 18380 +       class
4268 18381 +       %EM_BINARY_INPUT %EM_NOMINAL_INPUT %EM_ORDINA
     L_INPUT
4269 18382 +       %EM_BINARY_REJECTED %EM_NOMINAL_REJECTED %EM_
     ORDINAL_REJECTED
4270 18383 +    ;
4271 18384 +    run;
4272 18385 +    %let &ndummyvars = 0;
4273 18386 +    data _null_;
4274 18387 +    %let dsid = %sysfunc(open(work._classout));
4275 18388 +      %let &ndummyvars = %sysfunc(attrn(&dsid, NOBS)
     );
```

```
4276  18389 +    %let dsid = %sysfunc(close(&dsid));
4277  18390 +    run;
4278  18391 +
4279  18392 +    proc dmzip data=_dmdbdat dmdbcat=_dmdbcat;
4280  18393 +       input
4281  18394 +       %EM_BINARY_INPUT %EM_NOMINAL_INPUT %EM_ORDINA
      L_INPUT
4282  18395 +       %EM_BINARY_REJECTED %EM_NOMINAL_REJECTED %EM_
      ORDINAL_REJECTED
4283  18396 +        / level=nominal stdize=no;
4284  18397 +       make outvar = &outvar;
4285  18398 +       score data = &indata out =&outdata;
4286  18399 +       code  file=  "&fileref";
4287  18400 +    run;
4288  18401 +    %if &recreatecmeta eq Y %then %do;
4289  18402 +    proc contents data =&outvar out=_tmpds(keep=NAME
       LABEL);
4290  18403 +    data _tmpds;
4291  18404 +        set _tmpds;
4292  18405 +           ROLE = 'INPUT';
4293  18406 +           LEVEL = 'INTERVAL';
4294  18407 +           CREATOR='DMZIP';
4295  18408 +           if NAME = '_TYPE_' then delete;
4296  18409 +    run;
4297  18410 +    data &outcmeta;
4298  18411 +         set &incmeta _tmpds;
4299  18412 +    run;
4300  18413 +    %end;
4301  18414 +    proc datasets lib=work nolist;
4302  18415 +      delete  _dmdbdat _dmdbcat _classout
4303  18416 +    %if &recreatecmeta eq Y %then %do;
4304  18417 +    _tmpds
4305  18418 +    %end;
4306  18419 +    ;
4307  18420 +    quit;
4308  18421 +%Mend MakeDummyVariables;
```

```
4309 18422 +
4310 18423 +/*--- Determine Optimal Number of Cluster ----
4311 18424 +%macro FindClusNum(statds=, groupds=, minvariation=)
     ;
4312 18425 +   %global optnclus;
4313 18426 +   data varclus_tmp(drop=_NAME_);
4314 18427 +      set &statDs;
4315 18428 +      where _type_ ='PROPOR';
4316 18429 +   run;
4317 18430 +   proc sort data=varclus_tmp;
4318 18431 +      by _NCL_;
4319 18432 +   run;
4320 18433 +   proc transpose data=varclus_tmp out=varclus_tmp;
4321 18434 +      by _NCL_;
4322 18435 +      var %EM_INTERVAL_INPUT
4323 18436 +      %if &EM_PROPERTY_INCLUDECLASSVAR eq Y %then %d
     o;
4324 18437 +      %let dsid = %sysfunc(open(&EM_USER_OUTDUMMY));
4325 18438 +      %let nvar = %sysfunc(attrn(&dsid, NVAR));
4326 18439 +          %do i = 2 %to &nvar;
4327 18440 +          %let varname = %sysfunc(varname(&dsid, &i)
     );
4328 18441 +          &varname
4329 18442 +          %end;
4330 18443 +      %end;
4331 18444 +   ;
4332 18445 +   run;
4333 18446 +
4334 18447 +   %if &minVariation eq %then %do;
4335 18448 +       %let minVariation = &EM_PROPERTY_MINVARIATION
     ;
4336 18449 +   %end;
4337 18450 +   %if ^(0<&minVariation<100) %then %do;
4338 18451 +       %let minVariation = 90;
4339 18452 +   %end;
4340 18453 +
```

```
4341  18454 +   data _null_;
4342  18455 +      set varclus_tmp end=eof;
4343  18456 +      by _NCL_;
4344  18457 +      retain flag 0;
4345  18458 +      if first._ncl_ then flag=0;
4346  18459 +      if .<col1 < &minVariation then flag=1;
4347  18460 +      if last._ncl_ and ^flag then do;
4348  18461 +         call symput('OPTNCL', _ncl_);
4349  18462 +         stop;
4350  18463 +      end;
4351  18464 +     if eof then call symput('OPTNCL', _ncl_);
4352  18465 +   run;
4353  18466 +
4354  18467 +   %let optnclus = &OPTNCL;
4355  18468 +
4356  18469 +   data varclus_tmp(drop=_NCL_ _NAME_);
4357  18470 +      set &statDs;
4358  18471 +      where _type_ in('RSQUARED' 'GROUP') and _NCL_=
      &OPTNCL;
4359  18472 +   run;
4360  18473 +   proc sort data=varclus_tmp;
4361  18474 +      by _TYPE_;
4362  18475 +   run;
4363  18476 +   proc transpose data=varclus_tmp out=varclus_tmp;
4364  18477 +      by _TYPE_;
4365  18478 +   run;
4366  18479 +   proc sort data=varclus_tmp;
4367  18480 +      by _name_ _type_;
4368  18481 +   run;
4369  18482 +
4370  18483 +  proc transpose data=varclus_tmp out=&groupds;
4371  18484 +      by _NAME_;
4372  18485 +   run;
4373  18486 +   proc sort data=&groupDs(rename=(col1=Cluster col2
      =Rsquare _NAME_=VARIABLE));
4374  18487 +      by Cluster descending Rsquare;
```

```
4375 18488 +      where Cluster ne 0;
4376 18489 +   run;
4377 18490 +   proc datasets lib=work nolist mt=(DATA VIEW);
4378 18491 +      delete varclus_tmp;
4379 18492 +   run;
4380 18493 +   quit;
4381 18494 +%mend findClusNum;
4382 18495 +*/
4383 18496 +
4384 18497 +%macro getNclusfromTrain(inoutstat=, nc=);
4385 18498 +%global &nc;
4386 18499 +data _null_;
4387 18500 +    set &inoutstat end=eof;
4388 18501 +    if eof then do;
4389 18502 +    call symput("&nc", _ncl_);
4390 18503 +    end;
4391 18504 +run;
4392 18505 +%mend  getNclusfromTrain;
4393 18506 +
4394 18507 +%macro MakeDeltaCode(groupds=, outstatscore=, deltac
     odefile=);
4395 18508 +
4396 18509 +     *--- Build Code to Modify Metadata ---*;
4397 18510 +     filename X "&deltacodefile";
4398 18511 +     data _null_;
4399 18512 +        FILE X;
4400 18513 +        set &groupds end=eof;
4401 18514 +        /*by Cluster;*/
4402 18515 +         if _N_=1 then do;
4403 18516 +           %if &EM_PROPERTY_INCLUDECLASSVAR eq Y %th
     en %do;
4404 18517 +            put "if upcase(strip(ROLE)) ='INPUT' and
      upcase(strip(LEVEL)) ^='INTERVAL' then ROLE ='REJECTED' ;"
     ;
4405 18518 +           %end;
4406 18519 +           put "if upcase(strip(ROLE))='INPUT' and u
```

```
     pcase(strip(LEVEL))='INTERVAL' then do;";
4407 18520 +           put "if upcase(strip(NAME)) in (";
4408 18521 +        end;
4409 18522 +        if Strip(upcase(Selected)) eq 'YES' then do;
4410 18523 +           string = '"'!!trim(left(VARIABLE))!!'"';
4411 18524 +           put string;
4412 18525 +        end;
4413 18526 +        if eof then do;
4414 18527 +           put ') then ROLE="INPUT";';
4415 18528 +           put 'else ROLE="REJECTED";';
4416 18529 +           put 'end;';
4417 18530 +
4418 18531 +           %if %upcase(&EM_PROPERTY_HIDEVARIABLE) eq
      Y %then %do;
4419 18532 +             put 'if upcase(strip(ROLE)) = "REJECTED
     " then delete ;';
4420 18533 +           %end;
4421 18534 +        end;
4422 18535 +     run;
4423 18536 +     quit;
4424 18537 +
4425 18538 +     filename X;
4426 18539 +
4427 18540 +     quit;
4428 18541 +%mend MakeDeltaCode;
4429 18542 +
4430 18543 +%macro MakeVarClusCorrData(statds=, corrds=, corrplo
     tds= );
4431 18544 +    %if ^%sysfunc(exist(&statds)) %then %do;
4432 18545 +         %goto doendc;
4433 18546 +    %end;
4434 18547 +
4435 18548 +    data &corrds(drop=_TYPE_ _NCL_) ;
4436 18549 +       set &statds;
4437 18550 +       where _type_ eq 'CORR' ;
4438 18551 +    run ;
```

```
4439  18552 +    proc sort data=&corrds;
4440  18553 +       by _NAME_ ;
4441  18554 +    run ;
4442  18555 +    proc transpose data=&corrds out=&corrplotds name
     =_TMP_;
4443  18556 +      BY _NAME_ ;
4444  18557 +    run ;
4445  18558 +    data &corrplotds;
4446  18559 +       length _Y_ $100;
4447  18560 +       set &corrplotDs;
4448  18561 +       if _LABEL_ ne '' then _Y_=_LABEL_ ; else _Y_=
     _TMP_ ;
4449  18562 +    run ;
4450  18563 +    data varclus_match(rename=(_TMP_= _NAME_ _LABEL_
     =_X_));
4451  18564 +       set &corrplotds;
4452  18565 +       where _LABEL_ ne '' ;
4453  18566 +       keep _TMP_ _LABEL_ ;
4454  18567 +    run ;
4455  18568 +    data _null_;
4456  18569 +       nobs=0;
4457  18570 +       dsid = open('varclus_match');
4458  18571 +       if dsid then do;
4459  18572 +          nobs = attrn(dsid, 'NOBS');
4460  18573 +          dsid = close(dsid);
4461  18574 +       end;
4462  18575 +       call symput ('CORR_NOBS', nobs);
4463  18576 +    run;
4464  18577 +    %if &corr_nobs %then %do;
4465  18578 +        proc sort data=varclus_match;
4466  18579 +           by _name_;
4467  18580 +        run ;
4468  18581 +        proc sort data=&corrplotds;
4469  18582 +           by _name_;
4470  18583 +        run ;
4471  18584 +        data &corrplotds(keep= _X_ _Y_ col1 rename=(
```

```
     col1=Correlation)) ;
4472 18585 +           merge varclus_match &corrplotds;
4473 18586 +           by _NAME_ ;
4474 18587 +           if _X_ eq '' then _X_=_NAME_ ;
4475 18588 +           label _X_ = "%sysfunc(sasmsg(sashelp.dmin
     e, rpt_varclus_label_variable, noquote))";
4476 18589 +           label _Y_ = "%sysfunc(sasmsg(sashelp.dmin
     e, rpt_varclus_label_variable, noquote))";
4477 18590 +           label col1 = "%sysfunc(sasmsg(sashelp.dmi
     ne, rpt_correlation_vlabel, noquote))";
4478 18591 +
4479 18592 +       run ;
4480 18593 +    %end;
4481 18594 +    %else %do;
4482 18595 +        proc sort data=&corrplotds;
4483 18596 +           by _name_;
4484 18597 +        run ;
4485 18598 +        data &corrplotds(keep= _NAME_ _Y_ col1 renam
     e=(_NAME_=_X_ col1=Correlation)) ;
4486 18599 +           set &corrplotds;
4487 18600 +           label _NAME_ = "%sysfunc(sasmsg(sashelp.d
     mine, rpt_varclus_label_variable, noquote))" ;
4488 18601 +           label _Y_ = "%sysfunc(sasmsg(sashelp.dmin
     e, rpt_varclus_label_variable, noquote))" ;
4489 18602 +           label col1 = "%sysfunc(sasmsg(sashelp.dmi
     ne, rpt_correlation_vlabel, noquote))";
4490 18603 +
4491 18604 +        run ;
4492 18605 +    %end;
4493 18606 +    proc sort data=&corrplotds;
4494 18607 +       by _X_ _Y_;
4495 18608 +    run ;
4496 18609 +    proc datasets lib=work nolist mt=(DATA VIEW);
4497 18610 +       delete varclus_match;
4498 18611 +    run;
4499 18612 +    quit;
```

```
4500 18613 +
4501 18614 +%doendc:
4502 18615 +
4503 18616 +%mend MakeVarClusCorrData;
4504 18617 +
4505 18618 +%macro MakeStatPlotData(statds= , outstatplotds=);
4506 18619 +   %if %sysfunc(exist(&statds)) %then %do;
4507 18620 +
4508 18621 +       data varclus_tmp(drop=_NAME_ _NCL_) ;
4509 18622 +          set &statDs;
4510 18623 +          where _type_ in('MEAN', 'STD', 'N');
4511 18624 +       run ;
4512 18625 +       proc transpose data=varclus_tmp out=&outstatp
     lotds;
4513 18626 +          id _TYPE_ ;
4514 18627 +       run ;
4515 18628 +       data &outstatplotds;
4516 18629 +          set &outstatplotds(obs=1000);
4517 18630 +          label _name_= "%sysfunc(sasmsg(sashelp.dmi
     ne, rpt_varclus_label_name, noquote))";
4518 18631 +          label _label_="%sysfunc(sasmsg(sashelp.dmi
     ne, rpt_varclus_label_label, noquote))" ;
4519 18632 +          if MEAN ne 0 then SCALEDSTD= STD / MEAN ;
4520 18633 +          else SCALEDSTD= STD ;
4521 18634 +          label SCALEDSTD = "%sysfunc(sasmsg(sashelp
     .dmine, rpt_varclus_label_scaledstd, noquote))";
4522 18635 +       run ;
4523 18636 +       proc sort data=&outstatplotds;
4524 18637 +          by descending SCALEDSTD ;
4525 18638 +       run ;
4526 18639 +       proc datasets lib=work nolist mt=(DATA VIEW);
4527 18640 +          delete varclus_tmp;
4528 18641 +       run;
4529 18642 +       quit;
4530 18643 +    %end;
4531 18644 +
```

```
4532 18645 +%mend MakeStatPlotData;
4533 18646 +
4534 18647 +
4535 18648 +%macro CreateScoreCode(indata=, ncluscomp=, fileref=
     );
4536 18649 +     %EM_GETNAME(KEY=OUTSTATSCORE, type=DATA);
4537 18650 +    data &EM_USER_OUTSTATSCORE;
4538 18651 +          set &indata;
4539 18652 +          if (_TYPE_ in ('SCORE' 'MEAN' 'STD') and _
     NCL_ = &ncluscomp ) or (_TYPE_ in ('MEAN' 'STD'));
4540 18653 +          if _TYPE_ = 'MEAN' then _NAME_='MEAN';
4541 18654 +          if _TYPE_ = 'STD' then _NAME_='STD';
4542 18655 +          DROP _TYPE_ _NCL_;
4543 18656 +     run;
4544 18657 +
4545 18658 +     filename _file_  "&fileRef";
4546 18659 +
4547 18660 +     data _null_;
4548 18661 +        FILE _file_ MOD;
4549 18662 +        put ' ';
4550 18663 +        put '/*-----------------------------------
     ------------*/';
4551 18664 +        put '/* ' "%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_score_title_begin , noquote))" '*/';
4552 18665 +        put '/*-----------------------------------
     ------------*/';
4553 18666 +        put ' ';
4554 18667 +        %let dsid = %sysfunc(open(&EM_USER_OUTSTATSC
     ORE));
4555 18668 +
4556 18669 +        %let nvar = %sysfunc(attrn(&dsid, NVAR));
4557 18670 +        %let vn_name =%sysfunc(varnum(&dsid, _NAME_)
     );
4558 18671 +
4559 18672 +        %let k = 1;
4560 18673 +        %do %while(^%sysfunc(fetch(&dsid)));
```

```
4561  18674 +                %let _name = %sysfunc(getvarc(&dsid,
       &vn_name));
4562  18675 +                %if &k > 2 %then %do;
4563  18676 +                 %let cn = %eval(&k-2);
4564  18677 +                 put "&_name = 0 ; /*---" "%sysfunc(
      sasmsg(sashelp.dmine, rpt_varclus_score_cluscompnum, noquot
      e, &cn))"  "------ */";
4565  18678 +                %end;
4566  18679 +                %let k = %eval(&k+1);
4567  18680 +        %end;
4568  18681 +
4569  18682 +        %let rc = %sysfunc(rewind(&dsid));
4570  18683 +
4571  18684 +        %do i= 2 %to &nvar;
4572  18685 +            %let _varname =  %sysfunc(varname(&dsid,
       &i));
4573  18686 +            %do %while(^%sysfunc(fetch(&dsid)));
4574  18687 +                %let _name = %sysfunc(getvarc(&dsid,
       &vn_name));
4575  18688 +                %if &_name = MEAN %then
4576  18689 +                %let _mean = %sysfunc(getvarn(&dsid,
       &i));
4577  18690 +                %else %if &_name = STD %then
4578  18691 +                %let _std = %sysfunc(getvarn(&dsid,
      &i));
4579  18692 +                %else %do;
4580  18693 +                      %let coeff =  %sysfunc(getvarn
      (&dsid, &i));
4581  18694 +                      %let abscoeff = %sysfunc(abs(&
      coeff));
4582  18695 +                         %if &abscoeff >  0 %then %
      do;
4583  18696 +                          put "&_name = &_name+&coeff *
       (&_varname - &_mean)/&_std;";
4584  18697 +                           %end;
4585  18698 +                 %end;
```

```
4586 18699 +             %end;
4587 18700 +             %let rc = %sysfunc(rewind(&dsid));
4588 18701 +          %end;
4589 18702 +
4590 18703 +         %let dsid= %sysfunc(close(&dsid));
4591 18704 +       run;
4592 18705 +%mend CreateScoreCode;
4593 18706 +
4594 18707 +
4595 18708 +
4596 18709 +/*-------------------------------------------------
     --------
4597 18710 +    Instead of using %MakeRSquareData,
4598 18711 +    %MakeVarClusResultTable at macro2.source is used
4599 18712 + +-------------------------------------------------
     --------*/
4600 18713 +
4601 18714 +
4602 18715 +%macro MakeRSquareData(indata=, inClusRSquare=, outd
     ata=, ncluster=);
4603 18716 +
4604 18717 +/* modifying  from ods rsquare = data */
4605 18718 +
4606 18719 +data &outdata(drop= ControlVar  NumberOfClusters Cur
     rentCluster);
4607 18720 +    Length Cluster $16;
4608 18721 +    length Variable $32;
4609 18722 +    Length VariableLabel $64;
4610 18723 +    set &indata; retain CurrentCluster;
4611 18724 +    if NumberOfClusters ^= &ncluster then delete;
4612 18725 +    if strip(Cluster) eq '' then Cluster = CurrentCl
     uster;
4613 18726 +    CurrentCluster = Cluster;
4614 18727 +   run;
4615 18728 +proc sort data =&outdata ;
4616 18729 +     by Cluster RsquareRatio;
```

```
4617  18730 +run;
4618  18731 +data _tmprsq(drop=index);
4619  18732 +     set &outdata; by Cluster;
4620  18733 +     if first.Cluster then do;
4621  18734 +     index = strip(scan(Cluster,2 ));
4622  18735 +     Variable = "Clus"||index;
4623  18736 +     VariableLabel = "Cluster Component "||index;
4624  18737 +     OwnCluster = 1;
4625  18738 +     NextClosest = .;
4626  18739 +     RsquareRatio = 0;
4627  18740 +     output;
4628  18741 +     end;
4629  18742 +run;
4630  18743 +
4631  18744 +proc sort data = _tmprsq ;
4632  18745 +     by Cluster RsquareRatio;
4633  18746 +run;
4634  18747 +data &outdata;
4635  18748 +     set &outdata _tmprsq;
4636  18749 +by Cluster;
4637  18750 +run;
4638  18751 +
4639  18752 +
4640  18753 +/* Just create the Selected variable with all YES */
4641  18754 +
4642  18755 +data &outdata;
4643  18756 +    set &outdata; by cluster;
4644  18757 +    length Selected $8;
4645  18758 +    Selected = 'YES';
4646  18759 +    label  OwnCluster = 'R-Sqaure with Cluster Compo
     nent';
4647  18760 +    label  NextClosest = 'R-Sqaure with Next Cluster
      Component';
4648  18761 +    rename OwnCluster = RSqWithClusterComp;
4649  18762 +    rename NextClosest = RSqWithNextClusComp;
4650  18763 +run;
```

```
4651 18764 +
4652 18765 +
4653 18766 +/* Selected = Y/N will be done %score section -----
4654 18767 +
4655 18768 +%if &EM_PROPERTY_EXPORTEDCOMP ne CLUSTERCOMP %then %
     do;
4656 18769 +data &outdata;
4657 18770 +    set &outdata; by cluster;
4658 18771 +    length Selected $8;
4659 18772 +    if first.Cluster then Selected = 'YES';
4660 18773 +    else Selected = 'NO';
4661 18774 +    label  OwnCluster = 'R-Sqaure with Cluster Compo
     nent';
4662 18775 +    label  NextClosest = 'R-Sqaure with Next Cluster
      Component';
4663 18776 +    rename OwnCluster = RSqWithClusterComp;
4664 18777 +    rename NextClosest = RSqWithNextClusComp;
4665 18778 +run;
4666 18779 +%end;
4667 18780 +%else %do;
4668 18781 +data &outdata;
4669 18782 +    set &outdata; by cluster;
4670 18783 +    if last.Cluster then Selected = 'YES';
4671 18784 +    else Selected = 'NO';
4672 18785 +    label  OwnCluster = 'R-Sqaure with Cluster Compo
     nent';
4673 18786 +    label  NextClosest = 'R-Sqaure with Next Cluster
      Component';
4674 18787 +    rename OwnCluster = RSqWithClusterComp;
4675 18788 +    rename NextClosest = RSqWithNextClusComp;
4676 18789 +run;
4677 18790 +%end;
4678 18791 +-----------------------------------------------
     -----------*/
4679 18792 +
4680 18793 +%if %sysfunc(exist(&inClusRSquare)) %then %do;
```

```
4681 18794 +/* to calculate NextClosestClusRsq */
4682 18795 +proc transpose data = &inClusRSquare  out=_clusRsq;
4683 18796 +      by cluster;
4684 18797 +      run;
4685 18798 +data _clusRsq;
4686 18799 +     set _clusRsq;
4687 18800 +     if strip(upcase(Cluster)) eq strip(upcase(_NAME
     _)) then delete;
4688 18801 +run;
4689 18802 +
4690 18803 +proc sort data=_clusRsq;
4691 18804 +    by cluster col1;
4692 18805 +    run;
4693 18806 +data _clusRsq(drop=_NAME_ _LABEL_);
4694 18807 +     set _clusRsq; by cluster;
4695 18808 +     if last.Cluster then output;
4696 18809 +     label  COL1 = 'R-Sqaure with Next Cluster Compo
     nent';
4697 18810 +     rename COL1 = RSqWithNextClusComp;
4698 18811 +     rename Cluster = Variable;
4699 18812 +     label  Cluster = "Variable";
4700 18813 +run;
4701 18814 +
4702 18815 +proc sort data =&outdata;
4703 18816 +     by Variable;
4704 18817 +run;
4705 18818 +data &outdata;
4706 18819 +    merge &outdata _clusRsq;
4707 18820 +    by Variable;
4708 18821 +run;
4709 18822 +proc sort data =&outdata;
4710 18823 +by Cluster  RsquareRatio;
4711 18824 +run;
4712 18825 +quit;
4713 18826 +%end;
4714 18827 +
```

```
4715  18828 +proc datasets lib = work nolist;
4716  18829 +     delete _tmprsq _clusRsq;
4717  18830 +     run;
4718  18831 +quit;
4719  18832 +
4720  18833 +%mend MakeRSquareData;
4721  18834 +
4722  18835 +
4723  18836 +/*-------------------------------------------------
      ------------------------*/
4724  18837 +
4725  18838 +
4726  18839 +
4727  18840 +%macro ModifyCorr(indata=,
4728  18841 +                  outdata=,
4729  18842 +                  rsquare = Y
4730  18843 +                  );
4731  18844 +    data corr_tmp;
4732  18845 +        set &indata;
4733  18846 +    run;
4734  18847 +     proc sql;
4735  18848 +           update &indata
4736  18849 +           set
4737  18850 +     %let dsid = %sysfunc(open(work.corr_tmp));
4738  18851 +     %let nvar = %sysfunc(attrn(&dsid, NVAR));
4739  18852 +          %do i = 4 %to &nvar;
4740  18853 +          %let _name = %sysfunc(varname(&dsid, &i));
4741  18854 +             %if &rsquare eq Y %then %let _name_md =
       &_name.**2;
4742  18855 +            %else  %let _name_md = &_name;
4743  18856 +            %if &i < &nvar %then %do;
4744  18857 +               &_name = 1- &_name_md ,
4745  18858 +            %end;
4746  18859 +            %else %do;
4747  18860 +               &_name = &_name_md where _TYPE_ conta
      ins 'CORR' ;
```

```
4748 18861 +            %end;
4749 18862 +          %end;
4750 18863 +      %let dsid= %sysfunc(close(&dsid));
4751 18864 +
4752 18865 +      select * from &indata;
4753 18866 +      run;
4754 18867 +      proc datasets lib = work nolist;
4755 18868 +           delete corr_tmp;
4756 18869 +      run;
4757 18870 +      quit;
4758 18871 +
4759 18872 + %mend ModifyCorr;
4760 18873 +
4761 18874 + %macro MakeClusStructCorrData(indata=,outdata=, ncl
     uster=, Rsquare=N);
4762 18875 +    data &outdata(drop= _NCL_  _TYPE_);
4763 18876 +       set &indata;
4764 18877 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
     E_) eq 'STRUCTUR') then delete;
4765 18878 +       rename _NAME_ = Cluster;
4766 18879 +       label _NAME_ = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_clustername,  noquote))";
4767 18880 +    run;
4768 18881 +    %if &RSquare eq Y %then %do;
4769 18882 +     data corr_tmp;
4770 18883 +        set &outdata;
4771 18884 +     run;
4772 18885 +
4773 18886 +     data &outdata(drop=i);
4774 18887 +          set &outdata;
4775 18888 +          %let dsid = %sysfunc(open(work.corr_tmp));
4776 18889 +          %let nvar = %sysfunc(attrn(&dsid, NVAR));
4777 18890 +          %do i = 2 %to &nvar;
4778 18891 +            %let _name = %sysfunc(varname(&dsid, &i)
     );
4779 18892 +            %let _name_md = &_name.**2;
```

```
4780 18893 +                &_name = &_name_md;
4781 18894 +           %end;
4782 18895 +      %let dsid= %sysfunc(close(&dsid));
4783 18896 +      run;
4784 18897 +      proc datasets lib = work nolist;
4785 18898 +           delete corr_tmp;
4786 18899 +      run;
4787 18900 +
4788 18901 +    %end;
4789 18902 +     quit;
4790 18903 +%mend MakeClusStructCorrData;
4791 18904 +
4792 18905 +%macro MakeInterClusCorrData(indata=, outdata=, nclu
     ster=, RSquare=N, makeplotds=N, plotds=);
4793 18906 +    data &outdata(drop= _NCL_  _TYPE_);
4794 18907 +       set &indata;
4795 18908 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
     E_) eq 'CCORR') then delete;
4796 18909 +       rename _NAME_ = Cluster;
4797 18910 +       label _NAME_ = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_clustername,  noquote))";
4798 18911 +    run;
4799 18912 +    data corr_tmp;
4800 18913 +        set &outdata;
4801 18914 +    run;
4802 18915 +
4803 18916 +    %let dsid = %sysfunc(open(work.corr_tmp));
4804 18917 +    %let nclus2= %eval(&ncluster+1);
4805 18918 +    data &outdata;
4806 18919 +         set &outdata;
4807 18920 +          %do i = 2 %to &nclus2;
4808 18921 +          %let i_1 = %eval(&i-1);
4809 18922 +            %let _name = %sysfunc(varname(&dsid, &i)
     );
4810 18923 +            %let _newName = Clus&i_1;
4811 18924 +                rename &_name = &_newName; ;
```

```
4812 18925 +               *label &_name ="Cluster &i_1";
4813 18926 +               label &_name = "%sysfunc(sasmsg(sash
     elp.dmine, rpt_varclus_label_clusternum, noquote,  &i_1))";
4814 18927 +          %end;
4815 18928 +          keep Cluster
4816 18929 +          %do i = 2 %to &nclus2;
4817 18930 +               %let _name = %sysfunc(varname(&dsid,
      &i));
4818 18931 +               &_name
4819 18932 +          %end;
4820 18933 +          ;
4821 18934 +    %let dsid= %sysfunc(close(&dsid));
4822 18935 +     run;
4823 18936 +     quit;
4824 18937 +
4825 18938 +    %if &RSquare eq Y %then %do;
4826 18939 +
4827 18940 +       data corr_tmp;
4828 18941 +        set &outdata;
4829 18942 +     run;
4830 18943 +
4831 18944 +     data &outdata(drop=i);
4832 18945 +          set &outdata;
4833 18946 +          %let dsid = %sysfunc(open(work.corr_tmp));
4834 18947 +          %let nvar = %sysfunc(attrn(&dsid, NVAR));
4835 18948 +          %do i = 2 %to &nvar;
4836 18949 +            %let _name = %sysfunc(varname(&dsid, &i)
     );
4837 18950 +            %let _name_md = &_name.**2;
4838 18951 +                &_name = &_name_md;
4839 18952 +          %end;
4840 18953 +      %let dsid= %sysfunc(close(&dsid));
4841 18954 +      run;
4842 18955 +    %end;
4843 18956 +
4844 18957 +    %if &makeplotds eq Y %then %do;
```

```
4845 18958 +     proc transpose data = &outdata
4846 18959 +          out=&plotds(drop=_LABEL_ rename=(_NAME_ =
     Y Cluster=X Col1= Correlation));
4847 18960 +          by cluster;
4848 18961 +     run;
4849 18962 +     data &plotds;
4850 18963 +          set  &plotds;
4851 18964 +          label x="%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_label_cluster, noquote))";
4852 18965 +          label Y="%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_label_cluster, noquote))";
4853 18966 +     run;
4854 18967 +     %end;
4855 18968 +     proc datasets lib = work nolist;
4856 18969 +           delete corr_tmp;
4857 18970 +     run;
4858 18971 +     quit;
4859 18972 +%mend MakeInterClusCorrData;
4860 18973 +
4861 18974 +
4862 18975 +%macro MakeClusConstellData(indata=, outlink=, outno
     de=);
4863 18976 +
4864 18977 +data &outlink(drop = Selected);
4865 18978 +     set &indata;
4866 18979 +     LINKID = _N_;
4867 18980 +     label LINKID = "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_linkid,  noquote))";
4868 18981 +     if strip(upcase(Cluster)) eq strip(upcase(Varia
     ble)) then Variable = ClosestCluster;
4869 18982 +run;
4870 18983 +data &outnode(keep=NODEID TYPE LABEL);
4871 18984 +    set &indata;
4872 18985 +    length TYPE $16;
4873 18986 +    rename Variable = NODEID;
4874 18987 +    label  Variable= "%sysfunc(sasmsg(sashelp.dmine,
```

```
      rpt_varclus_label_nodeidvar, noquote))";
4875 18988 +    if strip(upcase(Cluster)) eq strip(upcase(Variab
     le))
4876 18989 +    then  TYPE = "CLUSTER";
4877 18990 +    else  TYPE="VARIABLE";
4878 18991 +    label TYPE = "%sysfunc(sasmsg(sashelp.dmine, rpt
     _varclus_label_nodetype, noquote))";
4879 18992 +run;
4880 18993 +quit;
4881 18994 +%mend MakeClusConstellData;
4882 18995 +
4883 18996 +
4884 18997 +
4885 18998 +%macro MakeClusConstellData(indata=, outlink=, outno
     de=);
4886 18999 +
4887 19000 +data &outlink(drop = Selected);
4888 19001 +     set &indata;
4889 19002 +     LINKID = _N_;
4890 19003 +     label LINKID = "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_linkid,  noquote))";
4891 19004 +     if strip(upcase(Cluster)) eq strip(upcase(Varia
     ble)) then Variable = ClosestCluster;
4892 19005 +run;
4893 19006 +data &outnode(keep=NODEID TYPE LABEL);
4894 19007 +    set &indata;
4895 19008 +    length TYPE $16;
4896 19009 +    rename Variable = NODEID;
4897 19010 +    label  Variable= "%sysfunc(sasmsg(sashelp.dmine,
      rpt_varclus_label_nodeidvar, noquote))";
4898 19011 +    if strip(upcase(Cluster)) eq strip(upcase(Variab
     le))
4899 19012 +    then  TYPE = "CLUSTER";
4900 19013 +    else  TYPE="VARIABLE";
4901 19014 +    label TYPE = "%sysfunc(sasmsg(sashelp.dmine, rpt
     _varclus_label_nodetype, noquote))";
```

```
4902  19015 +run;
4903  19016 +quit;
4904  19017 +%mend MakeClusConstellData;
4905  19018 +
4906  19019 +
4907  19020 +/*--- This will work only when inds is not a view da
      ta -------
4908  19021 +
4909  19022 +%macro getNVarNObs(inds=, nvar=, nobs=);
4910  19023 +
4911  19024 +    %global &nvar;
4912  19025 +    %global &nobs;
4913  19026 +    data _null_;
4914  19027 +         dsid = open("&inds");
4915  19028 +         nv = attrn(dsid, 'NVAR');
4916  19029 +         no = attrn(dsid, 'NOBS');
4917  19030 +         dsid = close(dsid);
4918  19031 +         call symput("&nvar", nv);
4919  19032 +         call symput("&nobs", no);
4920  19033 +    run;
4921  19034 +    quit;
4922  19035 +%mend  getNVarNObs;
4923  19036 +
4924  19037 ++-------------------------------------------------
      ------------*/
4925  19038 +
4926  19039 +
4927  19040 +%macro getNVar(inds=, nvar=);
4928  19041 +    %global &nvar;
4929  19042 +    data _null_;
4930  19043 +         dsid = open("&inds");
4931  19044 +         nv = attrn(dsid, 'NVAR');
4932  19045 +         dsid = close(dsid);
4933  19046 +         call symput("&nvar", nv);
4934  19047 +    run;
4935  19048 +    quit;
```

```
4936  19049 +%mend  getNVar;
4937  19050 +
4938  19051 +
4939  19052 +
4940  19053 +%macro getNObs(inds=, nobs=);
4941  19054 +    %global &nobs;
4942  19055 +    data _null_;
4943  19056 +        set &inds end=eof;
4944  19057 +        if eof then call symput("&nobs", _N_);
4945  19058 +    run;
4946  19059 +    quit;
4947  19060 +%mend  getNObs;
4948  19061 +
4949  19062 +%Macro CreateVarclusMeta(trainnum=);
4950  19063 +   %EM_GETNAME(KEY=VARCLUSMETA, TYPE=DATA);
4951  19064 +    data &EM_USER_VARCLUSMETA;
4952  19065 +         length TrainNum 8.;
4953  19066 +         length NewTrain $8;
4954  19067 +         length NGCluster 8.;
4955  19068 +         length ExportedComp $16;
4956  19069 +         length HideVariable $8;
4957  19070 +         TrainNum = &trainnum;
4958  19071 +         NewTrain = "Y";
4959  19072 +         ExportedComp = "&EM_PROPERTY_EXPORTEDCOMP";
4960  19073 +         HideVariable = "&EM_PROPERTY_HIDEVARIABLE";
4961  19074 +         NGCluster = 0; /* zero means no twostage */
4962  19075 +   run;
4963  19076 +   quit;
4964  19077 +%mend CreateVarclusMeta;
4965  NOTE: %INCLUDE (level 1) ending.
4966  NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMEXPL.V
      ARIABLECLUSTERING_MACROS2.SOURCE.
4967  19078 +
4968  19079 +%macro MakeInterClusCorrData(indata=, outdata=, nclu
      ster=, globalclusid=, RSquare=N, makeplotds=N, plotds=);
4969  19080 +    data &outdata(drop= _NCL_  _TYPE_);
```

```
4970 19081 +       set &indata;
4971 19082 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
     E_) eq 'CCORR') then delete;
4972 19083 +       %if &globalclusid ne %then %do;
4973 19084 +       _NAME_ = "GC&globalclusid._"||upcase(_NAME_);
4974 19085 +       rename _NAME_ = Cluster;
4975 19086 +       %end;
4976 19087 +       %else %do;
4977 19088 +        _NAME_ = upcase(_NAME_);
4978 19089 +       rename _NAME_ = Cluster;
4979 19090 +       %end;
4980 19091 +       label _NAME_ = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_clustername,  noquote))";
4981 19092 +    run;
4982 19093 +    data corr_tmp;
4983 19094 +        set &outdata;
4984 19095 +    run;
4985 19096 +
4986 19097 +    %let dsid = %sysfunc(open(work.corr_tmp));
4987 19098 +    %let nclus2= %eval(&ncluster+1);
4988 19099 +    data &outdata;
4989 19100 +         set &outdata;
4990 19101 +          %do i = 2 %to &nclus2;
4991 19102 +          %let i_1 = %eval(&i-1);
4992 19103 +            %let _name = %sysfunc(varname(&dsid, &i)
     );
4993 19104 +            %if &globalclusid ne %then
4994 19105 +                %do; %let _newName = GC&globalclusid
     ._CLUS&i_1;
4995 19106 +                     rename &_name = &_newName;
4996 19107 +                     *label &_name ="GC &globalclusi
     d : Cluster &i_1";
4997 19108 +                     label &_name = "%sysfunc(sasmsg
     (sashelp.dmine, rpt_varclus_label_gc_clusternum, noquote,
     &globalclusid, &i_1))";
4998 19109 +                %end;
```

```
4999  19110 +            %else
5000  19111 +                %do; %let _newName = CLUS&i_1;
5001  19112 +                     rename &_name = &_newName;
5002  19113 +                     *label &_name ="Cluster &i_1";
5003  19114 +                     label &_name = "%sysfunc(sasmsg
      (sashelp.dmine, rpt_varclus_label_clusternum, noquote,  &i_
      1))";
5004  19115 +                %end;
5005  19116 +           %end;
5006  19117 +           keep Cluster
5007  19118 +           %do i = 2 %to &nclus2;
5008  19119 +                %let _name = %sysfunc(varname(&dsid,
       &i));
5009  19120 +                &_name
5010  19121 +           %end;
5011  19122 +           ;
5012  19123 +     %let dsid= %sysfunc(close(&dsid));
5013  19124 +      run;
5014  19125 +      quit;
5015  19126 +
5016  19127 +     %if &RSquare eq Y %then %do;
5017  19128 +
5018  19129 +       data corr_tmp;
5019  19130 +        set &outdata;
5020  19131 +      run;
5021  19132 +
5022  19133 +      data &outdata;
5023  19134 +           set &outdata;
5024  19135 +           %let dsid = %sysfunc(open(work.corr_tmp));
5025  19136 +           %let nvar = %sysfunc(attrn(&dsid, NVAR));
5026  19137 +           %do i = 2 %to &nvar;
5027  19138 +             %let _name = %sysfunc(varname(&dsid, &i)
      );
5028  19139 +             %let _name_md = &_name.**2;
5029  19140 +                 &_name = &_name_md;
5030  19141 +           %end;
```

```
5031 19142 +      %let dsid= %sysfunc(close(&dsid));
5032 19143 +      run;
5033 19144 +    %end;
5034 19145 +
5035 19146 +    %if &makeplotds eq Y %then %do;
5036 19147 +     proc transpose data = &outdata
5037 19148 +          out=&plotds(drop=_LABEL_ rename=(_NAME_ =
     Y Cluster=X Col1= Correlation));
5038 19149 +          by cluster;
5039 19150 +     run;
5040 19151 +     data &plotds;
5041 19152 +          set  &plotds;
5042 19153 +          label x="%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_label_cluster, noquote))";
5043 19154 +          label Y="%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_label_cluster, noquote))";
5044 19155 +          label  Correlation="%sysfunc(sasmsg(sashel
     p.dmine, rpt_correlation_vlabel, noquote))";
5045 19156 +     run;
5046 19157 +     %end;
5047 19158 +     proc datasets lib = work nolist;
5048 19159 +           delete corr_tmp;
5049 19160 +     run;
5050 19161 +     quit;
5051 19162 +%mend MakeInterClusCorrData;
5052 19163 +
5053 19164 +%macro MakeOwnRSquare(indata=, outdata=, ncluster=,
     globalclusid=);
5054 19165 +    data _tmpds(drop= _NCL_);
5055 19166 +       set &indata;
5056 19167 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
     E_) in ('GROUP','RSQUARED')) then delete;
5057 19168 +       %if &globalclusid ne %then %do;
5058 19169 +           _NAME_ = "GC&globalclusid.";
5059 19170 +       rename _NAME_ = Cluster;
5060 19171 +       %end;
```

```
5061  19172 +       %else %do;
5062  19173 +        _NAME_ = "CLUS";
5063  19174 +        rename _NAME_ = Cluster;
5064  19175 +       %end;
5065  19176 +       label _NAME_ = "%sysfunc(sasmsg(sashelp.dmine
      , rpt_varclus_label_clustername,  noquote))";
5066  19177 +    run;
5067  19178 +    proc transpose data =_tmpds out =&outdata;
5068  19179 +    run;
5069  19180 +
5070  19181 +    data &outdata(drop=COL1);
5071  19182 +        %if &globalclusid ne %then %do;
5072  19183 +        length GCluster $16;
5073  19184 +        %end;
5074  19185 +        length Cluster $32;
5075  19186 +        length _NAME_ $32;
5076  19187 +         set &outdata;
5077  19188 +         _NAME_ = upcase(_NAME_);
5078  19189 +         rename _NAME_=Variable;
5079  19190 +         *label _NAME_="Variable";
5080  19191 +         label _NAME_ = "%sysfunc(sasmsg(sashelp.dmi
      ne, rpt_varclus_label_variable, noquote))" ;
5081  19192 +         label Cluster = "%sysfunc(sasmsg(sashelp.dm
      ine, rpt_varclus_label_cluster, noquote))" ;
5082  19193 +         label GCluster = "%sysfunc(sasmsg(sashelp.d
      mine, rpt_varclus_label_gcluster, noquote))" ;
5083  19194 +
5084  19195 +         %if &globalclusid ne %then %do;
5085  19196 +          GCluster = "GC&globalclusid";
5086  19197 +          Cluster = "GC&globalclusid._CLUS"||strip(C
      OL1);
5087  19198 +         %end;
5088  19199 +         %else %do;
5089  19200 +         Cluster = "CLUS"||strip(COL1);
5090  19201 +        %end;
5091  19202 +         rename COL2 = RSqWithOwnClusComp;
```

```
5092 19203 +         *label COL2 = "R-Square With Own Cluster Co
     mponent";
5093 19204 +         label COL2 = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_ownrsq, noquote))" ;
5094 19205 +
5095 19206 +   run;
5096 19207 +   proc sort data =&outdata;
5097 19208 +        by Cluster RSqWithOwnClusComp;
5098 19209 +   run;
5099 19210 +   proc datasets lib = work nolist;
5100 19211 +           delete _tmpds;
5101 19212 +   run;
5102 19213 +   quit;
5103 19214 +%mend MakeOwnRSquare;
5104 19215 +
5105 19216 +%macro MakeClusStructCorrData(indata=, outdata=, glo
     balclusid=, ncluster=, Rsquare=N);
5106 19217 +    data &outdata(drop= _NCL_  _TYPE_);
5107 19218 +       %if &globalclusid ne %then %do;
5108 19219 +        length GCluster $16;
5109 19220 +        %end;
5110 19221 +       set &indata;
5111 19222 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
     E_) eq 'STRUCTUR') then delete;
5112 19223 +       %if &globalclusid ne %then %do;
5113 19224 +         GCluster = "GC&globalclusid";
5114 19225 +         _NAME_ = "GC&globalclusid._"||upcase(_NAME_
     );
5115 19226 +         rename _NAME_ = Cluster;
5116 19227 +         label _NAME_ = "%sysfunc(sasmsg(sashelp.dmi
     ne, rpt_varclus_label_cluster, noquote))" ;
5117 19228 +         label GCluster = "%sysfunc(sasmsg(sashelp.d
     mine, rpt_varclus_label_gcluster, noquote))" ;
5118 19229 +
5119 19230 +       %end;
5120 19231 +       %else %do;
```

```
5121 19232 +         _NAME_ = upcase(_NAME_);
5122 19233 +         rename _NAME_ = Cluster;
5123 19234 +         label _NAME_ = "%sysfunc(sasmsg(sashelp.dmi
     ne, rpt_varclus_label_cluster, noquote))" ;
5124 19235 +         label GCluster = "%sysfunc(sasmsg(sashelp.d
     mine, rpt_varclus_label_gcluster, noquote))" ;
5125 19236 +
5126 19237 +       %end;
5127 19238 +     run;
5128 19239 +    %if &RSquare eq Y %then %do;
5129 19240 +     data corr_tmp;
5130 19241 +          set &outdata;
5131 19242 +     run;
5132 19243 +     %let istart = 2;
5133 19244 +     %if &globalclusid ne %then %let istart = 3;
5134 19245 +     data &outdata;
5135 19246 +          set &outdata;
5136 19247 +          %let dsid = %sysfunc(open(work.corr_tmp));
5137 19248 +          %let nvar = %sysfunc(attrn(&dsid, NVAR));
5138 19249 +          %do i =&istart %to &nvar;
5139 19250 +            %let _name = %sysfunc(varname(&dsid, &i)
     );
5140 19251 +            %let _name_md = &_name.**2;
5141 19252 +                &_name = &_name_md;
5142 19253 +          %end;
5143 19254 +      %let dsid= %sysfunc(close(&dsid));
5144 19255 +      run;
5145 19256 +      proc datasets lib = work nolist;
5146 19257 +           delete corr_tmp;
5147 19258 +      run;
5148 19259 +    %end;
5149 19260 +     quit;
5150 19261 +%mend MakeClusStructCorrData;
5151 19262 +
5152 19263 +/*
5153 19264 +%MakeClusStructCorrData(indata=playpen._outstat, out
```

```
      data=_structrsq , ncluster=7, Rsquare=Y);
5154  19265 +*/
5155  19266 +
5156  19267 +%macro FindNextClosestClusByVar(indata=, outdata=, g
      lobalclusid=, ncluster=);
5157  19268 +
5158  19269 +     /* The indata should be the outdata
5159  19270 +        from %MakeClusStructCorrData(indata=, outdat
      a=, ); */
5160  19271 +
5161  19272 +     proc sort data =&indata out=_tmpclusRsq;
5162  19273 +     by cluster;
5163  19274 +     run;
5164  19275 +
5165  19276 +     proc transpose data =_tmpclusRsq out=_tmpclusRs
      q;
5166  19277 +      by cluster;
5167  19278 +     run;
5168  19279 +
5169  19280 +     proc sort data=_tmpclusRsq;
5170  19281 +        by _NAME_ COL1;
5171  19282 +     run;
5172  19283 +
5173  19284 +     data _tmpclusRsq;
5174  19285 +         length _NAME_ $32;
5175  19286 +         set _tmpclusRsq; by _NAME_;
5176  19287 +          _NAME_ = upcase(_NAME_);
5177  19288 +          %if &ncluster ne 1 %then %do;
5178  19289 +             if last._NAME_ then delete;
5179  19290 +          %end;
5180  19291 +          %else %do;
5181  19292 +             COL1 = 0;
5182  19293 +          %end;
5183  19294 +     run;
5184  19295 +     /* need to sort again */
5185  19296 +     proc sort data=_tmpclusRsq;
```

```
5186 19297 +        by _NAME_ COL1;
5187 19298 +     run;
5188 19299 +
5189 19300 +     data &outdata;
5190 19301 +         set _tmpclusRsq; by _NAME_;
5191 19302 +         Cluster = upcase(Cluster);
5192 19303 +         if last._NAME_ then output;
5193 19304 +         *label  COL1 = 'R-Sqaure with Next Cluster
     Component';
5194 19305 +         label COL1 = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_nextrsq, noquote))" ;
5195 19306 +         rename COL1 = RSqWithNextClusComp;
5196 19307 +         Cluster = upcase(Cluster);
5197 19308 +         rename Cluster = ClosestCluster;
5198 19309 +         *label  Cluster = "Next Closest Cluster";
5199 19310 +         label Cluster = "%sysfunc(sasmsg(sashelp.dm
     ine, rpt_varclus_label_nextclus, noquote))" ;
5200 19311 +         rename _NAME_ = Variable;
5201 19312 +         label  _NAME_ = "%sysfunc(sasmsg(sashelp.dm
     ine, rpt_varclus_label_variable, noquote))" ;
5202 19313 +    run;
5203 19314 +
5204 19315 +
5205 19316 +    %if &globalclusid ne %then %do;
5206 19317 +    data &outdata;
5207 19318 +       length GCluster $16;
5208 19319 +       set &outdata;
5209 19320 +       GCluster = "GC&globalclusid";
5210 19321 +       run;
5211 19322 +    %end;
5212 19323 +    proc datasets lib = work nolist;
5213 19324 +           delete _tmpclusRsq;
5214 19325 +    run;
5215 19326 +    quit;
5216 19327 +%mend FindNextClosestClusByVar;
5217 19328 +
```

```
5218 19329 +
5219 19330 +%macro FindNextClosestClusByCluster(indata=, outdata
     =, globalclusid=, ncluster=);
5220 19331 +     /* The indata should be the outdata from %MakeI
     nterClusCorrData(indata=, outdata=, ); */
5221 19332 +     proc sort data =&indata out=_tmpclusRsq;
5222 19333 +     by cluster;
5223 19334 +     run;
5224 19335 +     proc transpose data =_tmpclusRsq out=_tmpclusRs
     q;
5225 19336 +     by cluster;
5226 19337 +     run;
5227 19338 +     proc sort data=_tmpclusRsq;
5228 19339 +        by _NAME_ col1;
5229 19340 +     run;
5230 19341 +     data _tmpclusRsq;
5231 19342 +         length _NAME_ $32;
5232 19343 +         set _tmpclusRsq; by _NAME_;
5233 19344 +          _NAME_ = upcase(_NAME_);
5234 19345 +          %if &ncluster ne 1 %then %do;
5235 19346 +             if last._NAME_ then delete;
5236 19347 +          %end;
5237 19348 +          %else %do;
5238 19349 +             COL1 = 0;
5239 19350 +          %end;
5240 19351 +     run;
5241 19352 +     data &outdata;
5242 19353 +         set _tmpclusRsq; by _NAME_;
5243 19354 +         Cluster = upcase(Cluster);
5244 19355 +         if last._NAME_ then output;
5245 19356 +         *label  COL1 = 'R-Sqaure with Next Cluster
     Component';
5246 19357 +         label COL1 = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_nextrsq, noquote))" ;
5247 19358 +         rename COL1 = RSqWithNextClusComp;
5248 19359 +         Cluster = upcase(Cluster);
```

```
5249 19360 +         rename Cluster = ClosestCluster;
5250 19361 +         *label  Cluster = "Next Closest Cluster";
5251 19362 +         label Cluster = "%sysfunc(sasmsg(sashelp.dm
     ine, rpt_varclus_label_nextclus, noquote))" ;
5252 19363 +         rename _NAME_ = Variable;
5253 19364 +         *label  _NAME_ = "Variable";
5254 19365 +         label  _NAME_ = "%sysfunc(sasmsg(sashelp.dm
     ine, rpt_varclus_label_variable, noquote))" ;
5255 19366 +
5256 19367 +    run;
5257 19368 +    %if &globalclusid ne %then %do;
5258 19369 +    data &outdata;
5259 19370 +       length GCluster $16;
5260 19371 +       set &outdata;
5261 19372 +       GCluster = "GC&globalclusid";
5262 19373 +       run;
5263 19374 +    %end;
5264 19375 +
5265 19376 +    proc datasets lib = work nolist;
5266 19377 +           delete _tmpclusRsq;
5267 19378 +    run;
5268 19379 +
5269 19380 +    quit;
5270 19381 +%mend FindNextClosestClusByCluster;
5271 19382 +
5272 19383 +%macro MakeVarClusResultTable(indata1=, indata2=, in
     data3=, outdata=, globalclusid=, ncluster=, selectedcomp=cl
     ustercomp);
5273 19384 +/*----
5274 19385 +  indata1=_ownRsq, indata2=_nextVarRsq, indata3=_nex
     tClusRSq,
5275 19386 +-----------*/
5276 19387 +
5277 19388 +proc sort data =&indata1;
5278 19389 +    by Variable;
5279 19390 +run;
```

```
5280  19391 +proc sort data =&indata2;
5281  19392 +    by Variable;
5282  19393 +run;
5283  19394 +data &outdata;
5284  19395 +    merge &indata1 &indata2;
5285  19396 +    by Variable;
5286  19397 +    length Type $16;
5287  19398 +    Type = 'Variable';
5288  19399 +    *label Type ='Type';
5289  19400 +    label Type = "%sysfunc(sasmsg(sashelp.dmine, rpt
      _varclus_label_type, noquote))" ;
5290  19401 +run;
5291  19402 +
5292  19403 +
5293  19404 +data &indata3;
5294  19405 +    set  &indata3 ;
5295  19406 +    length RSqWithOwnClusComp 8.;
5296  19407 +    Cluster = Variable;
5297  19408 +    RSqWithOwnClusComp = 1;
5298  19409 +    *label RSqWithOwnClusComp = "R-Square With Own C
      luster Component";
5299  19410 +    label RSqWithOwnClusComp = "%sysfunc(sasmsg(sash
      elp.dmine, rpt_varclus_label_ownrsq, noquote))" ;
5300  19411 +    length Type $16;
5301  19412 +    Type = 'ClusterComp';
5302  19413 +    label Type = "%sysfunc(sasmsg(sashelp.dmine, rpt
      _varclus_label_type, noquote))" ;
5303  19414 +
5304  19415 +;
5305  19416 +run;
5306  19417 +
5307  19418 +proc sort data=&outdata;
5308  19419 +    by Cluster;
5309  19420 +run;
5310  19421 +proc sort data =&indata3;
5311  19422 +    by Cluster;
```

```
5312  19423 +run;
5313  19424 +
5314  19425 +data &outdata;
5315  19426 +     set &outdata &indata3;
5316  19427 +     by Cluster;
5317  19428 +run;
5318  19429 +
5319  19430 +
5320  19431 +/* Create the Selected variable with all YES */
5321  19432 +
5322  19433 +data &outdata;
5323  19434 +     set &outdata;
5324  19435 +     length RsqRatio 8.;
5325  19436 +     length Selected $8;
5326  19437 +     *label RSqRatio = "1-R**2 Ratio";
5327  19438 +     label RSqRatio =  "%sysfunc(sasmsg(sashelp.dmin
      e, rpt_varclus_label_oneminusrsq, noquote))" ;
5328  19439 +     *label Selected = "Variable Selected";
5329  19440 +     label Selected = "%sysfunc(sasmsg(sashelp.dmine
      , rpt_varclus_label_varselected, noquote))" ;
5330  19441 +     RsqRatio = (1-RSqWithOwnClusComp)/(1-RSqWithNex
      tClusComp);
5331  19442 +     Selected ='YES';
5332  19443 +     rename _LABEL_ = Label;
5333  19444 +     label _LABEL_ =  "%sysfunc(sasmsg(sashelp.dmine
      , rpt_varclus_label_label, noquote))" ;
5334  19445 +run;
5335  19446 +
5336  19447 +
5337  19448 +/*----  Selected = Y/N will be assigened at the %sco
      re                  -------+
5338  19449 +        Just create the Selected variable with all Y
      ES at the step above
5339  19450 + +-----------------------------------------------
      ----------------------------+
5340  19451 +
```

```
5341  19452 +proc sort data=&outdata;
5342  19453 +    by Cluster RsqRatio;
5343  19454 +run;
5344  19455 +
5345  19456 +%if &selectedcomp eq CLUSTERCOMP %then %do;
5346  19457 +data &outdata;
5347  19458 +     set &outdata; by Cluster;
5348  19459 +     length Selected $8;
5349  19460 +     label Selected = "Variable Selected";
5350  19461 +     if  first.Cluster then Selected ='Yes';
5351  19462 +     else Selected = 'No';
5352  19463 +    run;
5353  19464 +%end;
5354  19465 +%else %do;
5355  19466 +data &outdata(drop = _var _varchange);
5356  19467 +     set &outdata; retain _var 0; by Cluster;
5357  19468 +     length Selected $8;
5358  19469 +     label Selected = "Variable Selected";
5359  19470 +     if first.Cluster then  _varchange = 0;
5360  19471 +     else _varchange =1;
5361  19472 +     if _var ne _varchange then  Selected  = 'Yes';
5362  19473 +     else Selected = 'No';
5363  19474 +     if last.cluster then  _var = 0;
5364  19475 +     else _var = _varchange;
5365  19476 +run;
5366  19477 +%end;
5367  19478 +
5368  19479 +-------------------------------------------------
     ----------*/
5369  19480 +
5370  19481 +quit;
5371  19482 +%mend MakeVarClusResultTable;
5372  19483 +
5373  19484 +%Macro MakePlotDataFromCorrTable(indata=, outdata=,
     globalclusid=);
5374  19485 +     proc sort data =&indata;
```

```
5375  19486 +        by cluster;
5376  19487 +     run;
5377  19488 +     proc transpose data =&indata
5378  19489 +          out=&outdata(drop=_LABEL_ rename=(_NAME_ =
       Y Cluster=X Col1= Correlation));
5379  19490 +          by cluster;
5380  19491 +     run;
5381  19492 +     data &outdata;
5382  19493 +          set  &outdata;
5383  19494 +          label x= "%sysfunc(sasmsg(sashelp.dmine, r
      pt_varclus_label_cluster, noquote))" ;
5384  19495 +          label Y= "%sysfunc(sasmsg(sashelp.dmine, r
      pt_varclus_label_cluster, noquote))" ;
5385  19496 +          label Correlation = "%sysfunc(sasmsg(sashe
      lp.dmine, rpt_correlation_vlabel, noquote))" ;
5386  19497 +     run;
5387  19498 +     %if &globalclusid ne %then %do;
5388  19499 +     data &outdata;
5389  19500 +          Length GCluster $16;
5390  19501 +          label GCluster = "%sysfunc(sasmsg(sashelp.
      dmine, rpt_varclus_label_gcluster, noquote))" ;
5391  19502 +          set  &outdata;
5392  19503 +          GCluster ="GC&globalclusid.";
5393  19504 +          run;
5394  19505 +     %end;
5395  19506 +
5396  19507 +%Mend MakePlotDataFromCorrTable;
5397  19508 +
5398  19509 +
5399  19510 +%macro MakeCorrelation(indata=,
5400  19511 +                       outstat=_tmpoutstat,
5401  19512 +                       corrmatrix=N,
5402  19513 +                       outcorr=_tmpoutcorr,
5403  19514 +                       includeclassvar=N,
5404  19515 +                       target=,
5405  19516 +                       freq=,
```

```
5406  19517 +                        weight=);
5407  19518 +   %if &target eq  %then %do;
5408  19519 +
5409  19520 +       proc varclus data=&indata outstat=&outstat hi
      maxclusters=1 noprint;
5410  19521 +            var %EM_INTERVAL_INPUT %EM_INTERVAL_REJE
      CTED
5411  19522 +       %if &includeclassvar eq Y %then %do;
5412  19523 +          %let dsid = %sysfunc(open(&EM_USER_OUTDUMM
      Y));
5413  19524 +          %let nvar = %sysfunc(attrn(&dsid, NVAR));
5414  19525 +          %do i = 2 %to &nvar;
5415  19526 +          %let varname = %sysfunc(varname(&dsid, &i)
      );
5416  19527 +          &varname
5417  19528 +          %end;
5418  19529 +       %end;
5419  19530 +       ;
5420  19531 +      %if &freq ne  %then %do;
5421  19532 +          freq &freq;
5422  19533 +      %end;
5423  19534 +      %if &weight ne  %then %do;
5424  19535 +          weight &weight;
5425  19536 +      %end;
5426  19537 +
5427  19538 +      run;
5428  19539 +      %if &corrmatrix eq Y %then %do;
5429  19540 +      data  &outcorr (drop = _NCL_ _TYPE_);
5430  19541 +           set  &outstat ;
5431  19542 +           if _TYPE_ ='CORR' then output;
5432  19543 +      run;
5433  19544 +      %end;
5434  19545 +    %end;
5435  19546 +    %else %do;
5436  19547 +       proc corr data=&indata outp=&outstat noprint;
5437  19548 +            var
```

```
5438  19549 +       %let dsid = %sysfunc(open(&indata));
5439  19550 +       %let nvar = %sysfunc(attrn(&dsid, NVAR));
5440  19551 +          %do i = 1 %to &nvar;
5441  19552 +              %let _name = %sysfunc(varname(&dsid, &
      i));
5442  19553 +              %if &_name ne &target %then;
5443  19554 +              &_name
5444  19555 +           %end;
5445  19556 +        %let dsid= %sysfunc(close(&dsid));
5446  19557 +        ;
5447  19558 +        with &target;
5448  19559 +        run;
5449  19560 +    %end;
5450  19561 +    quit;
5451  19562 +%mend MakeCorrelation;
5452  19563 +
5453  19564 +
5454  19565 +%macro MakeCorrelationDistance(indata=,
5455  19566 +                                outdata=,
5456  19567 +                                rsquare = N
5457  19568 +                                );
5458  19569 +    data corr_tmp;
5459  19570 +        set &indata;
5460  19571 +        if _N_ = 1 then do;
5461  19572 +           output;
5462  19573 +           stop;
5463  19574 +         end;
5464  19575 +    run;
5465  19576 +    %if &outdata ne  %then %let  _outdata = &outdata
      ;
5466  19577 +    %else %let _outdata = &indata;
5467  19578 +
5468  19579 +    data &_outdata;
5469  19580 +         set &indata;
5470  19581 +
5471  19582 +         %let dsid = %sysfunc(open(work.corr_tmp));
```

```
5472  19583 +         %let nvar = %sysfunc(attrn(&dsid, NVAR));
5473  19584 +          %do i = 2 %to &nvar;
5474  19585 +              %let _name = %sysfunc(varname(&dsid, &
      i));
5475  19586 +              %if &rsquare eq Y %then %let _name_md
      = &_name.**2;
5476  19587 +              %else  %let _name_md = &_name;
5477  19588 +              &_name = 1- &_name_md;
5478  19589 +          %end;
5479  19590 +      %let dsid= %sysfunc(close(&dsid));
5480  19591 +      run;
5481  19592 +      proc datasets lib = work nolist;
5482  19593 +           delete corr_tmp;
5483  19594 +      run;
5484  19595 +      quit;
5485  19596 +  %mend MakeCorrelationDistance;
5486  19597 +
5487  19598 +
5488  19599 +%macro UpdateOutStatCorrToDistance(indata=, /* indat
      a should be a outstat from proc varclus */
5489  19600 +                                    rsquare = N
5490  19601 +                                    );
5491  19602 +    data corr_tmp;
5492  19603 +        set &indata;
5493  19604 +    run;
5494  19605 +    proc sql noprint;
5495  19606 +           update &indata
5496  19607 +           set
5497  19608 +     %let dsid = %sysfunc(open(work.corr_tmp));
5498  19609 +     %let nvar = %sysfunc(attrn(&dsid, NVAR));
5499  19610 +          %do i = 4 %to &nvar;
5500  19611 +          %let _name = %sysfunc(varname(&dsid, &i));
5501  19612 +             %if &rsquare eq Y %then %let _name_md =
       &_name.**2;
5502  19613 +            %else  %let _name_md = &_name;
5503  19614 +            %if &i < &nvar %then %do;
```

```
5504 19615 +              &_name = 1- &_name_md ,
5505 19616 +           %end;
5506 19617 +           %else %do;
5507 19618 +              &_name = &_name_md where _TYPE_ eq 'C
     ORR' ;
5508 19619 +           %end;
5509 19620 +         %end;
5510 19621 +      %let dsid= %sysfunc(close(&dsid));
5511 19622 +
5512 19623 +      select * from &indata;
5513 19624 +      run;
5514 19625 +     data &indata( drop = _NCL_);
5515 19626 +          set &indata;
5516 19627 +          if _TYPE_ not in ('CORR', 'STD', 'N', 'ME
     AN') then delete;
5517 19628 +          if _TYPE_ ='CORR' then _TYPE_ ='DISTANCE'
     ;
5518 19629 +     run;
5519 19630 +     data &indata(DROP = _NCL_);
5520 19631 +          set &indata;
5521 19632 +          if _TYPE_ = 'CORR' then _TYPE_ ='DISTANCE
     ';
5522 19633 +          if _TYPE_ not in ('DISTANCE', 'N', 'STD',
      'MEAN') then delete;
5523 19634 +          rename _NAME_ = _VAR_;
5524 19635 +     run;
5525 19636 +     proc datasets lib = work nolist;
5526 19637 +          delete corr_tmp;
5527 19638 +     run;
5528 19639 +     quit;
5529 19640 + %mend UpdateOutStatCorrToDistance;
5530 19641 +
5531 19642 +
5532 19643 +%macro HierClusWithCorr(indata= ,
5533 19644 +                        ncluster=,
5534 19645 +                        method = Ward,
```

```
5535 19646 +                        outtree = _outtree,
5536 19647 +                        idvar =_VAR_,
5537 19648 +                        outdata=,
5538 19649 +                        rescore = N,
5539 19650 +                        newncluster=
5540 19651 +                        );
5541 19652 +      %global &newncluster;
5542 19653 +      %if &rescore ne Y %then %do;
5543 19654 +      proc cluster data=&indata(type=Distance where=
     (upcase(strip(_TYPE_)) = "DISTANCE"))
5544 19655 +                   method=&method outtree=&outtree n
     oprint;
5545 19656 +           id &idvar;
5546 19657 +      run;
5547 19658 +      %end;
5548 19659 +      proc tree data=&outtree nclusters = &ncluster
     out=&outdata noprint;
5549 19660 +      run;
5550 19661 +      /* ----- Check some variables like CL1, CL5...
     , remove them ----*/
5551 19662 +      proc contents data =&indata out=_outcontent(ke
     ep=NAME) noprint;
5552 19663 +      run;
5553 19664 +      data _outcontent;
5554 19665 +          set _outcontent;
5555 19666 +          if NAME in ('_TYPE_' , '_VAR_') then delet
     e;
5556 19667 +          index = 1;
5557 19668 +          rename NAME = _NAME_;
5558 19669 +      run;
5559 19670 +      proc sort data=_outcontent;
5560 19671 +          by _NAME_;
5561 19672 +      run;
5562 19673 +      proc sort data =&outdata;
5563 19674 +           by _NAME_;
5564 19675 +      run;
```

```
5565 19676 +      data &outdata(drop=index);
5566 19677 +          merge &outdata _outcontent;
5567 19678 +          by _NAME_;
5568 19679 +          if index = . then delete;
5569 19680 +      run;
5570 19681 +      /*-----------------------------------------
     ---------------*/
5571 19682 +      data &outdata;
5572 19683 +           length CLUSNAME $16;
5573 19684 +           set &outdata;
5574 19685 +           if CLUSTER > &ncluster then delete;
5575 19686 +           CLUSNAME='GC'||strip(CLUSTER);
5576 19687 +           *label CLUSNAME = "Cluster Name";
5577 19688 +           label CLUSNAME  = "%sysfunc(sasmsg(sashel
     p.dmine, rpt_varclus_label_clustername, noquote))" ;
5578 19689 +           rename _NAME_ = VARIABLE ;
5579 19690 +           *label _NAME_ = "Variable";
5580 19691 +           *label CLUSTER = "Cluster";
5581 19692 +           label _NAME_ ="%sysfunc(sasmsg(sashelp.dm
     ine, rpt_varclus_label_variable, noquote))" ;
5582 19693 +           label CLUSTER ="%sysfunc(sasmsg(sashelp.d
     mine, rpt_varclus_label_cluster, noquote))" ;
5583 19694 +      run;
5584 19695 +      proc sort data=&outdata out=&outdata;
5585 19696 +           by CLUSTER;
5586 19697 +      run;
5587 19698 +      proc means data =&outdata noprint;
5588 19699 +           output out=_meanout;
5589 19700 +      run;
5590 19701 +      data _null_;
5591 19702 +           set _meanout;
5592 19703 +           if strip(_STAT_) eq 'MAX' then do;
5593 19704 +           call symput("&newncluster", CLUSTER);
5594 19705 +           stop;
5595 19706 +           end;
5596 19707 +      run;
```

```
5597 19708 +
5598 19709 +      proc datasets lib = work nolist;
5599 19710 +           delete _outcontent _meanout;
5600 19711 +      run;
5601 19712 +      quit;
5602 19713 +%mend  HierClusWithCorr;
5603 19714 +
5604 19715 +%macro CreateScoreCode2(indata=, ncluscomp=, globalc
     lusid=, fileref=);
5605 19716 +
5606 19717 +     data _tmpindata;
5607 19718 +          set &indata;
5608 19719 +          if (_TYPE_ in ('SCORE' 'MEAN' 'STD') and _
     NCL_ = &ncluscomp ) or (_TYPE_ in ('MEAN' 'STD'));
5609 19720 +          if _TYPE_ = 'MEAN' then _NAME_='MEAN';
5610 19721 +          if _TYPE_ = 'STD' then _NAME_='STD';
5611 19722 +          if _TYPE_ = 'SCORE' then _NAME_=upcase("GC
     &globalclusid._"||_NAME_);
5612 19723 +          DROP _TYPE_ _NCL_;
5613 19724 +     run;
5614 19725 +
5615 19726 +     filename _file_  "&fileRef";
5616 19727 +
5617 19728 +     data _null_;
5618 19729 +        FILE _file_ MOD;
5619 19730 +        put ' ';
5620 19731 +        put "/*-------------------------------------
     ------------*/";
5621 19732 +        put '/* ' "%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_score_title_gclus, noquote, &globalclusid))" '*/'
     ;
5622 19733 +        put "/*-------------------------------------
     ------------*/";
5623 19734 +        put ' ';
5624 19735 +        %let dsid = %sysfunc(open(work._tmpindata));
5625 19736 +
```

```
5626 19737 +        %let nvar = %sysfunc(attrn(&dsid, NVAR));
5627 19738 +        %let vn_name =%sysfunc(varnum(&dsid, _NAME_)
     );
5628 19739 +        %let k = 1;
5629 19740 +        %do %while(^%sysfunc(fetch(&dsid)));
5630 19741 +              %let _name = %sysfunc(getvarc(&dsid,
     &vn_name));
5631 19742 +              %if &k > 2 %then %do;
5632 19743 +               %let cn = %eval(&k-2);
5633 19744 +                put "&_name = 0 ; /*---" "%sysfunc(
     sasmsg(sashelp.dmine, rpt_varclus_score_gcluscompnum, noquo
     te, &globalclusid, &cn))" "------ */";
5634 19745 +               %end;
5635 19746 +               %let k = %eval(&k+1);
5636 19747 +
5637 19748 +        %end;
5638 19749 +        %let rc = %sysfunc(rewind(&dsid));
5639 19750 +
5640 19751 +        %do j= 2 %to &nvar;
5641 19752 +            %let _varname = %sysfunc(varname(&dsid,
     &j));
5642 19753 +            %do %while(^%sysfunc(fetch(&dsid)));
5643 19754 +               %let _name = %sysfunc(getvarc(&dsid,
      &vn_name));
5644 19755 +               %if &_name = MEAN %then
5645 19756 +               %let _mean = %sysfunc(getvarn(&dsid,
      &j));
5646 19757 +               %else %if &_name = STD %then
5647 19758 +               %let _std = %sysfunc(getvarn(&dsid,
     &j));
5648 19759 +               %else %do;
5649 19760 +                     %let coeff =  %sysfunc(getvarn
     (&dsid, &j));
5650 19761 +                     %let abscoeff = %sysfunc(abs(&
     coeff));
5651 19762 +                        %if &abscoeff >  0 %then %
```

```
     do;
5652 19763 +                        put "&_name = &_name+&coeff *
      (&_varname - &_mean)/&_std;";
5653 19764 +                           %end;
5654 19765 +                %end;
5655 19766 +             %end;
5656 19767 +             %let rc = %sysfunc(rewind(&dsid));
5657 19768 +          %end;
5658 19769 +
5659 19770 +        %let dsid= %sysfunc(close(&dsid));
5660 19771 +
5661 19772 +     run;
5662 19773 +
5663 19774 +     filename _file_;
5664 19775 +     proc datasets lib = work nolist;
5665 19776 +           delete _tmpindata;
5666 19777 +     run;
5667 19778 +     quit;
5668 19779 +%mend CreateScoreCode2;
5669 19780 +
5670 19781 +
5671 19782 +%macro MakeDeltaCode2(groupds=,deltacodefile=);
5672 19783 +
5673 19784 +     /*--- Build Code to Modify Metadata ---*/
5674 19785 +     filename X "&deltacodefile";
5675 19786 +     data _null_;
5676 19787 +        FILE X;
5677 19788 +        set &groupds end=eof;
5678 19789 +        if _N_=1 then do;
5679 19790 +           %if &EM_PROPERTY_INCLUDECLASSVAR eq Y %th
     en %do;
5680 19791 +            put "if upcase(strip(ROLE)) ='INPUT' and
      upcase(strip(LEVEL)) ^='INTERVAL' then ROLE ='REJECTED' ;"
     ;
5681 19792 +           %end;
5682 19793 +           put "if upcase(strip(ROLE))='INPUT' and u
```

```
     pcase(strip(LEVEL))='INTERVAL' then do;";
5683 19794 +           put "if upcase(strip(NAME)) in (";
5684 19795 +        end;
5685 19796 +        if Strip(upcase(Selected)) eq 'YES' then do;
5686 19797 +           string = '"'!!trim(left(VARIABLE))!!'"';
5687 19798 +           put string;
5688 19799 +        end;
5689 19800 +        if eof then do;
5690 19801 +           put ') then ROLE="INPUT";';
5691 19802 +           put 'else ROLE="REJECTED";';
5692 19803 +           put 'end;';
5693 19804 +
5694 19805 +           %if %upcase(&EM_PROPERTY_HIDEVARIABLE) eq
      Y %then %do;
5695 19806 +             put 'if upcase(strip(ROLE)) = "REJECTED
     " then delete ;';
5696 19807 +           %end;
5697 19808 +        end;
5698 19809 +     run;
5699 19810 +     quit;
5700 19811 +
5701 19812 +     filename X;
5702 19813 +     quit;
5703 19814 +%mend MakeDeltaCode2;
5704 19815 +
5705 19816 +%macro getInitialGClusterNumber(indata=, ninput=, nd
     ummy=0, div=100, ngc=);
5706 19817 +  %global &ngc;
5707 19818 +  data _null_;
5708 19819 +  %if &indata ne %then %do;
5709 19820 +      %let dsid = %sysfunc(open(&indata));
5710 19821 +           %let nvar = %sysfunc(attrn(&dsid, NVAR));
5711 19822 +      %let dsid = %sysfunc(close(&dsid));
5712 19823 +  %end;
5713 19824 +  %else %do;
5714 19825 +       %let nvar = %eval(&ninput+&ndummy); ;
```

```
5715 19826 +  %end;
5716 19827 +  %let numgc = %eval(&nvar/&div+2);
5717 19828 +  %let &ngc = &numgc;
5718 19829 +   run;
5719 19830 +  quit;
5720 19831 +%mend getInitialGClusterNumber;
5721 19832 +
5722 19833 +
5723 19834 +%macro MakeGobalConstellData(indata=, outlink=, outn
     ode=);
5724 19835 +data &outlink(drop = Selected);
5725 19836 +     set &indata;
5726 19837 +     LINKID = _N_;
5727 19838 +     label LINKID = "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_linkid, noquote))" ;
5728 19839 +run;
5729 19840 +data &outnode(keep=NODEID TYPE LABEL);
5730 19841 +    set &indata;
5731 19842 +    length TYPE $16;
5732 19843 +    rename VARIABLE = NODEID;
5733 19844 +    *label  CLUSNAME="Node ID";
5734 19845 +    label CLUSNAME= "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_nodeidvar, noquote))" ;
5735 19846 +    TYPE = "VARIABLE";
5736 19847 +    *label TYPE = "Node Type";
5737 19848 +    label TYPE =  "%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_label_nodetype, noquote))" ;
5738 19849 +    run;
5739 19850 +data _tmp(keep=NODEID TYPE LABEL);
5740 19851 +    set &indata;
5741 19852 +    length TYPE $16;
5742 19853 +    rename CLUSNAME = NODEID;
5743 19854 +    label  CLUSNAME= "%sysfunc(sasmsg(sashelp.dmine,
      rpt_varclus_label_nodeidvar, noquote))" ;
5744 19855 +    TYPE = "GCLUSTER";
5745 19856 +    label TYPE = "%sysfunc(sasmsg(sashelp.dmine, rpt
```

```
     _varclus_label_nodetype, noquote))" ;
5746 19857 +    run;
5747 19858 +proc sort data=_tmp;
5748 19859 +    by NODEID;
5749 19860 +run;
5750 19861 +data _tmp;
5751 19862 +    set _tmp; by NODEID;
5752 19863 +    if first.NODEID then output;
5753 19864 +run;
5754 19865 +proc sort data=&outnode;
5755 19866 +    by NODEID;
5756 19867 +run;
5757 19868 +data  &outnode;
5758 19869 +    set _tmp &outnode;
5759 19870 +run;
5760 19871 +proc datasets lib = work nolist;
5761 19872 +     delete _tmp;
5762 19873 +run;
5763 19874 +quit;
5764 19875 +%mend MakeGobalConstellData;
5765 19876 +
5766 19877 +/* Make contellation plot data among GCLUSTERS */
5767 19878 +
5768 19879 +%Macro MakeGClusterConstData(indata=, inoutrsq=, out
     node=, outlink=);
5769 19880 +
5770 19881 +data &outlink(keep = _NAME_ _PARENT_ _LABEL_ LINKID)
     ;
5771 19882 +     set &indata;
5772 19883 +     LINKID = _N_;
5773 19884 +     if upcase(substr(strip(_NAME_),1, 2))="CL" then
      do;
5774 19885 +        _NAME_ = "ROOT"||upcase(substr(strip(_NAME_)
     ,5));
5775 19886 +     end;
5776 19887 +     if _PARENT_ ne " " and upcase(substr(strip(_PAR
```

```
     ENT_),1, 2))="CL" then do;
5777 19888 +        _PARENT_ = "ROOT"||upcase(substr(strip(_PARE
     NT_),5));
5778 19889 +     end;
5779 19890 +     if upcase(substr(strip(_LABEL_),1, 2))="CL" the
     n do;
5780 19891 +        _LABEL_ = "ROOT"||upcase(substr(strip(_LABEL
     _),5));
5781 19892 +    end;
5782 19893 +run;
5783 19894 +
5784 19895 +data _tmp_outrsquare;
5785 19896 +        set &inoutrsq;
5786 19897 +        if upcase(strip(TYPE)) = 'CLUSTERCOMP' then
     delete;
5787 19898 +run;
5788 19899 +
5789 19900 +proc freq data =_tmp_outrsquare noprint;
5790 19901 +         tables GCluster/out=_tmp_GCLUSFREQ(rename=(
     GCLUSTER=_NAME_));
5791 19902 +run;
5792 19903 +
5793 19904 +data &outnode(keep=_NAME_ TYPE LABEL);
5794 19905 +    set &outlink;
5795 19906 +    length TYPE $16;
5796 19907 +    length LABEL $100;
5797 19908 +    /*label CLUSNAME= "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_nodeidvar, noquote))" ;*/
5798 19909 +
5799 19910 +    if upcase(substr(strip(_NAME_),1, 2))='GC' then
     do;
5800 19911 +    TYPE = "GCLUSTER";
5801 19912 +    LABEL = "%sysfunc(sasmsg(sashelp.dmine, rpt_varc
     lus_label_gcluster, noquote)):"||_NAME_;
5802 19913 +    end;
5803 19914 +    else do;
```

```
5804 19915 +    TYPE= "ROOT";
5805 19916 +    LABEL= _NAME_;
5806 19917 +    end;
5807 19918 +    label TYPE  =  "%sysfunc(sasmsg(sashelp.dmine, r
     pt_varclus_label_nodetype, noquote))" ;
5808 19919 +    label LABEL =  "%sysfunc(sasmsg(sashelp.dmine, r
     pt_varclus_label_label, noquote))" ;
5809 19920 +    label _NAME_ =  "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_nodeidvar, noquote))" ;
5810 19921 +    run;
5811 19922 +
5812 19923 +proc sort data=&outnode;
5813 19924 +     by _NAME_;
5814 19925 +proc sort data=_tmp_GCLUSFREQ;
5815 19926 +     by _NAME_;
5816 19927 +run;
5817 19928 +
5818 19929 +data  &outnode;
5819 19930 +     merge &outnode _tmp_GCLUSFREQ; by _NAME_;
5820 19931 +     if COUNT=. then COUNT=1;
5821 19932 +run;
5822 19933 +
5823 19934 +proc datasets lib = work nolist;
5824 19935 +     delete _tmp_outrsquare _tmp_GCLUSFREQ;
5825 19936 +run;
5826 19937 +
5827 19938 +quit;
5828 19939 +%Mend MakeGClusterConstData;
5829 19940 +
5830 19941 +
5831 19942 +%macro CreateGClusterScoreCode(indata=,  globalclusi
     d=, fileref=);
5832 19943 +
5833 19944 +    data _gscoretmpds;
5834 19945 +          set &indata;
5835 19946 +          if (_TYPE_ in ('SCORE' 'MEAN' 'STD') and _
```

```
     NCL_ = 1 ) or (_TYPE_ in ('MEAN' 'STD'));
5836 19947 +          if _TYPE_ = 'MEAN' then _NAME_='MEAN';
5837 19948 +          if _TYPE_ = 'STD' then _NAME_='STD';
5838 19949 +          if _TYPE_ = 'SCORE' then _NAME_ = "GC"||st
     rip(&globalclusid);
5839 19950 +          DROP _TYPE_ _NCL_;
5840 19951 +     run;
5841 19952 +
5842 19953 +    /*  %let gscorefile =  %bquote(&EM_NODEDIR)&EM_D
     SEP.gclusterscore.sas;
5843 19954 +        GCluster Component &globalclusid ------ */
5844 19955 +
5845 19956 +    filename _file_  "&fileref";
5846 19957 +
5847 19958 +    data _null_;
5848 19959 +        %if &globalclusid eq 1 %then %do;
5849 19960 +          FILE _file_;
5850 19961 +        %end;
5851 19962 +        %else %do;
5852 19963 +          FILE _file_ MOD;
5853 19964 +        %end;
5854 19965 +
5855 19966 +        %let dsid = %sysfunc(open(work._gscoretmpds)
     );
5856 19967 +        %let nvar = %sysfunc(attrn(&dsid, NVAR));
5857 19968 +        %let vn_name =%sysfunc(varnum(&dsid, _NAME_)
     );
5858 19969 +
5859 19970 +        %let k = 1;
5860 19971 +        %do %while(^%sysfunc(fetch(&dsid)));
5861 19972 +                %let _name = %sysfunc(getvarc(&dsid,
      &vn_name));
5862 19973 +                %if &k > 2 %then %do;
5863 19974 +                put "&_name = 0 ; ";
5864 19975 +                %end;
5865 19976 +                %let k = %eval(&k+1);
```

```
5866  19977 +        %end;
5867  19978 +
5868  19979 +        %let rc = %sysfunc(rewind(&dsid));
5869  19980 +        %do i= 2 %to &nvar;
5870  19981 +            %let _varname =  %sysfunc(varname(&dsid,
       &i));
5871  19982 +            %do %while(^%sysfunc(fetch(&dsid)));
5872  19983 +                %let _name = %sysfunc(getvarc(&dsid,
       &vn_name));
5873  19984 +                %if &_name = MEAN %then
5874  19985 +                %let _mean = %sysfunc(getvarn(&dsid,
       &i));
5875  19986 +                %else %if &_name = STD %then
5876  19987 +                %let _std = %sysfunc(getvarn(&dsid,
      &i));
5877  19988 +                %else %do;
5878  19989 +                      %let coeff =  %sysfunc(getvarn
      (&dsid, &i));
5879  19990 +                      %let abscoeff = %sysfunc(abs(&
      coeff));
5880  19991 +                         %if &abscoeff >  0 %then %
      do;
5881  19992 +                          put "&_name = &_name+&coeff *
       (&_varname - &_mean)/&_std;";
5882  19993 +                           %end;
5883  19994 +                %end;
5884  19995 +            %end;
5885  19996 +            %let rc = %sysfunc(rewind(&dsid));
5886  19997 +
5887  19998 +         %end;
5888  19999 +
5889  20000 +        %let dsid= %sysfunc(close(&dsid));
5890  20001 +       run;
5891  20002 +
5892  20003 +
5893  20004 +       proc datasets lib=work nolist;
```

```
5894  20005 +            delete _gscoretmpds;
5895  20006 +       run;
5896  20007 +      quit;
5897  20008 +
5898  20009 +%mend CreateGClusterScoreCode;
5899  20010 +
5900  20011 +
5901  20012 +%macro MakeGClusterCorrelation(Indata=, ngcluster=,
      gscorecode=, outrsquare=);
5902  20013 +
5903  20014 +   %EM_REGISTER(KEY=GSCORE, TYPE=DATA);
5904  20015 +   %EM_GETNAME(KEY=GSCORE, TYPE=DATA);
5905  20016 +   %EM_REGISTER(KEY=GSCORESTAT, TYPE=DATA);
5906  20017 +   %EM_GETNAME(KEY=GSCORESTAT, TYPE=DATA);
5907  20018 +   %EM_REGISTER(KEY=GSCORETREE, TYPE=DATA);
5908  20019 +   %EM_GETNAME(KEY=GSCORETREE, TYPE=DATA);
5909  20020 +   %EM_REGISTER(KEY=GSCORECORR, TYPE=DATA);
5910  20021 +   %EM_GETNAME(KEY=GSCORECORR, TYPE=DATA);
5911  20022 +   %EM_REGISTER(KEY=GSCORECORRPLOT, TYPE=DATA);
5912  20023 +   %EM_GETNAME(KEY=GSCORECORRPLOT, TYPE=DATA);
5913  20024 +   %EM_REGISTER(KEY=GCLUSLINK, TYPE=DATA);
5914  20025 +   %EM_GETNAME(KEY=GCLUSLINK, TYPE=DATA);
5915  20026 +   %EM_REGISTER(KEY=GCLUSNODE, TYPE=DATA);
5916  20027 +   %EM_GETNAME(KEY=GCLUSNODE, TYPE=DATA);
5917  20028 +
5918  20029 +   filename gsfile "&gscorecode";
5919  20030 +
5920  20031 +   data &EM_USER_GSCORE;
5921  20032 +             set &indata;
5922  20033 +             %include  gsfile;
5923  20034 +          keep
5924  20035 +          %do i=1 %to &ngcluster;
5925  20036 +           %let gcvarname = GC&i;
5926  20037 +           &gcvarname
5927  20038 +          %end;
5928  20039 +     ;
```

```
5929 20040 +   run;
5930 20041 +
5931 20042 +   proc varclus data=&EM_USER_GSCORE outstat=&EM_USE
     R_GSCORESTAT outtree=&EM_USER_GSCORETREE
5932 20043 +    %if %upcase(&EM_PROPERTY_CLUSCOMP) eq CENTROID %
     then %do; centroid %end;
5933 20044 +    %if %upcase(&EM_PROPERTY_CLUSSOURCE) eq COV %the
     n %do; cov %end;
5934 20045 +    %if %upcase(&EM_PROPERTY_CLUSHIERACHY) eq Y %the
     n %do; hi %end;
5935 20046 +    noprint  ;
5936 20047 +     var
5937 20048 +       %do i=1 %to &ngcluster;
5938 20049 +         %let gcvarname = GC&i;
5939 20050 +         &gcvarname
5940 20051 +        %end;
5941 20052 +   ;
5942 20053 +   run;
5943 20054 +
5944 20055 +
5945 20056 +   %MakeVarClusCorrData(statds=&EM_USER_GSCORESTAT,
     corrds=&EM_USER_GSCORECORR, corrplotds=&EM_USER_GSCORECORRP
     LOT );
5946 20057 +   data &EM_USER_GSCORECORRPLOT ;
5947 20058 +        set &EM_USER_GSCORECORRPLOT;
5948 20059 +        rename _X_ = X;
5949 20060 +        rename _Y_ = Y;
5950 20061 +        label _X_ = "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_gcluster, noquote))" ;
5951 20062 +        label _Y_ = "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_gcluster, noquote))" ;
5952 20063 +   run;
5953 20064 +
5954 20065 +   %MakeGClusterConstData(indata=&EM_USER_GSCORETREE
     , inoutrsq=&outrsquare, outnode=&EM_USER_GCLUSNODE, outlink
     =&EM_USER_GCLUSLINK);
```

```
5955  20066 +
5956  20067 +   data &EM_USER_GSCORETREE;
5957  20068 +        length _NAME_ $32;
5958  20069 +        length _LABEL_ $100;
5959  20070 +        set &EM_USER_GSCORETREE(DROP=_LABEL_);
5960  20071 +        if upcase(substr(strip(_NAME_),1, 2))='GC' t
      hen do;
5961  20072 +         _LABEL_ = "%sysfunc(sasmsg(sashelp.dmine, r
      pt_varclus_label_gcluster, noquote)):"||_NAME_;
5962  20073 +        end;else do;
5963  20074 +         _LABEL_ = _NAME_;
5964  20075 +        end;
5965  20076 +       label _LABEL_ = "%sysfunc(sasmsg(sashelp.dmin
      e, rpt_varclus_label_variable, noquote))";
5966  20077 +
5967  20078 +   run;
5968  20079 +
5969  20080 +   quit;
5970  20081 +
5971  20082 +%mend MakeGClusterCorrelation;
5972  20083 +
5973  20084 +
5974  NOTE: %INCLUDE (level 1) ending.
5975  NOTE: Fileref TEMP has been deassigned.
5976  NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMEXPL.V
      ARIABLECLUSTERING_SCORE.SOURCE.
5977  20085 +%macro score;
5978  20086 +
5979  20087 +filename temp catalog 'sashelp.emexpl.variablecluste
      ring_score_macros.source';
5980  20088 +%include temp;
5981  20089 +filename temp;
5982  20090 +
5983  20091 +%EM_GETNAME(key=VARCLUSMETA, type=DATA) ;
5984  20092 +
5985  20093 +data _null_;
```

```
5986 20094 +         set &EM_USER_VARCLUSMETA;
5987 20095 +         if _N_=1 then
5988 20096 +         call symput('_trainnum', TrainNum);
5989 20097 +         call symput('_exportedcomp', ExportedComp);
5990 20098 +         call symput('_hidevariable', HideVariable);
5991 20099 +         call symput('_newTrain', NewTrain);
5992 20100 +run;
5993 20101 +
5994 20102 +%if &_newTrain = N %then %do;
5995 20103 +        %if &EM_PROPERTY_EXPORTEDCOMP eq &_exportedc
     omp %then %let _exportedCompChanged = N;
5996 20104 +        %else %let _exportedCompChanged = Y;
5997 20105 +        %if &EM_PROPERTY_HIDEVARIABLE eq &_hidevaria
     ble %then %let _hideVariableChanged = N;
5998 20106 +        %else  %let _hideVariableChanged = Y;
5999 20107 +%end;
6000 20108 +%else %do;
6001 20109 +        %let _exportedCompChanged = Y;
6002 20110 +        %let _hideVariableChanged = Y;
6003 20111 +%end;
6004 20112 +
6005 20113 +%if (&_trainnum = 1 ) or %upcase(&EM_PROPERTY_TWOSTA
     GECLUS) = NO %then %do;
6006 20114 +     filename temp catalog 'sashelp.emexpl.variablec
     lustering_score1.source';
6007 20115 +     %include temp;
6008 20116 +     filename temp;
6009 20117 +     %score1(ExportedCompChanged=&_exportedCompChang
     ed, HideVariableChanged=&_hideVariableChanged);
6010 20118 +%end;
6011 20119 +%if (&_trainnum = 2 ) or %upcase(&EM_PROPERTY_TWOSTA
     GECLUS) = YES %then %do;
6012 20120 +     filename temp catalog 'sashelp.emexpl.variablec
     lustering_score2.source';
6013 20121 +     %include temp;
6014 20122 +     filename temp;
```

```
6015 20123 +     %score2(ExportedCompChanged=&_exportedCompChang
     ed, HideVariableChanged=&_hideVariableChanged);
6016 20124 +%end;
6017 20125 +
6018 20126 +  /* store  current property values */
6019 20127 + data &EM_USER_VARCLUSMETA;
6020 20128 +       set &EM_USER_VARCLUSMETA;
6021 20129 +        ExportedComp = "&EM_PROPERTY_EXPORTEDCOMP";
6022 20130 +        HideVariable = "&EM_PROPERTY_HIDEVARIABLE";
6023 20131 + run;
6024 20132 +
6025 20133 +%mend score;
6026 20134 +
6027 20135 +
6028 NOTE: %INCLUDE (level 1) ending.
6029 NOTE: Fileref TEMP has been deassigned.
6030 NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMEXPL.V
     ARIABLECLUSTERING_SCORE_MACROS.SOURCE.
6031 20136 +
6032 20137 +%macro ChangeVariableSelectionFlag(Indata=, gcluster
     =);
6033 20138 +     proc sort data =&indata;
6034 20139 +               by Cluster RsqRatio Type;
6035 20140 +     run;
6036 20141 +     /*-----------------------------------------
     -------------------------
6037 20142 +      The Type variabe is used becuse the variable n
     eed to be selected
6038 20143 +        instead of Cluster component when only one v
     ariable is in the cluster
6039 20144 +     +-----------------------------------------
     -------------------------*/
6040 20145 +
6041 20146 +     %if &EM_PROPERTY_EXPORTEDCOMP eq CLUSTERCOMP %t
     hen %do;
6042 20147 +         data &indata;
```

```
6043  20148 +              set &indata; by cluster;
6044  20149 +              if first.Cluster then Selected = 'YES'
      ;
6045  20150 +              else Selected = 'NO';
6046  20151 +         run;
6047  20152 +         quit;
6048  20153 +    %end;
6049  20154 +    %else %do;
6050  20155 +         data &indata(drop = _var _varchange);
6051  20156 +              set &indata; retain _var 0; by Cluster
      ;
6052  20157 +             if first.Cluster then  _varchange = 0;
6053  20158 +                 else _varchange =1;
6054  20159 +             if _var ne _varchange then  Selected  =
       'YES';
6055  20160 +             else Selected = 'NO';
6056  20161 +             if last.cluster then  _var = 0;
6057  20162 +             else _var = _varchange;
6058  20163 +         run;
6059  20164 +         quit;
6060  20165 +    %end;
6061  20166 +%mend ChangeVariableSelectionFlag;
6062  20167 +
6063  NOTE: %INCLUDE (level 1) ending.
6064  NOTE: Fileref TEMP has been deassigned.
6065
6066  NOTE: Numeric values have been converted to character value
      s at the places given by: (Line):(Column).
6067        3:15
6068  NOTE: There were 1 observations read from the data set EMWS
      4.VARCLUS_VARCLUSMETA.
6069  NOTE: DATA statement used (Total process time):
6070        real time           0.00 seconds
6071        user cpu time       0.00 seconds
6072        system cpu time     0.00 seconds
6073        memory              159319.37k
```

```
6074        OS Memory           169600.00k
6075        Timestamp           07/01/2024 05:54:29 AM
6076        Step Count                        1  Switch Count  0
6077        Page Faults                       0
6078        Page Reclaims                     63
6079        Page Swaps                        0
6080        Voluntary Context Switches        7
6081        Involuntary Context Switches      0
6082        Block Input Operations            288
6083        Block Output Operations           0
6084
6085
6086 NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMEXPL.V
     ARIABLECLUSTERING_SCORE1.SOURCE.
6087 20168 +
6088 20169 +%macro score1(ExportedCompChanged=,
6089 20170 +              HideVariableChanged=);
6090 20171 +
6091 20172 +   %EM_GETNAME(KEY=OUTRSQUARE, TYPE=DATA);
6092 20173 +   %if &ExportedCompChanged = Y %then %do;
6093 20174 +   %ChangeVariableSelectionFlag(Indata=&EM_USER_OUTR
     SQUARE);
6094 20175 +   %end;
6095 20176 +   %MakeDeltaCode(groupds=&EM_USER_OUTRSQUARE, Delta
     CodeFile=&EM_FILE_CDELTA_TRAIN);
6096 20177 +
6097 20178 +%mend score1;
6098 20179 +
6099 NOTE: %INCLUDE (level 1) ending.
6100 NOTE: Fileref TEMP has been deassigned.
6101
6102 NOTE: There were 7 observations read from the data set EMWS
     4.VARCLUS_OUTRSQUARE.
6103 NOTE: The data set EMWS4.VARCLUS_OUTRSQUARE has 7 observati
     ons and 9 variables.
6104 NOTE: PROCEDURE SORT used (Total process time):
```

```
6105        real time           0.01 seconds
6106        user cpu time       0.00 seconds
6107        system cpu time     0.01 seconds
6108        memory              159319.37k
6109        OS Memory           169600.00k
6110        Timestamp           07/01/2024 05:54:29 AM
6111        Step Count                       1  Switch Count  0
6112        Page Faults                      0
6113        Page Reclaims                    116
6114        Page Swaps                       0
6115        Voluntary Context Switches       32
6116        Involuntary Context Switches     0
6117        Block Input Operations           0
6118        Block Output Operations          264
6119
6120
6121
6122 NOTE: There were 7 observations read from the data set EMWS
     4.VARCLUS_OUTRSQUARE.
6123 NOTE: The data set EMWS4.VARCLUS_OUTRSQUARE has 7 observati
     ons and 9 variables.
6124 NOTE: DATA statement used (Total process time):
6125        real time           0.01 seconds
6126        user cpu time       0.00 seconds
6127        system cpu time     0.00 seconds
6128        memory              159319.37k
6129        OS Memory           169600.00k
6130        Timestamp           07/01/2024 05:54:29 AM
6131        Step Count                       1  Switch Count  0
6132        Page Faults                      0
6133        Page Reclaims                    131
6134        Page Swaps                       0
6135        Voluntary Context Switches       39
6136        Involuntary Context Switches     0
6137        Block Input Operations           288
6138        Block Output Operations          264
```

```
6139
6140
6141
6142 NOTE: The file X is:
6143       Filename=/home/u63452984/case-study-s2192852/Workspac
     es/EMWS4/VarClus/CDELTA_TRAIN.sas,
6144       Owner Name=u63452984,Group Name=oda,
6145       Access Permission=-rw-r--r--,
6146       Last Modified=07 January 2024 05:54:29
6147
6148 NOTE: 9 records were written to the file X.
6149       The minimum record length was 4.
6150       The maximum record length was 93.
6151 NOTE: There were 7 observations read from the data set EMWS
     4.VARCLUS_OUTRSQUARE.
6152 NOTE: DATA statement used (Total process time):
6153       real time           0.00 seconds
6154       user cpu time       0.00 seconds
6155       system cpu time     0.00 seconds
6156       memory              159319.37k
6157       OS Memory           169600.00k
6158       Timestamp           07/01/2024 05:54:29 AM
6159       Step Count                        1  Switch Count  0
6160       Page Faults                       0
6161       Page Reclaims                     67
6162       Page Swaps                        0
6163       Voluntary Context Switches        14
6164       Involuntary Context Switches      0
6165       Block Input Operations            288
6166       Block Output Operations           8
6167
6168
6169 NOTE: Fileref X has been deassigned.
6170
6171 NOTE: There were 1 observations read from the data set EMWS
     4.VARCLUS_VARCLUSMETA.
```

```
6172 NOTE: The data set EMWS4.VARCLUS_VARCLUSMETA has 1 observat
     ions and 5 variables.
6173 NOTE: DATA statement used (Total process time):
6174       real time           0.01 seconds
6175       user cpu time       0.01 seconds
6176       system cpu time     0.00 seconds
6177       memory              159319.37k
6178       OS Memory           169600.00k
6179       Timestamp           07/01/2024 05:54:29 AM
6180       Step Count                        1  Switch Count  0
6181       Page Faults                       0
6182       Page Reclaims                     127
6183       Page Swaps                        0
6184       Voluntary Context Switches        33
6185       Involuntary Context Switches      0
6186       Block Input Operations            0
6187       Block Output Operations           264
6188
6189
6190 20180  *-------------------------------------------------
     ---------*;
6191 20181  * End SCORE: VarClus;
6192 20182  *-------------------------------------------------
     ---------*;
6193 20183
6194
6195 20184  filename emflow "/home/u63452984/case-study-s2192852
     /Workspaces/EMWS4/VarClus/EMFLOWSCORE.sas";
6196 20185  *-------------------------------------------------
     ---------*;
6197 20186  * VarClus: Scoring DATA data;
6198 20187  *-------------------------------------------------
     ---------*;
6199 20188  data EMWS4.VarClus_TRAIN
6200 20189  / view=EMWS4.VarClus_TRAIN
6201 20190  ;
```

```
6202  20191  set EMWS4.Impt_TRAIN
6203  20192  ;
6204  20193  %inc emflow;
6205  NOTE: %INCLUDE (level 1) file EMFLOW is file /home/u6345298
      4/case-study-s2192852/Workspaces/EMWS4/VarClus/EMFLOWSCORE.
      sas.
6206  20194 +*** Begin Class Look-up, Standardization, Replacemen
      t ;
6207  20195 +
6208  20196 +*** Generate dummy variables for IMP_Churn ;
6209  20197 +label IMP_Churn0 = 'IMP_Churn=0' ;
6210  20198 +label IMP_Churn1 = 'IMP_Churn=1' ;
6211  20199 +if missing( IMP_Churn ) then do;
6212  20200 +   IMP_Churn0 = .;
6213  20201 +   IMP_Churn1 = .;
6214  20202 +end;
6215  20203 +else do;
6216  20204 +   length _dm12 $ 12; drop _dm12 ;
6217  20205 +   _dm12 = put( IMP_Churn , BEST12. );
6218  20206 +   %DMNORMIP( _dm12 )
6219  20207 +   if _dm12 = '0'  then do;
6220  20208 +      IMP_Churn0 = 1;
6221  20209 +      IMP_Churn1 = 0;
6222  20210 +   end;
6223  20211 +   else if _dm12 = '1'  then do;
6224  20212 +      IMP_Churn0 = 0;
6225  20213 +      IMP_Churn1 = 1;
6226  20214 +   end;
6227  20215 +   else do;
6228  20216 +      delete;
6229  20217 +   end;
6230  20218 +end;
6231  20219 +
6232  20220 +*** End Class Look-up, Standardization, Replacement
      ;
6233  20221 +
```

```
6234  20222 +
6235  20223 +/*-------------------------------------------------*
      /
6236  20224 +/* Varclus Score Code Begins*/
6237  20225 +/*-------------------------------------------------*
      /
6238  20226 +
6239  20227 +Clus1 = 0 ; /*---Cluster Component 1------ */
6240  20228 +Clus2 = 0 ; /*---Cluster Component 2------ */
6241  20229 +Clus2 = Clus2+0.17953445314506 * (Age - 44.950238009
      5203)/15.3394846010672;
6242  20230 +Clus2 = Clus2+0.51108967932882 * (IMP_TotalSpent - 4
      848.78184146293)/3750.10837018972;
6243  20231 +Clus2 = Clus2+0.50823468474197 * (TotalPurchases - 5
      .34157366294651)/4.05963182425794;
6244  20232 +Clus1 = Clus1+0.5 * (IMP_Churn0 - 0.7766310652426)/0
      .41651193657331;
6245  20233 +Clus1 = Clus1+-0.5 * (IMP_Churn1 - 0.22336893475739)
      /0.41651193657331;
6246  NOTE: %INCLUDE (level 1) ending.
6247  20234  run;
6248
6249  NOTE: DATA STEP view saved on file EMWS4.VARCLUS_TRAIN.
6250  NOTE: A stored DATA STEP view cannot run under a different
      operating system.
6251  NOTE: View EMWS4.IMPT_TRAIN.VIEW used (Total process time):
6252        real time           0.08 seconds
6253        user cpu time       0.02 seconds
6254        system cpu time     0.06 seconds
6255        memory              165042.00k
6256        OS Memory           176008.00k
6257        Timestamp           07/01/2024 05:54:29 AM
6258        Step Count                        1  Switch Count  7
6259        Page Faults                       0
6260        Page Reclaims                     33191
6261        Page Swaps                        0
```

```
6262        Voluntary Context Switches        29
6263        Involuntary Context Switches      0
6264        Block Input Operations            0
6265        Block Output Operations           264
6266
6267  NOTE: DATA statement used (Total process time):
6268        real time           0.09 seconds
6269        user cpu time       0.02 seconds
6270        system cpu time     0.07 seconds
6271        memory              165042.00k
6272        OS Memory           176008.00k
6273        Timestamp           07/01/2024 05:54:29 AM
6274        Step Count                        1  Switch Count  4
6275        Page Faults                       0
6276        Page Reclaims                     33261
6277        Page Swaps                        0
6278        Voluntary Context Switches        33
6279        Involuntary Context Switches      0
6280        Block Input Operations            0
6281        Block Output Operations           264
6282
6283
6284  20235  quit;
6285  20236  filename emflow;
6286  NOTE: Fileref EMFLOW has been deassigned.
6287
6288  20238  *-------------------------------------------------
     ---------*;
6289  20239  * VarClus: Computing metadata for TRAIN data;
6290  20240  *-------------------------------------------------
     ---------*;
6291
6292  NOTE: View EMWS4.VARCLUS_TRAIN.VIEW used (Total process tim
     e):
6293        real time           0.12 seconds
6294        user cpu time       0.03 seconds
```

```
6295        system cpu time     0.10 seconds
6296        memory              230156.62k
6297        OS Memory           241288.00k
6298        Timestamp           07/01/2024 05:54:30 AM
6299        Step Count                        1  Switch Count  9
6300        Page Faults                       0
6301        Page Reclaims                     49515
6302        Page Swaps                        0
6303        Voluntary Context Switches        34
6304        Involuntary Context Switches      1
6305        Block Input Operations            0
6306        Block Output Operations           0
6307
6308  NOTE: View EMWS4.VARCLUS_TRAIN.VIEW used (Total process tim
      e):
6309        real time           0.12 seconds
6310        user cpu time       0.04 seconds
6311        system cpu time     0.08 seconds
6312        memory              235225.43k
6313        OS Memory           245728.00k
6314        Timestamp           07/01/2024 05:54:30 AM
6315        Step Count                        1  Switch Count  9
6316        Page Faults                       0
6317        Page Reclaims                     49514
6318        Page Swaps                        0
6319        Voluntary Context Switches        33
6320        Involuntary Context Switches      0
6321        Block Input Operations            0
6322        Block Output Operations           0
6323
6324  *------------------------------------------------------------
      --*
6325  * Report Log
6326  Date:                07 January 2024
6327  Time:                05:54:31
6328  *------------------------------------------------------------
```

```
     --*
6329 20610  %let EMEXCEPTIONSTRING=;
6330 20611  *-------------------------------------------------
     ---------*;
6331 20612  * REPORT: VarClus;
6332 20613  *-------------------------------------------------
     ---------*;
6333 20614  %let EM_ACTION = REPORT;
6334 20615  %let syscc = 0;
6335 20616
6336 20617  %macro main;
6337 20618
6338 20619     filename temp catalog 'sashelp.emexpl.variableclu
     stering_macros.source';
6339 20620     %include temp;
6340 20621     filename temp catalog 'sashelp.emexpl.variableclu
     stering_macros2.source';
6341 20622     %include temp;
6342 20623     filename temp;
6343 20624
6344 20625    %SetProperties;
6345 20626
6346 20627     %if %upcase(&EM_ACTION) = CREATE %then %do;
6347 20628         filename temp catalog 'sashelp.emexpl.variabl
     eclustering_create.source';
6348 20629         %include temp;
6349 20630         filename temp;
6350 20631         %create;
6351 20632     %end;
6352 20633     %else
6353 20634     %if %upcase(&EM_ACTION) = TRAIN %then %do;
6354 20635          filename temp catalog 'sashelp.emexpl.variab
     leclustering_train.source';
6355 20636             %include temp;
6356 20637             filename temp;
6357 20638             %train;
```

```
6358  20639     %end;
6359  20640     %else
6360  20641     %if %upcase(&EM_ACTION) = SCORE %then %do;
6361  20642             filename temp catalog 'sashelp.emexpl.var
      iableclustering_score.source';
6362  20643             %include temp;
6363  20644             filename temp;
6364  20645             %score;
6365  20646     %end;
6366  20647     %else
6367  20648     %if %upcase(&EM_ACTION) = REPORT %then %do;
6368  20649             filename temp catalog 'sashelp.emexpl.var
      iableclustering_report.source';
6369  20650             %include temp;
6370  20651             filename temp;
6371  20652             %report;
6372  20653     %end;
6373  20654     /*
6374  20655     %if %upcase(&EM_ACTION) = OPENTESTTABLE %then %do
      ;
6375  20656         %put 'OPENING TABLE';
6376  20657     %end;
6377  20658     %if %upcase(&EM_ACTION) = CLOSETESTTABLE %then %d
      o;
6378  20659         %put 'CLOSE TABLE';
6379  20660     %end;
6380  20661     */
6381  20662  %mend main;
6382  20663  %main;
6383  NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMEXPL.V
      ARIABLECLUSTERING_MACROS.SOURCE.
6384  20664 +
6385  20665 +/* Initialize property macro variables */
6386  20666 +%macro SetProperties;
6387  20667 +   %em_checkmacro(name=EM_PROPERTY_MAXCLUS,       gl
      obal=Y, value=DEFAULT);
```

```
6388 20668 +   %em_checkmacro(name=EM_PROPERTY_HIDEVARIABLE,  gl
     obal=Y, value=Y);
6389 20669 +   %em_checkmacro(name=EM_PROPERTY_PRINTOPTION,   gl
     obal=Y, value=SHORT);
6390 20670 +   %em_checkmacro(name=EM_PROPERTY_CLUSSOURCE,    gl
     obal=Y, value=CORR);
6391 20671 +   %em_checkmacro(name=EM_PROPERTY_CLUSCOMP,      gl
     obal=Y, value=PRINCIPAL);
6392 20672 +   %em_checkmacro(name=EM_PROPERTY_CLUSHIERACHY,
       global=Y, value=Y);
6393 20673 +   %em_checkmacro(name=EM_PROPERTY_INCLUDECLASSVAR,
          global=Y, value=N);
6394 20674 +   %em_checkmacro(name=EM_PROPERTY_EXPORTEDCOMP,
       global=Y, value=CLUSTERCOMP);
6395 20675 +   %em_checkmacro(name=EM_PROPERTY_MAXEIGEN,
      global=Y, value=DEFAULT);
6396 20676 +   %em_checkmacro(name=EM_PROPERTY_PROPORTION,
     global=Y, value=DEFAULT);
6397 20677 +   %em_checkmacro(name=EM_PROPERTY_PRINTOPTION,
      global=Y, value=SHORT);
6398 20678 +   %em_checkmacro(name=EM_PROPERTY_TWOSTAGECLUS,
       global=Y, value=AUTO);
6399 20679 +   %em_checkmacro(name=EM_PROPERTY_SUPPRESSSAMPWARN,
            global=Y, value=N);
6400 20680 +
6401 20681 +%mend SetProperties;
6402 20682 +
6403 20683 +%Macro MakeDummyVariables(indata=,
6404 20684 +                          outvar=,
6405 20685 +                          outdata=,
6406 20686 +                          fileref=,
6407 20687 +                          recreatecmeta=N, /* option
     al */
6408 20688 +                          incmeta=,     /* optional
     */
6409 20689 +                          outcmeta=,    /* optional
```

```
     */
6410 20690 +                          ndummyvars=_ndummyvars
6411 20691 +                          );
6412 20692 +    %global &ndummyvars;
6413 20693 +
6414 20694 +    proc dmdb batch data=&indata out=_dmdbdat dmdbca
     t=_dmdbcat classout=_classout;;
6415 20695 +       class
6416 20696 +       %EM_BINARY_INPUT %EM_NOMINAL_INPUT %EM_ORDINA
     L_INPUT
6417 20697 +       %EM_BINARY_REJECTED %EM_NOMINAL_REJECTED %EM_
     ORDINAL_REJECTED
6418 20698 +    ;
6419 20699 +    run;
6420 20700 +    %let &ndummyvars = 0;
6421 20701 +    data _null_;
6422 20702 +    %let dsid = %sysfunc(open(work._classout));
6423 20703 +      %let &ndummyvars = %sysfunc(attrn(&dsid, NOBS)
     );
6424 20704 +    %let dsid = %sysfunc(close(&dsid));
6425 20705 +    run;
6426 20706 +
6427 20707 +    proc dmzip data=_dmdbdat dmdbcat=_dmdbcat;
6428 20708 +       input
6429 20709 +       %EM_BINARY_INPUT %EM_NOMINAL_INPUT %EM_ORDINA
     L_INPUT
6430 20710 +       %EM_BINARY_REJECTED %EM_NOMINAL_REJECTED %EM_
     ORDINAL_REJECTED
6431 20711 +        / level=nominal stdize=no;
6432 20712 +       make outvar = &outvar;
6433 20713 +       score data = &indata out =&outdata;
6434 20714 +       code  file=  "&fileref";
6435 20715 +    run;
6436 20716 +    %if &recreatecmeta eq Y %then %do;
6437 20717 +    proc contents data =&outvar out=_tmpds(keep=NAME
      LABEL);
```

```
6438 20718 +    data _tmpds;
6439 20719 +        set _tmpds;
6440 20720 +           ROLE = 'INPUT';
6441 20721 +           LEVEL = 'INTERVAL';
6442 20722 +           CREATOR='DMZIP';
6443 20723 +           if NAME = '_TYPE_' then delete;
6444 20724 +    run;
6445 20725 +    data &outcmeta;
6446 20726 +         set &incmeta _tmpds;
6447 20727 +    run;
6448 20728 +    %end;
6449 20729 +    proc datasets lib=work nolist;
6450 20730 +      delete  _dmdbdat _dmdbcat _classout
6451 20731 +    %if &recreatecmeta eq Y %then %do;
6452 20732 +    _tmpds
6453 20733 +    %end;
6454 20734 +    ;
6455 20735 +    quit;
6456 20736 +%Mend MakeDummyVariables;
6457 20737 +
6458 20738 +/*--- Determine Optimal Number of Cluster ----
6459 20739 +%macro FindClusNum(statds=, groupds=, minvariation=)
     ;
6460 20740 +   %global optnclus;
6461 20741 +   data varclus_tmp(drop=_NAME_);
6462 20742 +      set &statDs;
6463 20743 +      where _type_ ='PROPOR';
6464 20744 +   run;
6465 20745 +   proc sort data=varclus_tmp;
6466 20746 +      by _NCL_;
6467 20747 +   run;
6468 20748 +   proc transpose data=varclus_tmp out=varclus_tmp;
6469 20749 +      by _NCL_;
6470 20750 +      var %EM_INTERVAL_INPUT
6471 20751 +      %if &EM_PROPERTY_INCLUDECLASSVAR eq Y %then %d
     o;
```

```
6472  20752 +      %let dsid = %sysfunc(open(&EM_USER_OUTDUMMY));
6473  20753 +      %let nvar = %sysfunc(attrn(&dsid, NVAR));
6474  20754 +          %do i = 2 %to &nvar;
6475  20755 +          %let varname = %sysfunc(varname(&dsid, &i)
      );
6476  20756 +          &varname
6477  20757 +          %end;
6478  20758 +      %end;
6479  20759 +   ;
6480  20760 +   run;
6481  20761 +
6482  20762 +   %if &minVariation eq %then %do;
6483  20763 +       %let minVariation = &EM_PROPERTY_MINVARIATION
      ;
6484  20764 +   %end;
6485  20765 +   %if ^(0<&minVariation<100) %then %do;
6486  20766 +       %let minVariation = 90;
6487  20767 +   %end;
6488  20768 +
6489  20769 +   data _null_;
6490  20770 +      set varclus_tmp end=eof;
6491  20771 +      by _NCL_;
6492  20772 +      retain flag 0;
6493  20773 +      if first._ncl_ then flag=0;
6494  20774 +      if .<col1 < &minVariation then flag=1;
6495  20775 +      if last._ncl_ and ^flag then do;
6496  20776 +         call symput('OPTNCL', _ncl_);
6497  20777 +         stop;
6498  20778 +      end;
6499  20779 +     if eof then call symput('OPTNCL', _ncl_);
6500  20780 +   run;
6501  20781 +
6502  20782 +   %let optnclus = &OPTNCL;
6503  20783 +
6504  20784 +   data varclus_tmp(drop=_NCL_ _NAME_);
6505  20785 +      set &statDs;
```

```
6506 20786 +      where _type_ in('RSQUARED' 'GROUP') and _NCL_=
     &OPTNCL;
6507 20787 +   run;
6508 20788 +   proc sort data=varclus_tmp;
6509 20789 +      by _TYPE_;
6510 20790 +   run;
6511 20791 +   proc transpose data=varclus_tmp out=varclus_tmp;
6512 20792 +      by _TYPE_;
6513 20793 +   run;
6514 20794 +   proc sort data=varclus_tmp;
6515 20795 +      by _name_ _type_;
6516 20796 +   run;
6517 20797 +
6518 20798 +  proc transpose data=varclus_tmp out=&groupds;
6519 20799 +      by _NAME_;
6520 20800 +   run;
6521 20801 +   proc sort data=&groupDs(rename=(col1=Cluster col2
     =Rsquare _NAME_=VARIABLE));
6522 20802 +      by Cluster descending Rsquare;
6523 20803 +      where Cluster ne 0;
6524 20804 +   run;
6525 20805 +   proc datasets lib=work nolist mt=(DATA VIEW);
6526 20806 +      delete varclus_tmp;
6527 20807 +   run;
6528 20808 +   quit;
6529 20809 +%mend findClusNum;
6530 20810 +*/
6531 20811 +
6532 20812 +%macro getNclusfromTrain(inoutstat=, nc=);
6533 20813 +%global &nc;
6534 20814 +data _null_;
6535 20815 +    set &inoutstat end=eof;
6536 20816 +    if eof then do;
6537 20817 +    call symput("&nc", _ncl_);
6538 20818 +    end;
6539 20819 +run;
```

```
6540 20820 +%mend  getNclusfromTrain;
6541 20821 +
6542 20822 +%macro MakeDeltaCode(groupds=, outstatscore=, deltac
     odefile=);
6543 20823 +
6544 20824 +     *--- Build Code to Modify Metadata ---*;
6545 20825 +     filename X "&deltacodefile";
6546 20826 +     data _null_;
6547 20827 +        FILE X;
6548 20828 +        set &groupds end=eof;
6549 20829 +        /*by Cluster;*/
6550 20830 +         if _N_=1 then do;
6551 20831 +           %if &EM_PROPERTY_INCLUDECLASSVAR eq Y %th
     en %do;
6552 20832 +            put "if upcase(strip(ROLE)) ='INPUT' and
      upcase(strip(LEVEL)) ^='INTERVAL' then ROLE ='REJECTED' ;"
     ;
6553 20833 +           %end;
6554 20834 +           put "if upcase(strip(ROLE))='INPUT' and u
     pcase(strip(LEVEL))='INTERVAL' then do;";
6555 20835 +           put "if upcase(strip(NAME)) in (";
6556 20836 +        end;
6557 20837 +        if Strip(upcase(Selected)) eq 'YES' then do;
6558 20838 +           string = '"'!!trim(left(VARIABLE))!!'"';
6559 20839 +           put string;
6560 20840 +        end;
6561 20841 +        if eof then do;
6562 20842 +           put ') then ROLE="INPUT";';
6563 20843 +           put 'else ROLE="REJECTED";';
6564 20844 +           put 'end;';
6565 20845 +
6566 20846 +           %if %upcase(&EM_PROPERTY_HIDEVARIABLE) eq
      Y %then %do;
6567 20847 +             put 'if upcase(strip(ROLE)) = "REJECTED
     " then delete ;';
6568 20848 +           %end;
```

```
6569  20849 +        end;
6570  20850 +     run;
6571  20851 +     quit;
6572  20852 +
6573  20853 +     filename X;
6574  20854 +
6575  20855 +     quit;
6576  20856 +%mend MakeDeltaCode;
6577  20857 +
6578  20858 +%macro MakeVarClusCorrData(statds=, corrds=, corrplo
      tds= );
6579  20859 +    %if ^%sysfunc(exist(&statds)) %then %do;
6580  20860 +         %goto doendc;
6581  20861 +    %end;
6582  20862 +
6583  20863 +    data &corrds(drop=_TYPE_ _NCL_) ;
6584  20864 +       set &statds;
6585  20865 +       where _type_ eq 'CORR' ;
6586  20866 +    run ;
6587  20867 +    proc sort data=&corrds;
6588  20868 +       by _NAME_ ;
6589  20869 +    run ;
6590  20870 +    proc transpose data=&corrds out=&corrplotds name
      =_TMP_;
6591  20871 +      BY _NAME_ ;
6592  20872 +    run ;
6593  20873 +    data &corrplotds;
6594  20874 +       length _Y_ $100;
6595  20875 +       set &corrplotDs;
6596  20876 +       if _LABEL_ ne '' then _Y_=_LABEL_ ; else _Y_=
      _TMP_ ;
6597  20877 +    run ;
6598  20878 +    data varclus_match(rename=(_TMP_= _NAME_ _LABEL_
      =_X_));
6599  20879 +       set &corrplotds;
6600  20880 +       where _LABEL_ ne '' ;
```

```
6601  20881 +       keep _TMP_ _LABEL_ ;
6602  20882 +    run ;
6603  20883 +    data _null_;
6604  20884 +       nobs=0;
6605  20885 +       dsid = open('varclus_match');
6606  20886 +       if dsid then do;
6607  20887 +          nobs = attrn(dsid, 'NOBS');
6608  20888 +          dsid = close(dsid);
6609  20889 +       end;
6610  20890 +       call symput ('CORR_NOBS', nobs);
6611  20891 +    run;
6612  20892 +    %if &corr_nobs %then %do;
6613  20893 +        proc sort data=varclus_match;
6614  20894 +           by _name_;
6615  20895 +        run ;
6616  20896 +        proc sort data=&corrplotds;
6617  20897 +           by _name_;
6618  20898 +        run ;
6619  20899 +        data &corrplotds(keep= _X_ _Y_ col1 rename=(
     col1=Correlation)) ;
6620  20900 +           merge varclus_match &corrplotds;
6621  20901 +           by _NAME_ ;
6622  20902 +           if _X_ eq '' then _X_=_NAME_ ;
6623  20903 +           label _X_ = "%sysfunc(sasmsg(sashelp.dmin
     e, rpt_varclus_label_variable, noquote))";
6624  20904 +           label _Y_ = "%sysfunc(sasmsg(sashelp.dmin
     e, rpt_varclus_label_variable, noquote))";
6625  20905 +           label col1 = "%sysfunc(sasmsg(sashelp.dmi
     ne, rpt_correlation_vlabel, noquote))";
6626  20906 +
6627  20907 +       run ;
6628  20908 +    %end;
6629  20909 +    %else %do;
6630  20910 +        proc sort data=&corrplotds;
6631  20911 +           by _name_;
6632  20912 +        run ;
```

```
6633 20913 +        data &corrplotds(keep= _NAME_ _Y_ col1 renam
     e=(_NAME_=_X_ col1=Correlation)) ;
6634 20914 +           set &corrplotds;
6635 20915 +           label _NAME_ = "%sysfunc(sasmsg(sashelp.d
     mine, rpt_varclus_label_variable, noquote))" ;
6636 20916 +           label _Y_ = "%sysfunc(sasmsg(sashelp.dmin
     e, rpt_varclus_label_variable, noquote))" ;
6637 20917 +           label col1 = "%sysfunc(sasmsg(sashelp.dmi
     ne, rpt_correlation_vlabel, noquote))";
6638 20918 +
6639 20919 +        run ;
6640 20920 +    %end;
6641 20921 +    proc sort data=&corrplotds;
6642 20922 +       by _X_ _Y_;
6643 20923 +    run ;
6644 20924 +    proc datasets lib=work nolist mt=(DATA VIEW);
6645 20925 +       delete varclus_match;
6646 20926 +    run;
6647 20927 +    quit;
6648 20928 +
6649 20929 +%doendc:
6650 20930 +
6651 20931 +%mend MakeVarClusCorrData;
6652 20932 +
6653 20933 +%macro MakeStatPlotData(statds= , outstatplotds=);
6654 20934 +   %if %sysfunc(exist(&statds)) %then %do;
6655 20935 +
6656 20936 +       data varclus_tmp(drop=_NAME_ _NCL_) ;
6657 20937 +          set &statDs;
6658 20938 +          where _type_ in('MEAN', 'STD', 'N');
6659 20939 +       run ;
6660 20940 +       proc transpose data=varclus_tmp out=&outstatp
     lotds;
6661 20941 +          id _TYPE_ ;
6662 20942 +       run ;
6663 20943 +       data &outstatplotds;
```

```
6664  20944 +          set &outstatplotds(obs=1000);
6665  20945 +          label _name_= "%sysfunc(sasmsg(sashelp.dmi
      ne, rpt_varclus_label_name, noquote))";
6666  20946 +          label _label_="%sysfunc(sasmsg(sashelp.dmi
      ne, rpt_varclus_label_label, noquote))" ;
6667  20947 +          if MEAN ne 0 then SCALEDSTD= STD / MEAN ;
6668  20948 +          else SCALEDSTD= STD ;
6669  20949 +          label SCALEDSTD = "%sysfunc(sasmsg(sashelp
      .dmine, rpt_varclus_label_scaledstd, noquote))";
6670  20950 +       run ;
6671  20951 +       proc sort data=&outstatplotds;
6672  20952 +          by descending SCALEDSTD ;
6673  20953 +       run ;
6674  20954 +       proc datasets lib=work nolist mt=(DATA VIEW);
6675  20955 +          delete varclus_tmp;
6676  20956 +       run;
6677  20957 +       quit;
6678  20958 +    %end;
6679  20959 +
6680  20960 +%mend MakeStatPlotData;
6681  20961 +
6682  20962 +
6683  20963 +%macro CreateScoreCode(indata=, ncluscomp=, fileref=
      );
6684  20964 +     %EM_GETNAME(KEY=OUTSTATSCORE, type=DATA);
6685  20965 +    data &EM_USER_OUTSTATSCORE;
6686  20966 +          set &indata;
6687  20967 +          if (_TYPE_ in ('SCORE' 'MEAN' 'STD') and _
      NCL_ = &ncluscomp ) or (_TYPE_ in ('MEAN' 'STD'));
6688  20968 +          if _TYPE_ = 'MEAN' then _NAME_='MEAN';
6689  20969 +          if _TYPE_ = 'STD' then _NAME_='STD';
6690  20970 +          DROP _TYPE_ _NCL_;
6691  20971 +     run;
6692  20972 +
6693  20973 +     filename _file_  "&fileRef";
6694  20974 +
```

```
6695 20975 +     data _null_;
6696 20976 +        FILE _file_ MOD;
6697 20977 +        put ' ';
6698 20978 +        put '/*-----------------------------------
     ------------*/';
6699 20979 +        put '/* ' "%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_score_title_begin , noquote))" '*/';
6700 20980 +        put '/*-----------------------------------
     ------------*/';
6701 20981 +        put ' ';
6702 20982 +        %let dsid = %sysfunc(open(&EM_USER_OUTSTATSC
     ORE));
6703 20983 +
6704 20984 +        %let nvar = %sysfunc(attrn(&dsid, NVAR));
6705 20985 +        %let vn_name =%sysfunc(varnum(&dsid, _NAME_)
     );
6706 20986 +
6707 20987 +        %let k = 1;
6708 20988 +        %do %while(^%sysfunc(fetch(&dsid)));
6709 20989 +                %let _name = %sysfunc(getvarc(&dsid,
      &vn_name));
6710 20990 +                %if &k > 2 %then %do;
6711 20991 +                 %let cn = %eval(&k-2);
6712 20992 +                 put "&_name = 0 ; /*---" "%sysfunc(
     sasmsg(sashelp.dmine, rpt_varclus_score_cluscompnum, noquot
     e, &cn))"  "------ */";
6713 20993 +                %end;
6714 20994 +                %let k = %eval(&k+1);
6715 20995 +        %end;
6716 20996 +
6717 20997 +        %let rc = %sysfunc(rewind(&dsid));
6718 20998 +
6719 20999 +        %do i= 2 %to &nvar;
6720 21000 +            %let _varname =  %sysfunc(varname(&dsid,
      &i));
6721 21001 +            %do %while(^%sysfunc(fetch(&dsid)));
```

```
6722 21002 +                %let _name = %sysfunc(getvarc(&dsid,
      &vn_name));
6723 21003 +                %if &_name = MEAN %then
6724 21004 +                %let _mean = %sysfunc(getvarn(&dsid,
      &i));
6725 21005 +                %else %if &_name = STD %then
6726 21006 +                %let _std = %sysfunc(getvarn(&dsid,
     &i));
6727 21007 +                %else %do;
6728 21008 +                      %let coeff =  %sysfunc(getvarn
     (&dsid, &i));
6729 21009 +                      %let abscoeff = %sysfunc(abs(&
     coeff));
6730 21010 +                         %if &abscoeff >  0 %then %
     do;
6731 21011 +                          put "&_name = &_name+&coeff *
      (&_varname - &_mean)/&_std;";
6732 21012 +                             %end;
6733 21013 +                %end;
6734 21014 +             %end;
6735 21015 +             %let rc = %sysfunc(rewind(&dsid));
6736 21016 +          %end;
6737 21017 +
6738 21018 +        %let dsid= %sysfunc(close(&dsid));
6739 21019 +       run;
6740 21020 +%mend CreateScoreCode;
6741 21021 +
6742 21022 +
6743 21023 +
6744 21024 +/*-------------------------------------------------
     --------
6745 21025 +    Instead of using %MakeRSquareData,
6746 21026 +    %MakeVarClusResultTable at macro2.source is used
6747 21027 + +-------------------------------------------------
     --------*/
6748 21028 +
```

```
6749 21029 +
6750 21030 +%macro MakeRSquareData(indata=, inClusRSquare=, outd
     ata=, ncluster=);
6751 21031 +
6752 21032 +/* modifying  from ods rsquare = data */
6753 21033 +
6754 21034 +data &outdata(drop= ControlVar  NumberOfClusters Cur
     rentCluster);
6755 21035 +    Length Cluster $16;
6756 21036 +    length Variable $32;
6757 21037 +    Length VariableLabel $64;
6758 21038 +    set &indata; retain CurrentCluster;
6759 21039 +    if NumberOfClusters ^= &ncluster then delete;
6760 21040 +    if strip(Cluster) eq '' then Cluster = CurrentCl
     uster;
6761 21041 +    CurrentCluster = Cluster;
6762 21042 +   run;
6763 21043 +proc sort data =&outdata ;
6764 21044 +     by Cluster RsquareRatio;
6765 21045 +run;
6766 21046 +data _tmprsq(drop=index);
6767 21047 +     set &outdata; by Cluster;
6768 21048 +     if first.Cluster then do;
6769 21049 +     index = strip(scan(Cluster,2 ));
6770 21050 +     Variable = "Clus"||index;
6771 21051 +     VariableLabel = "Cluster Component "||index;
6772 21052 +     OwnCluster = 1;
6773 21053 +     NextClosest = .;
6774 21054 +     RsquareRatio = 0;
6775 21055 +     output;
6776 21056 +     end;
6777 21057 +run;
6778 21058 +
6779 21059 +proc sort data = _tmprsq ;
6780 21060 +     by Cluster RsquareRatio;
6781 21061 +run;
```

```
6782  21062 +data &outdata;
6783  21063 +     set &outdata _tmprsq;
6784  21064 +by Cluster;
6785  21065 +run;
6786  21066 +
6787  21067 +
6788  21068 +/* Just create the Selected variable with all YES */
6789  21069 +
6790  21070 +data &outdata;
6791  21071 +    set &outdata; by cluster;
6792  21072 +    length Selected $8;
6793  21073 +    Selected = 'YES';
6794  21074 +    label  OwnCluster = 'R-Sqaure with Cluster Compo
      nent';
6795  21075 +    label  NextClosest = 'R-Sqaure with Next Cluster
       Component';
6796  21076 +    rename OwnCluster = RSqWithClusterComp;
6797  21077 +    rename NextClosest = RSqWithNextClusComp;
6798  21078 +run;
6799  21079 +
6800  21080 +
6801  21081 +/* Selected = Y/N will be done %score section -----
6802  21082 +
6803  21083 +%if &EM_PROPERTY_EXPORTEDCOMP ne CLUSTERCOMP %then %
      do;
6804  21084 +data &outdata;
6805  21085 +    set &outdata; by cluster;
6806  21086 +    length Selected $8;
6807  21087 +    if first.Cluster then Selected = 'YES';
6808  21088 +    else Selected = 'NO';
6809  21089 +    label  OwnCluster = 'R-Sqaure with Cluster Compo
      nent';
6810  21090 +    label  NextClosest = 'R-Sqaure with Next Cluster
       Component';
6811  21091 +    rename OwnCluster = RSqWithClusterComp;
6812  21092 +    rename NextClosest = RSqWithNextClusComp;
```

```
6813  21093 +run;
6814  21094 +%end;
6815  21095 +%else %do;
6816  21096 +data &outdata;
6817  21097 +    set &outdata; by cluster;
6818  21098 +    if last.Cluster then Selected = 'YES';
6819  21099 +    else Selected = 'NO';
6820  21100 +    label  OwnCluster = 'R-Sqaure with Cluster Compo
      nent';
6821  21101 +    label  NextClosest = 'R-Sqaure with Next Cluster
       Component';
6822  21102 +    rename OwnCluster = RSqWithClusterComp;
6823  21103 +    rename NextClosest = RSqWithNextClusComp;
6824  21104 +run;
6825  21105 +%end;
6826  21106 +-----------------------------------------------
      -----------*/
6827  21107 +
6828  21108 +%if %sysfunc(exist(&inClusRSquare)) %then %do;
6829  21109 +/* to calculate NextClosestClusRsq */
6830  21110 +proc transpose data = &inClusRSquare  out=_clusRsq;
6831  21111 +      by cluster;
6832  21112 +      run;
6833  21113 +data _clusRsq;
6834  21114 +     set _clusRsq;
6835  21115 +     if strip(upcase(Cluster)) eq strip(upcase(_NAME
      _)) then delete;
6836  21116 +run;
6837  21117 +
6838  21118 +proc sort data=_clusRsq;
6839  21119 +    by cluster col1;
6840  21120 +    run;
6841  21121 +data _clusRsq(drop=_NAME_ _LABEL_);
6842  21122 +     set _clusRsq; by cluster;
6843  21123 +     if last.Cluster then output;
6844  21124 +     label  COL1 = 'R-Sqaure with Next Cluster Compo
```

```
     nent';
6845 21125 +     rename COL1 = RSqWithNextClusComp;
6846 21126 +     rename Cluster = Variable;
6847 21127 +     label  Cluster = "Variable";
6848 21128 +run;
6849 21129 +
6850 21130 +proc sort data =&outdata;
6851 21131 +     by Variable;
6852 21132 +run;
6853 21133 +data &outdata;
6854 21134 +    merge &outdata _clusRsq;
6855 21135 +    by Variable;
6856 21136 +run;
6857 21137 +proc sort data =&outdata;
6858 21138 +by Cluster  RsquareRatio;
6859 21139 +run;
6860 21140 +quit;
6861 21141 +%end;
6862 21142 +
6863 21143 +proc datasets lib = work nolist;
6864 21144 +     delete _tmprsq _clusRsq;
6865 21145 +     run;
6866 21146 +quit;
6867 21147 +
6868 21148 +%mend MakeRSquareData;
6869 21149 +
6870 21150 +
6871 21151 +/*-----------------------------------------------
     ----------------------*/
6872 21152 +
6873 21153 +
6874 21154 +
6875 21155 +%macro ModifyCorr(indata=,
6876 21156 +                  outdata=,
6877 21157 +                  rsquare = Y
6878 21158 +                  );
```

```
6879  21159 +    data corr_tmp;
6880  21160 +        set &indata;
6881  21161 +    run;
6882  21162 +     proc sql;
6883  21163 +           update &indata
6884  21164 +           set
6885  21165 +     %let dsid = %sysfunc(open(work.corr_tmp));
6886  21166 +     %let nvar = %sysfunc(attrn(&dsid, NVAR));
6887  21167 +          %do i = 4 %to &nvar;
6888  21168 +          %let _name = %sysfunc(varname(&dsid, &i));
6889  21169 +             %if &rsquare eq Y %then %let _name_md =
      &_name.**2;
6890  21170 +            %else  %let _name_md = &_name;
6891  21171 +            %if &i < &nvar %then %do;
6892  21172 +               &_name = 1- &_name_md ,
6893  21173 +            %end;
6894  21174 +            %else %do;
6895  21175 +               &_name = &_name_md where _TYPE_ conta
      ins 'CORR' ;
6896  21176 +            %end;
6897  21177 +          %end;
6898  21178 +      %let dsid= %sysfunc(close(&dsid));
6899  21179 +
6900  21180 +      select * from &indata;
6901  21181 +      run;
6902  21182 +      proc datasets lib = work nolist;
6903  21183 +           delete corr_tmp;
6904  21184 +      run;
6905  21185 +      quit;
6906  21186 +
6907  21187 + %mend ModifyCorr;
6908  21188 +
6909  21189 + %macro MakeClusStructCorrData(indata=,outdata=, ncl
      uster=, Rsquare=N);
6910  21190 +    data &outdata(drop= _NCL_  _TYPE_);
6911  21191 +       set &indata;
```

```
6912 21192 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
     E_) eq 'STRUCTUR') then delete;
6913 21193 +       rename _NAME_ = Cluster;
6914 21194 +       label _NAME_ = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_clustername,  noquote))";
6915 21195 +    run;
6916 21196 +    %if &RSquare eq Y %then %do;
6917 21197 +     data corr_tmp;
6918 21198 +        set &outdata;
6919 21199 +     run;
6920 21200 +
6921 21201 +     data &outdata(drop=i);
6922 21202 +          set &outdata;
6923 21203 +          %let dsid = %sysfunc(open(work.corr_tmp));
6924 21204 +          %let nvar = %sysfunc(attrn(&dsid, NVAR));
6925 21205 +          %do i = 2 %to &nvar;
6926 21206 +            %let _name = %sysfunc(varname(&dsid, &i)
     );
6927 21207 +            %let _name_md = &_name.**2;
6928 21208 +                &_name = &_name_md;
6929 21209 +          %end;
6930 21210 +      %let dsid= %sysfunc(close(&dsid));
6931 21211 +      run;
6932 21212 +      proc datasets lib = work nolist;
6933 21213 +           delete corr_tmp;
6934 21214 +      run;
6935 21215 +
6936 21216 +    %end;
6937 21217 +     quit;
6938 21218 +%mend MakeClusStructCorrData;
6939 21219 +
6940 21220 +%macro MakeInterClusCorrData(indata=, outdata=, nclu
     ster=, RSquare=N, makeplotds=N, plotds=);
6941 21221 +    data &outdata(drop= _NCL_  _TYPE_);
6942 21222 +       set &indata;
6943 21223 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
```

```
     E_) eq 'CCORR') then delete;
6944 21224 +       rename _NAME_ = Cluster;
6945 21225 +       label _NAME_ = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_clustername,  noquote))";
6946 21226 +    run;
6947 21227 +    data corr_tmp;
6948 21228 +        set &outdata;
6949 21229 +    run;
6950 21230 +
6951 21231 +    %let dsid = %sysfunc(open(work.corr_tmp));
6952 21232 +    %let nclus2= %eval(&ncluster+1);
6953 21233 +    data &outdata;
6954 21234 +         set &outdata;
6955 21235 +          %do i = 2 %to &nclus2;
6956 21236 +          %let i_1 = %eval(&i-1);
6957 21237 +            %let _name = %sysfunc(varname(&dsid, &i)
     );
6958 21238 +            %let _newName = Clus&i_1;
6959 21239 +                rename &_name = &_newName; ;
6960 21240 +                *label &_name ="Cluster &i_1";
6961 21241 +                label &_name = "%sysfunc(sasmsg(sash
     elp.dmine, rpt_varclus_label_clusternum, noquote,  &i_1))";
6962 21242 +          %end;
6963 21243 +          keep Cluster
6964 21244 +          %do i = 2 %to &nclus2;
6965 21245 +                %let _name = %sysfunc(varname(&dsid,
      &i));
6966 21246 +                &_name
6967 21247 +          %end;
6968 21248 +          ;
6969 21249 +    %let dsid= %sysfunc(close(&dsid));
6970 21250 +     run;
6971 21251 +     quit;
6972 21252 +
6973 21253 +    %if &RSquare eq Y %then %do;
6974 21254 +
```

```
6975  21255 +       data corr_tmp;
6976  21256 +        set &outdata;
6977  21257 +     run;
6978  21258 +
6979  21259 +     data &outdata(drop=i);
6980  21260 +          set &outdata;
6981  21261 +          %let dsid = %sysfunc(open(work.corr_tmp));
6982  21262 +          %let nvar = %sysfunc(attrn(&dsid, NVAR));
6983  21263 +          %do i = 2 %to &nvar;
6984  21264 +            %let _name = %sysfunc(varname(&dsid, &i)
      );
6985  21265 +            %let _name_md = &_name.**2;
6986  21266 +                &_name = &_name_md;
6987  21267 +          %end;
6988  21268 +      %let dsid= %sysfunc(close(&dsid));
6989  21269 +      run;
6990  21270 +    %end;
6991  21271 +
6992  21272 +    %if &makeplotds eq Y %then %do;
6993  21273 +     proc transpose data = &outdata
6994  21274 +          out=&plotds(drop=_LABEL_ rename=(_NAME_ =
      Y Cluster=X Col1= Correlation));
6995  21275 +          by cluster;
6996  21276 +     run;
6997  21277 +     data &plotds;
6998  21278 +          set  &plotds;
6999  21279 +          label x="%sysfunc(sasmsg(sashelp.dmine, rp
      t_varclus_label_cluster, noquote))";
7000  21280 +          label Y="%sysfunc(sasmsg(sashelp.dmine, rp
      t_varclus_label_cluster, noquote))";
7001  21281 +     run;
7002  21282 +     %end;
7003  21283 +     proc datasets lib = work nolist;
7004  21284 +           delete corr_tmp;
7005  21285 +     run;
7006  21286 +     quit;
```

```
7007 21287 +%mend MakeInterClusCorrData;
7008 21288 +
7009 21289 +
7010 21290 +%macro MakeClusConstellData(indata=, outlink=, outno
     de=);
7011 21291 +
7012 21292 +data &outlink(drop = Selected);
7013 21293 +     set &indata;
7014 21294 +     LINKID = _N_;
7015 21295 +     label LINKID = "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_linkid,  noquote))";
7016 21296 +     if strip(upcase(Cluster)) eq strip(upcase(Varia
     ble)) then Variable = ClosestCluster;
7017 21297 +run;
7018 21298 +data &outnode(keep=NODEID TYPE LABEL);
7019 21299 +    set &indata;
7020 21300 +    length TYPE $16;
7021 21301 +    rename Variable = NODEID;
7022 21302 +    label  Variable= "%sysfunc(sasmsg(sashelp.dmine,
      rpt_varclus_label_nodeidvar, noquote))";
7023 21303 +    if strip(upcase(Cluster)) eq strip(upcase(Variab
     le))
7024 21304 +    then  TYPE = "CLUSTER";
7025 21305 +    else  TYPE="VARIABLE";
7026 21306 +    label TYPE = "%sysfunc(sasmsg(sashelp.dmine, rpt
     _varclus_label_nodetype, noquote))";
7027 21307 +run;
7028 21308 +quit;
7029 21309 +%mend MakeClusConstellData;
7030 21310 +
7031 21311 +
7032 21312 +
7033 21313 +%macro MakeClusConstellData(indata=, outlink=, outno
     de=);
7034 21314 +
7035 21315 +data &outlink(drop = Selected);
```

```
7036 21316 +     set &indata;
7037 21317 +     LINKID = _N_;
7038 21318 +     label LINKID = "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_linkid,  noquote))";
7039 21319 +     if strip(upcase(Cluster)) eq strip(upcase(Varia
     ble)) then Variable = ClosestCluster;
7040 21320 +run;
7041 21321 +data &outnode(keep=NODEID TYPE LABEL);
7042 21322 +    set &indata;
7043 21323 +    length TYPE $16;
7044 21324 +    rename Variable = NODEID;
7045 21325 +    label  Variable= "%sysfunc(sasmsg(sashelp.dmine,
      rpt_varclus_label_nodeidvar, noquote))";
7046 21326 +    if strip(upcase(Cluster)) eq strip(upcase(Variab
     le))
7047 21327 +    then  TYPE = "CLUSTER";
7048 21328 +    else  TYPE="VARIABLE";
7049 21329 +    label TYPE = "%sysfunc(sasmsg(sashelp.dmine, rpt
     _varclus_label_nodetype, noquote))";
7050 21330 +run;
7051 21331 +quit;
7052 21332 +%mend MakeClusConstellData;
7053 21333 +
7054 21334 +
7055 21335 +/*--- This will work only when inds is not a view da
     ta -------
7056 21336 +
7057 21337 +%macro getNVarNObs(inds=, nvar=, nobs=);
7058 21338 +
7059 21339 +    %global &nvar;
7060 21340 +    %global &nobs;
7061 21341 +    data _null_;
7062 21342 +         dsid = open("&inds");
7063 21343 +         nv = attrn(dsid, 'NVAR');
7064 21344 +         no = attrn(dsid, 'NOBS');
7065 21345 +         dsid = close(dsid);
```

```
7066  21346 +         call symput("&nvar", nv);
7067  21347 +         call symput("&nobs", no);
7068  21348 +    run;
7069  21349 +    quit;
7070  21350 +%mend  getNVarNObs;
7071  21351 +
7072  21352 ++-------------------------------------------------
      ------------*/
7073  21353 +
7074  21354 +
7075  21355 +%macro getNVar(inds=, nvar=);
7076  21356 +    %global &nvar;
7077  21357 +    data _null_;
7078  21358 +         dsid = open("&inds");
7079  21359 +         nv = attrn(dsid, 'NVAR');
7080  21360 +         dsid = close(dsid);
7081  21361 +         call symput("&nvar", nv);
7082  21362 +    run;
7083  21363 +    quit;
7084  21364 +%mend  getNVar;
7085  21365 +
7086  21366 +
7087  21367 +
7088  21368 +%macro getNObs(inds=, nobs=);
7089  21369 +    %global &nobs;
7090  21370 +    data _null_;
7091  21371 +        set &inds end=eof;
7092  21372 +        if eof then call symput("&nobs", _N_);
7093  21373 +    run;
7094  21374 +    quit;
7095  21375 +%mend  getNObs;
7096  21376 +
7097  21377 +%Macro CreateVarclusMeta(trainnum=);
7098  21378 +   %EM_GETNAME(KEY=VARCLUSMETA, TYPE=DATA);
7099  21379 +    data &EM_USER_VARCLUSMETA;
7100  21380 +         length TrainNum 8.;
```

```
7101  21381 +         length NewTrain $8;
7102  21382 +         length NGCluster 8.;
7103  21383 +         length ExportedComp $16;
7104  21384 +         length HideVariable $8;
7105  21385 +         TrainNum = &trainnum;
7106  21386 +         NewTrain = "Y";
7107  21387 +         ExportedComp = "&EM_PROPERTY_EXPORTEDCOMP";
7108  21388 +         HideVariable = "&EM_PROPERTY_HIDEVARIABLE";
7109  21389 +         NGCluster = 0; /* zero means no twostage */
7110  21390 +   run;
7111  21391 +   quit;
7112  21392 +%mend CreateVarclusMeta;
7113  NOTE: %INCLUDE (level 1) ending.
7114  NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMEXPL.V
      ARIABLECLUSTERING_MACROS2.SOURCE.
7115  21393 +
7116  21394 +%macro MakeInterClusCorrData(indata=, outdata=, nclu
      ster=, globalclusid=, RSquare=N, makeplotds=N, plotds=);
7117  21395 +    data &outdata(drop= _NCL_  _TYPE_);
7118  21396 +       set &indata;
7119  21397 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
      E_) eq 'CCORR') then delete;
7120  21398 +       %if &globalclusid ne %then %do;
7121  21399 +       _NAME_ = "GC&globalclusid._"||upcase(_NAME_);
7122  21400 +       rename _NAME_ = Cluster;
7123  21401 +       %end;
7124  21402 +       %else %do;
7125  21403 +        _NAME_ = upcase(_NAME_);
7126  21404 +       rename _NAME_ = Cluster;
7127  21405 +       %end;
7128  21406 +       label _NAME_ = "%sysfunc(sasmsg(sashelp.dmine
      , rpt_varclus_label_clustername,  noquote))";
7129  21407 +    run;
7130  21408 +    data corr_tmp;
7131  21409 +        set &outdata;
7132  21410 +    run;
```

```
7133 21411 +
7134 21412 +    %let dsid = %sysfunc(open(work.corr_tmp));
7135 21413 +    %let nclus2= %eval(&ncluster+1);
7136 21414 +    data &outdata;
7137 21415 +         set &outdata;
7138 21416 +          %do i = 2 %to &nclus2;
7139 21417 +          %let i_1 = %eval(&i-1);
7140 21418 +            %let _name = %sysfunc(varname(&dsid, &i)
     );
7141 21419 +            %if &globalclusid ne %then
7142 21420 +                %do; %let _newName = GC&globalclusid
     ._CLUS&i_1;
7143 21421 +                     rename &_name = &_newName;
7144 21422 +                     *label &_name ="GC &globalclusi
     d : Cluster &i_1";
7145 21423 +                     label &_name = "%sysfunc(sasmsg
     (sashelp.dmine, rpt_varclus_label_gc_clusternum, noquote,
     &globalclusid, &i_1))";
7146 21424 +                %end;
7147 21425 +            %else
7148 21426 +                %do; %let _newName = CLUS&i_1;
7149 21427 +                     rename &_name = &_newName;
7150 21428 +                     *label &_name ="Cluster &i_1";
7151 21429 +                     label &_name = "%sysfunc(sasmsg
     (sashelp.dmine, rpt_varclus_label_clusternum, noquote,  &i_
     1))";
7152 21430 +                %end;
7153 21431 +          %end;
7154 21432 +          keep Cluster
7155 21433 +          %do i = 2 %to &nclus2;
7156 21434 +                %let _name = %sysfunc(varname(&dsid,
      &i));
7157 21435 +                &_name
7158 21436 +          %end;
7159 21437 +          ;
7160 21438 +    %let dsid= %sysfunc(close(&dsid));
```

```
7161  21439 +     run;
7162  21440 +     quit;
7163  21441 +
7164  21442 +    %if &RSquare eq Y %then %do;
7165  21443 +
7166  21444 +       data corr_tmp;
7167  21445 +        set &outdata;
7168  21446 +     run;
7169  21447 +
7170  21448 +     data &outdata;
7171  21449 +          set &outdata;
7172  21450 +          %let dsid = %sysfunc(open(work.corr_tmp));
7173  21451 +          %let nvar = %sysfunc(attrn(&dsid, NVAR));
7174  21452 +          %do i = 2 %to &nvar;
7175  21453 +            %let _name = %sysfunc(varname(&dsid, &i)
      );
7176  21454 +            %let _name_md = &_name.**2;
7177  21455 +                &_name = &_name_md;
7178  21456 +          %end;
7179  21457 +      %let dsid= %sysfunc(close(&dsid));
7180  21458 +      run;
7181  21459 +    %end;
7182  21460 +
7183  21461 +    %if &makeplotds eq Y %then %do;
7184  21462 +     proc transpose data = &outdata
7185  21463 +          out=&plotds(drop=_LABEL_ rename=(_NAME_ =
      Y Cluster=X Col1= Correlation));
7186  21464 +          by cluster;
7187  21465 +     run;
7188  21466 +     data &plotds;
7189  21467 +          set  &plotds;
7190  21468 +          label x="%sysfunc(sasmsg(sashelp.dmine, rp
      t_varclus_label_cluster, noquote))";
7191  21469 +          label Y="%sysfunc(sasmsg(sashelp.dmine, rp
      t_varclus_label_cluster, noquote))";
7192  21470 +          label  Correlation="%sysfunc(sasmsg(sashel
```

```
     p.dmine, rpt_correlation_vlabel, noquote))";
7193 21471 +     run;
7194 21472 +     %end;
7195 21473 +     proc datasets lib = work nolist;
7196 21474 +           delete corr_tmp;
7197 21475 +     run;
7198 21476 +     quit;
7199 21477 +%mend MakeInterClusCorrData;
7200 21478 +
7201 21479 +%macro MakeOwnRSquare(indata=, outdata=, ncluster=,
     globalclusid=);
7202 21480 +    data _tmpds(drop= _NCL_);
7203 21481 +       set &indata;
7204 21482 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
     E_) in ('GROUP','RSQUARED')) then delete;
7205 21483 +       %if &globalclusid ne %then %do;
7206 21484 +           _NAME_ = "GC&globalclusid.";
7207 21485 +       rename _NAME_ = Cluster;
7208 21486 +       %end;
7209 21487 +       %else %do;
7210 21488 +        _NAME_ = "CLUS";
7211 21489 +        rename _NAME_ = Cluster;
7212 21490 +       %end;
7213 21491 +       label _NAME_ = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_clustername,  noquote))";
7214 21492 +    run;
7215 21493 +    proc transpose data =_tmpds out =&outdata;
7216 21494 +    run;
7217 21495 +
7218 21496 +    data &outdata(drop=COL1);
7219 21497 +        %if &globalclusid ne %then %do;
7220 21498 +        length GCluster $16;
7221 21499 +        %end;
7222 21500 +        length Cluster $32;
7223 21501 +        length _NAME_ $32;
7224 21502 +         set &outdata;
```

```
7225 21503 +         _NAME_ = upcase(_NAME_);
7226 21504 +         rename _NAME_=Variable;
7227 21505 +         *label _NAME_="Variable";
7228 21506 +         label _NAME_ = "%sysfunc(sasmsg(sashelp.dmi
     ne, rpt_varclus_label_variable, noquote))" ;
7229 21507 +         label Cluster = "%sysfunc(sasmsg(sashelp.dm
     ine, rpt_varclus_label_cluster, noquote))" ;
7230 21508 +         label GCluster = "%sysfunc(sasmsg(sashelp.d
     mine, rpt_varclus_label_gcluster, noquote))" ;
7231 21509 +
7232 21510 +         %if &globalclusid ne %then %do;
7233 21511 +          GCluster = "GC&globalclusid";
7234 21512 +          Cluster = "GC&globalclusid._CLUS"||strip(C
     OL1);
7235 21513 +         %end;
7236 21514 +         %else %do;
7237 21515 +         Cluster = "CLUS"||strip(COL1);
7238 21516 +        %end;
7239 21517 +         rename COL2 = RSqWithOwnClusComp;
7240 21518 +         *label COL2 = "R-Square With Own Cluster Co
     mponent";
7241 21519 +         label COL2 = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_ownrsq, noquote))" ;
7242 21520 +
7243 21521 +   run;
7244 21522 +   proc sort data =&outdata;
7245 21523 +        by Cluster RSqWithOwnClusComp;
7246 21524 +   run;
7247 21525 +   proc datasets lib = work nolist;
7248 21526 +           delete _tmpds;
7249 21527 +   run;
7250 21528 +   quit;
7251 21529 +%mend MakeOwnRSquare;
7252 21530 +
7253 21531 +%macro MakeClusStructCorrData(indata=, outdata=, glo
     balclusid=, ncluster=, Rsquare=N);
```

```
7254 21532 +    data &outdata(drop= _NCL_  _TYPE_);
7255 21533 +       %if &globalclusid ne %then %do;
7256 21534 +        length GCluster $16;
7257 21535 +        %end;
7258 21536 +       set &indata;
7259 21537 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
     E_) eq 'STRUCTUR') then delete;
7260 21538 +       %if &globalclusid ne %then %do;
7261 21539 +         GCluster = "GC&globalclusid";
7262 21540 +         _NAME_ = "GC&globalclusid._"||upcase(_NAME_
     );
7263 21541 +         rename _NAME_ = Cluster;
7264 21542 +         label _NAME_ = "%sysfunc(sasmsg(sashelp.dmi
     ne, rpt_varclus_label_cluster, noquote))" ;
7265 21543 +         label GCluster = "%sysfunc(sasmsg(sashelp.d
     mine, rpt_varclus_label_gcluster, noquote))" ;
7266 21544 +
7267 21545 +       %end;
7268 21546 +       %else %do;
7269 21547 +         _NAME_ = upcase(_NAME_);
7270 21548 +         rename _NAME_ = Cluster;
7271 21549 +         label _NAME_ = "%sysfunc(sasmsg(sashelp.dmi
     ne, rpt_varclus_label_cluster, noquote))" ;
7272 21550 +         label GCluster = "%sysfunc(sasmsg(sashelp.d
     mine, rpt_varclus_label_gcluster, noquote))" ;
7273 21551 +
7274 21552 +       %end;
7275 21553 +     run;
7276 21554 +    %if &RSquare eq Y %then %do;
7277 21555 +     data corr_tmp;
7278 21556 +          set &outdata;
7279 21557 +     run;
7280 21558 +     %let istart = 2;
7281 21559 +     %if &globalclusid ne %then %let istart = 3;
7282 21560 +     data &outdata;
7283 21561 +          set &outdata;
```

```
7284  21562 +          %let dsid = %sysfunc(open(work.corr_tmp));
7285  21563 +          %let nvar = %sysfunc(attrn(&dsid, NVAR));
7286  21564 +          %do i =&istart %to &nvar;
7287  21565 +            %let _name = %sysfunc(varname(&dsid, &i)
      );
7288  21566 +            %let _name_md = &_name.**2;
7289  21567 +                &_name = &_name_md;
7290  21568 +          %end;
7291  21569 +       %let dsid= %sysfunc(close(&dsid));
7292  21570 +       run;
7293  21571 +       proc datasets lib = work nolist;
7294  21572 +            delete corr_tmp;
7295  21573 +       run;
7296  21574 +    %end;
7297  21575 +     quit;
7298  21576 +%mend MakeClusStructCorrData;
7299  21577 +
7300  21578 +/*
7301  21579 +%MakeClusStructCorrData(indata=playpen._outstat, out
      data=_structrsq , ncluster=7, Rsquare=Y);
7302  21580 +*/
7303  21581 +
7304  21582 +%macro FindNextClosestClusByVar(indata=, outdata=, g
      lobalclusid=, ncluster=);
7305  21583 +
7306  21584 +     /* The indata should be the outdata
7307  21585 +        from %MakeClusStructCorrData(indata=, outdat
      a=, ); */
7308  21586 +
7309  21587 +     proc sort data =&indata out=_tmpclusRsq;
7310  21588 +     by cluster;
7311  21589 +     run;
7312  21590 +
7313  21591 +     proc transpose data =_tmpclusRsq out=_tmpclusRs
      q;
7314  21592 +      by cluster;
```

```
7315  21593 +     run;
7316  21594 +
7317  21595 +     proc sort data=_tmpclusRsq;
7318  21596 +        by _NAME_ COL1;
7319  21597 +     run;
7320  21598 +
7321  21599 +     data _tmpclusRsq;
7322  21600 +         length _NAME_ $32;
7323  21601 +         set _tmpclusRsq; by _NAME_;
7324  21602 +          _NAME_ = upcase(_NAME_);
7325  21603 +          %if &ncluster ne 1 %then %do;
7326  21604 +             if last._NAME_ then delete;
7327  21605 +          %end;
7328  21606 +          %else %do;
7329  21607 +             COL1 = 0;
7330  21608 +          %end;
7331  21609 +     run;
7332  21610 +     /* need to sort again */
7333  21611 +     proc sort data=_tmpclusRsq;
7334  21612 +        by _NAME_ COL1;
7335  21613 +     run;
7336  21614 +
7337  21615 +     data &outdata;
7338  21616 +         set _tmpclusRsq; by _NAME_;
7339  21617 +         Cluster = upcase(Cluster);
7340  21618 +         if last._NAME_ then output;
7341  21619 +         *label  COL1 = 'R-Sqaure with Next Cluster
     Component';
7342  21620 +         label COL1 = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_nextrsq, noquote))" ;
7343  21621 +         rename COL1 = RSqWithNextClusComp;
7344  21622 +         Cluster = upcase(Cluster);
7345  21623 +         rename Cluster = ClosestCluster;
7346  21624 +         *label  Cluster = "Next Closest Cluster";
7347  21625 +         label Cluster = "%sysfunc(sasmsg(sashelp.dm
     ine, rpt_varclus_label_nextclus, noquote))" ;
```

```
7348  21626 +         rename _NAME_ = Variable;
7349  21627 +         label  _NAME_ = "%sysfunc(sasmsg(sashelp.dm
      ine, rpt_varclus_label_variable, noquote))" ;
7350  21628 +    run;
7351  21629 +
7352  21630 +
7353  21631 +    %if &globalclusid ne %then %do;
7354  21632 +    data &outdata;
7355  21633 +       length GCluster $16;
7356  21634 +       set &outdata;
7357  21635 +       GCluster = "GC&globalclusid";
7358  21636 +       run;
7359  21637 +    %end;
7360  21638 +    proc datasets lib = work nolist;
7361  21639 +           delete _tmpclusRsq;
7362  21640 +    run;
7363  21641 +    quit;
7364  21642 +%mend FindNextClosestClusByVar;
7365  21643 +
7366  21644 +
7367  21645 +%macro FindNextClosestClusByCluster(indata=, outdata
      =, globalclusid=, ncluster=);
7368  21646 +     /* The indata should be the outdata from %MakeI
      nterClusCorrData(indata=, outdata=, ); */
7369  21647 +     proc sort data =&indata out=_tmpclusRsq;
7370  21648 +     by cluster;
7371  21649 +     run;
7372  21650 +     proc transpose data =_tmpclusRsq out=_tmpclusRs
      q;
7373  21651 +     by cluster;
7374  21652 +     run;
7375  21653 +     proc sort data=_tmpclusRsq;
7376  21654 +        by _NAME_ col1;
7377  21655 +     run;
7378  21656 +     data _tmpclusRsq;
7379  21657 +         length _NAME_ $32;
```

```
7380  21658 +         set _tmpclusRsq; by _NAME_;
7381  21659 +          _NAME_ = upcase(_NAME_);
7382  21660 +          %if &ncluster ne 1 %then %do;
7383  21661 +             if last._NAME_ then delete;
7384  21662 +          %end;
7385  21663 +          %else %do;
7386  21664 +             COL1 = 0;
7387  21665 +          %end;
7388  21666 +     run;
7389  21667 +     data &outdata;
7390  21668 +         set _tmpclusRsq; by _NAME_;
7391  21669 +         Cluster = upcase(Cluster);
7392  21670 +         if last._NAME_ then output;
7393  21671 +         *label  COL1 = 'R-Sqaure with Next Cluster
      Component';
7394  21672 +         label COL1 = "%sysfunc(sasmsg(sashelp.dmine
      , rpt_varclus_label_nextrsq, noquote))" ;
7395  21673 +         rename COL1 = RSqWithNextClusComp;
7396  21674 +         Cluster = upcase(Cluster);
7397  21675 +         rename Cluster = ClosestCluster;
7398  21676 +         *label  Cluster = "Next Closest Cluster";
7399  21677 +         label Cluster = "%sysfunc(sasmsg(sashelp.dm
      ine, rpt_varclus_label_nextclus, noquote))" ;
7400  21678 +         rename _NAME_ = Variable;
7401  21679 +         *label  _NAME_ = "Variable";
7402  21680 +         label  _NAME_ = "%sysfunc(sasmsg(sashelp.dm
      ine, rpt_varclus_label_variable, noquote))" ;
7403  21681 +
7404  21682 +    run;
7405  21683 +    %if &globalclusid ne %then %do;
7406  21684 +    data &outdata;
7407  21685 +       length GCluster $16;
7408  21686 +       set &outdata;
7409  21687 +       GCluster = "GC&globalclusid";
7410  21688 +       run;
7411  21689 +    %end;
```

```
7412 21690 +
7413 21691 +    proc datasets lib = work nolist;
7414 21692 +           delete _tmpclusRsq;
7415 21693 +    run;
7416 21694 +
7417 21695 +    quit;
7418 21696 +%mend FindNextClosestClusByCluster;
7419 21697 +
7420 21698 +%macro MakeVarClusResultTable(indata1=, indata2=, in
     data3=, outdata=, globalclusid=, ncluster=, selectedcomp=cl
     ustercomp);
7421 21699 +/*----
7422 21700 +  indata1=_ownRsq, indata2=_nextVarRsq, indata3=_nex
     tClusRSq,
7423 21701 +-----------*/
7424 21702 +
7425 21703 +proc sort data =&indata1;
7426 21704 +    by Variable;
7427 21705 +run;
7428 21706 +proc sort data =&indata2;
7429 21707 +    by Variable;
7430 21708 +run;
7431 21709 +data &outdata;
7432 21710 +    merge &indata1 &indata2;
7433 21711 +    by Variable;
7434 21712 +    length Type $16;
7435 21713 +    Type = 'Variable';
7436 21714 +    *label Type ='Type';
7437 21715 +    label Type = "%sysfunc(sasmsg(sashelp.dmine, rpt
     _varclus_label_type, noquote))" ;
7438 21716 +run;
7439 21717 +
7440 21718 +
7441 21719 +data &indata3;
7442 21720 +    set  &indata3 ;
7443 21721 +    length RSqWithOwnClusComp 8.;
```

```
7444  21722 +    Cluster = Variable;
7445  21723 +    RSqWithOwnClusComp = 1;
7446  21724 +    *label RSqWithOwnClusComp = "R-Square With Own C
      luster Component";
7447  21725 +    label RSqWithOwnClusComp = "%sysfunc(sasmsg(sash
      elp.dmine, rpt_varclus_label_ownrsq, noquote))" ;
7448  21726 +    length Type $16;
7449  21727 +    Type = 'ClusterComp';
7450  21728 +    label Type = "%sysfunc(sasmsg(sashelp.dmine, rpt
      _varclus_label_type, noquote))" ;
7451  21729 +
7452  21730 +;
7453  21731 +run;
7454  21732 +
7455  21733 +proc sort data=&outdata;
7456  21734 +    by Cluster;
7457  21735 +run;
7458  21736 +proc sort data =&indata3;
7459  21737 +    by Cluster;
7460  21738 +run;
7461  21739 +
7462  21740 +data &outdata;
7463  21741 +     set &outdata &indata3;
7464  21742 +     by Cluster;
7465  21743 +run;
7466  21744 +
7467  21745 +
7468  21746 +/* Create the Selected variable with all YES */
7469  21747 +
7470  21748 +data &outdata;
7471  21749 +     set &outdata;
7472  21750 +     length RsqRatio 8.;
7473  21751 +     length Selected $8;
7474  21752 +     *label RSqRatio = "1-R**2 Ratio";
7475  21753 +     label RSqRatio =  "%sysfunc(sasmsg(sashelp.dmin
      e, rpt_varclus_label_oneminusrsq, noquote))" ;
```

```
7476 21754 +     *label Selected = "Variable Selected";
7477 21755 +     label Selected = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_varselected, noquote))" ;
7478 21756 +     RsqRatio = (1-RSqWithOwnClusComp)/(1-RSqWithNex
     tClusComp);
7479 21757 +     Selected ='YES';
7480 21758 +     rename _LABEL_ = Label;
7481 21759 +     label _LABEL_ =  "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_label, noquote))" ;
7482 21760 +run;
7483 21761 +
7484 21762 +
7485 21763 +/*----  Selected = Y/N will be assigened at the %sco
     re                  -------+
7486 21764 +        Just create the Selected variable with all Y
     ES at the step above
7487 21765 + +-------------------------------------------------
     ----------------------------+
7488 21766 +
7489 21767 +proc sort data=&outdata;
7490 21768 +    by Cluster RsqRatio;
7491 21769 +run;
7492 21770 +
7493 21771 +%if &selectedcomp eq CLUSTERCOMP %then %do;
7494 21772 +data &outdata;
7495 21773 +     set &outdata; by Cluster;
7496 21774 +     length Selected $8;
7497 21775 +     label Selected = "Variable Selected";
7498 21776 +     if  first.Cluster then Selected ='Yes';
7499 21777 +     else Selected = 'No';
7500 21778 +    run;
7501 21779 +%end;
7502 21780 +%else %do;
7503 21781 +data &outdata(drop = _var _varchange);
7504 21782 +     set &outdata; retain _var 0; by Cluster;
7505 21783 +     length Selected $8;
```

```
7506 21784 +     label Selected = "Variable Selected";
7507 21785 +     if first.Cluster then  _varchange = 0;
7508 21786 +     else _varchange =1;
7509 21787 +     if _var ne _varchange then  Selected  = 'Yes';
7510 21788 +     else Selected = 'No';
7511 21789 +     if last.cluster then  _var = 0;
7512 21790 +     else _var = _varchange;
7513 21791 +run;
7514 21792 +%end;
7515 21793 +
7516 21794 +-------------------------------------------------
     ----------*/
7517 21795 +
7518 21796 +quit;
7519 21797 +%mend MakeVarClusResultTable;
7520 21798 +
7521 21799 +%Macro MakePlotDataFromCorrTable(indata=, outdata=,
     globalclusid=);
7522 21800 +     proc sort data =&indata;
7523 21801 +        by cluster;
7524 21802 +     run;
7525 21803 +     proc transpose data =&indata
7526 21804 +          out=&outdata(drop=_LABEL_ rename=(_NAME_ =
      Y Cluster=X Col1= Correlation));
7527 21805 +          by cluster;
7528 21806 +     run;
7529 21807 +     data &outdata;
7530 21808 +          set  &outdata;
7531 21809 +          label x= "%sysfunc(sasmsg(sashelp.dmine, r
     pt_varclus_label_cluster, noquote))" ;
7532 21810 +          label Y= "%sysfunc(sasmsg(sashelp.dmine, r
     pt_varclus_label_cluster, noquote))" ;
7533 21811 +          label Correlation = "%sysfunc(sasmsg(sashe
     lp.dmine, rpt_correlation_vlabel, noquote))" ;
7534 21812 +     run;
7535 21813 +     %if &globalclusid ne %then %do;
```

```
7536 21814 +     data &outdata;
7537 21815 +          Length GCluster $16;
7538 21816 +          label GCluster = "%sysfunc(sasmsg(sashelp.
     dmine, rpt_varclus_label_gcluster, noquote))" ;
7539 21817 +          set  &outdata;
7540 21818 +          GCluster ="GC&globalclusid.";
7541 21819 +          run;
7542 21820 +     %end;
7543 21821 +
7544 21822 +%Mend MakePlotDataFromCorrTable;
7545 21823 +
7546 21824 +
7547 21825 +%macro MakeCorrelation(indata=,
7548 21826 +                       outstat=_tmpoutstat,
7549 21827 +                       corrmatrix=N,
7550 21828 +                       outcorr=_tmpoutcorr,
7551 21829 +                       includeclassvar=N,
7552 21830 +                       target=,
7553 21831 +                       freq=,
7554 21832 +                       weight=);
7555 21833 +   %if &target eq  %then %do;
7556 21834 +
7557 21835 +       proc varclus data=&indata outstat=&outstat hi
      maxclusters=1 noprint;
7558 21836 +            var %EM_INTERVAL_INPUT %EM_INTERVAL_REJE
     CTED
7559 21837 +       %if &includeclassvar eq Y %then %do;
7560 21838 +          %let dsid = %sysfunc(open(&EM_USER_OUTDUMM
     Y));
7561 21839 +          %let nvar = %sysfunc(attrn(&dsid, NVAR));
7562 21840 +          %do i = 2 %to &nvar;
7563 21841 +          %let varname = %sysfunc(varname(&dsid, &i)
     );
7564 21842 +          &varname
7565 21843 +          %end;
7566 21844 +       %end;
```

```
7567  21845 +       ;
7568  21846 +      %if &freq ne  %then %do;
7569  21847 +          freq &freq;
7570  21848 +      %end;
7571  21849 +      %if &weight ne  %then %do;
7572  21850 +          weight &weight;
7573  21851 +      %end;
7574  21852 +
7575  21853 +      run;
7576  21854 +      %if &corrmatrix eq Y %then %do;
7577  21855 +      data  &outcorr (drop = _NCL_ _TYPE_);
7578  21856 +           set  &outstat ;
7579  21857 +           if _TYPE_ ='CORR' then output;
7580  21858 +      run;
7581  21859 +      %end;
7582  21860 +    %end;
7583  21861 +    %else %do;
7584  21862 +       proc corr data=&indata outp=&outstat noprint;
7585  21863 +            var
7586  21864 +       %let dsid = %sysfunc(open(&indata));
7587  21865 +       %let nvar = %sysfunc(attrn(&dsid, NVAR));
7588  21866 +          %do i = 1 %to &nvar;
7589  21867 +              %let _name = %sysfunc(varname(&dsid, &
      i));
7590  21868 +              %if &_name ne &target %then;
7591  21869 +              &_name
7592  21870 +           %end;
7593  21871 +        %let dsid= %sysfunc(close(&dsid));
7594  21872 +        ;
7595  21873 +        with &target;
7596  21874 +        run;
7597  21875 +    %end;
7598  21876 +    quit;
7599  21877 +%mend MakeCorrelation;
7600  21878 +
7601  21879 +
```

```
7602  21880 +%macro MakeCorrelationDistance(indata=,
7603  21881 +                              outdata=,
7604  21882 +                              rsquare = N
7605  21883 +                              );
7606  21884 +    data corr_tmp;
7607  21885 +        set &indata;
7608  21886 +        if _N_ = 1 then do;
7609  21887 +           output;
7610  21888 +           stop;
7611  21889 +         end;
7612  21890 +    run;
7613  21891 +    %if &outdata ne  %then %let  _outdata = &outdata
      ;
7614  21892 +    %else %let _outdata = &indata;
7615  21893 +
7616  21894 +    data &_outdata;
7617  21895 +         set &indata;
7618  21896 +
7619  21897 +         %let dsid = %sysfunc(open(work.corr_tmp));
7620  21898 +         %let nvar = %sysfunc(attrn(&dsid, NVAR));
7621  21899 +          %do i = 2 %to &nvar;
7622  21900 +              %let _name = %sysfunc(varname(&dsid, &
      i));
7623  21901 +              %if &rsquare eq Y %then %let _name_md
      = &_name.**2;
7624  21902 +              %else  %let _name_md = &_name;
7625  21903 +              &_name = 1- &_name_md;
7626  21904 +          %end;
7627  21905 +      %let dsid= %sysfunc(close(&dsid));
7628  21906 +      run;
7629  21907 +      proc datasets lib = work nolist;
7630  21908 +           delete corr_tmp;
7631  21909 +      run;
7632  21910 +      quit;
7633  21911 +  %mend MakeCorrelationDistance;
7634  21912 +
```

```
7635 21913 +
7636 21914 +%macro UpdateOutStatCorrToDistance(indata=, /* indat
     a should be a outstat from proc varclus */
7637 21915 +                                    rsquare = N
7638 21916 +                                    );
7639 21917 +    data corr_tmp;
7640 21918 +        set &indata;
7641 21919 +    run;
7642 21920 +    proc sql noprint;
7643 21921 +           update &indata
7644 21922 +           set
7645 21923 +     %let dsid = %sysfunc(open(work.corr_tmp));
7646 21924 +     %let nvar = %sysfunc(attrn(&dsid, NVAR));
7647 21925 +          %do i = 4 %to &nvar;
7648 21926 +          %let _name = %sysfunc(varname(&dsid, &i));
7649 21927 +             %if &rsquare eq Y %then %let _name_md =
      &_name.**2;
7650 21928 +            %else  %let _name_md = &_name;
7651 21929 +            %if &i < &nvar %then %do;
7652 21930 +               &_name = 1- &_name_md ,
7653 21931 +            %end;
7654 21932 +            %else %do;
7655 21933 +               &_name = &_name_md where _TYPE_ eq 'C
     ORR' ;
7656 21934 +            %end;
7657 21935 +          %end;
7658 21936 +      %let dsid= %sysfunc(close(&dsid));
7659 21937 +
7660 21938 +      select * from &indata;
7661 21939 +      run;
7662 21940 +     data &indata( drop = _NCL_);
7663 21941 +           set &indata;
7664 21942 +           if _TYPE_ not in ('CORR', 'STD', 'N', 'ME
     AN') then delete;
7665 21943 +           if _TYPE_ ='CORR' then _TYPE_ ='DISTANCE'
     ;
```

```
7666  21944 +     run;
7667  21945 +     data &indata(DROP = _NCL_);
7668  21946 +          set &indata;
7669  21947 +          if _TYPE_ = 'CORR' then _TYPE_ ='DISTANCE
      ';
7670  21948 +          if _TYPE_ not in ('DISTANCE', 'N', 'STD',
       'MEAN') then delete;
7671  21949 +          rename _NAME_ = _VAR_;
7672  21950 +     run;
7673  21951 +     proc datasets lib = work nolist;
7674  21952 +          delete corr_tmp;
7675  21953 +     run;
7676  21954 +     quit;
7677  21955 + %mend UpdateOutStatCorrToDistance;
7678  21956 +
7679  21957 +
7680  21958 +%macro HierClusWithCorr(indata= ,
7681  21959 +                         ncluster=,
7682  21960 +                         method = Ward,
7683  21961 +                         outtree = _outtree,
7684  21962 +                         idvar =_VAR_,
7685  21963 +                         outdata=,
7686  21964 +                         rescore = N,
7687  21965 +                         newncluster=
7688  21966 +                         );
7689  21967 +      %global &newncluster;
7690  21968 +      %if &rescore ne Y %then %do;
7691  21969 +      proc cluster data=&indata(type=Distance where=
      (upcase(strip(_TYPE_)) = "DISTANCE"))
7692  21970 +                   method=&method outtree=&outtree n
      oprint;
7693  21971 +          id &idvar;
7694  21972 +      run;
7695  21973 +      %end;
7696  21974 +      proc tree data=&outtree nclusters = &ncluster
      out=&outdata noprint;
```

```
7697  21975 +      run;
7698  21976 +      /* ----- Check some variables like CL1, CL5...
      , remove them ----*/
7699  21977 +      proc contents data =&indata out=_outcontent(ke
      ep=NAME) noprint;
7700  21978 +      run;
7701  21979 +      data _outcontent;
7702  21980 +          set _outcontent;
7703  21981 +          if NAME in ('_TYPE_' , '_VAR_') then delet
      e;
7704  21982 +          index = 1;
7705  21983 +          rename NAME = _NAME_;
7706  21984 +      run;
7707  21985 +      proc sort data=_outcontent;
7708  21986 +          by _NAME_;
7709  21987 +      run;
7710  21988 +      proc sort data =&outdata;
7711  21989 +           by _NAME_;
7712  21990 +      run;
7713  21991 +      data &outdata(drop=index);
7714  21992 +          merge &outdata _outcontent;
7715  21993 +          by _NAME_;
7716  21994 +          if index = . then delete;
7717  21995 +      run;
7718  21996 +      /*-----------------------------------------
      ---------------*/
7719  21997 +      data &outdata;
7720  21998 +           length CLUSNAME $16;
7721  21999 +           set &outdata;
7722  22000 +           if CLUSTER > &ncluster then delete;
7723  22001 +           CLUSNAME='GC'||strip(CLUSTER);
7724  22002 +           *label CLUSNAME = "Cluster Name";
7725  22003 +           label CLUSNAME  = "%sysfunc(sasmsg(sashel
      p.dmine, rpt_varclus_label_clustername, noquote))" ;
7726  22004 +           rename _NAME_ = VARIABLE ;
7727  22005 +           *label _NAME_ = "Variable";
```

```
7728  22006 +           *label CLUSTER = "Cluster";
7729  22007 +           label _NAME_ ="%sysfunc(sasmsg(sashelp.dm
      ine, rpt_varclus_label_variable, noquote))" ;
7730  22008 +           label CLUSTER ="%sysfunc(sasmsg(sashelp.d
      mine, rpt_varclus_label_cluster, noquote))" ;
7731  22009 +      run;
7732  22010 +      proc sort data=&outdata out=&outdata;
7733  22011 +           by CLUSTER;
7734  22012 +      run;
7735  22013 +      proc means data =&outdata noprint;
7736  22014 +           output out=_meanout;
7737  22015 +      run;
7738  22016 +      data _null_;
7739  22017 +            set _meanout;
7740  22018 +            if strip(_STAT_) eq 'MAX' then do;
7741  22019 +            call symput("&newncluster", CLUSTER);
7742  22020 +            stop;
7743  22021 +            end;
7744  22022 +      run;
7745  22023 +
7746  22024 +      proc datasets lib = work nolist;
7747  22025 +           delete _outcontent _meanout;
7748  22026 +      run;
7749  22027 +      quit;
7750  22028 +%mend  HierClusWithCorr;
7751  22029 +
7752  22030 +%macro CreateScoreCode2(indata=, ncluscomp=, globalc
      lusid=, fileref=);
7753  22031 +
7754  22032 +     data _tmpindata;
7755  22033 +          set &indata;
7756  22034 +          if (_TYPE_ in ('SCORE' 'MEAN' 'STD') and _
      NCL_ = &ncluscomp ) or (_TYPE_ in ('MEAN' 'STD'));
7757  22035 +          if _TYPE_ = 'MEAN' then _NAME_='MEAN';
7758  22036 +          if _TYPE_ = 'STD' then _NAME_='STD';
7759  22037 +          if _TYPE_ = 'SCORE' then _NAME_=upcase("GC
```

```
     &globalclusid._"||_NAME_);
7760 22038 +          DROP _TYPE_ _NCL_;
7761 22039 +     run;
7762 22040 +
7763 22041 +     filename _file_  "&fileRef";
7764 22042 +
7765 22043 +     data _null_;
7766 22044 +        FILE _file_ MOD;
7767 22045 +        put ' ';
7768 22046 +        put "/*------------------------------------
     ------------*/";
7769 22047 +        put '/* ' "%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_score_title_gclus, noquote, &globalclusid))" '*/'
     ;
7770 22048 +        put "/*------------------------------------
     ------------*/";
7771 22049 +        put ' ';
7772 22050 +        %let dsid = %sysfunc(open(work._tmpindata));
7773 22051 +
7774 22052 +        %let nvar = %sysfunc(attrn(&dsid, NVAR));
7775 22053 +        %let vn_name =%sysfunc(varnum(&dsid, _NAME_)
     );
7776 22054 +        %let k = 1;
7777 22055 +        %do %while(^%sysfunc(fetch(&dsid)));
7778 22056 +               %let _name = %sysfunc(getvarc(&dsid,
     &vn_name));
7779 22057 +               %if &k > 2 %then %do;
7780 22058 +                %let cn = %eval(&k-2);
7781 22059 +                 put "&_name = 0 ; /*---" "%sysfunc(
     sasmsg(sashelp.dmine, rpt_varclus_score_gcluscompnum, noquo
     te, &globalclusid, &cn))" "------ */";
7782 22060 +                %end;
7783 22061 +                %let k = %eval(&k+1);
7784 22062 +
7785 22063 +        %end;
7786 22064 +        %let rc = %sysfunc(rewind(&dsid));
```

```
7787  22065 +
7788  22066 +        %do j= 2 %to &nvar;
7789  22067 +            %let _varname = %sysfunc(varname(&dsid,
      &j));
7790  22068 +            %do %while(^%sysfunc(fetch(&dsid)));
7791  22069 +                %let _name = %sysfunc(getvarc(&dsid,
       &vn_name));
7792  22070 +                %if &_name = MEAN %then
7793  22071 +                %let _mean = %sysfunc(getvarn(&dsid,
       &j));
7794  22072 +                %else %if &_name = STD %then
7795  22073 +                %let _std = %sysfunc(getvarn(&dsid,
      &j));
7796  22074 +                %else %do;
7797  22075 +                      %let coeff =  %sysfunc(getvarn
      (&dsid, &j));
7798  22076 +                      %let abscoeff = %sysfunc(abs(&
      coeff));
7799  22077 +                         %if &abscoeff >  0 %then %
      do;
7800  22078 +                          put "&_name = &_name+&coeff *
       (&_varname - &_mean)/&_std;";
7801  22079 +                           %end;
7802  22080 +                %end;
7803  22081 +            %end;
7804  22082 +            %let rc = %sysfunc(rewind(&dsid));
7805  22083 +         %end;
7806  22084 +
7807  22085 +        %let dsid= %sysfunc(close(&dsid));
7808  22086 +
7809  22087 +     run;
7810  22088 +
7811  22089 +     filename _file_;
7812  22090 +     proc datasets lib = work nolist;
7813  22091 +           delete _tmpindata;
7814  22092 +     run;
```

```
7815 22093 +     quit;
7816 22094 +%mend CreateScoreCode2;
7817 22095 +
7818 22096 +
7819 22097 +%macro MakeDeltaCode2(groupds=,deltacodefile=);
7820 22098 +
7821 22099 +     /*--- Build Code to Modify Metadata ---*/
7822 22100 +     filename X "&deltacodefile";
7823 22101 +     data _null_;
7824 22102 +        FILE X;
7825 22103 +        set &groupds end=eof;
7826 22104 +        if _N_=1 then do;
7827 22105 +           %if &EM_PROPERTY_INCLUDECLASSVAR eq Y %th
     en %do;
7828 22106 +            put "if upcase(strip(ROLE)) ='INPUT' and
      upcase(strip(LEVEL)) ^='INTERVAL' then ROLE ='REJECTED' ;"
     ;
7829 22107 +           %end;
7830 22108 +           put "if upcase(strip(ROLE))='INPUT' and u
     pcase(strip(LEVEL))='INTERVAL' then do;";
7831 22109 +           put "if upcase(strip(NAME)) in (";
7832 22110 +        end;
7833 22111 +        if Strip(upcase(Selected)) eq 'YES' then do;
7834 22112 +           string = '"'!!trim(left(VARIABLE))!!'"';
7835 22113 +           put string;
7836 22114 +        end;
7837 22115 +        if eof then do;
7838 22116 +           put ') then ROLE="INPUT";';
7839 22117 +           put 'else ROLE="REJECTED";';
7840 22118 +           put 'end;';
7841 22119 +
7842 22120 +           %if %upcase(&EM_PROPERTY_HIDEVARIABLE) eq
      Y %then %do;
7843 22121 +             put 'if upcase(strip(ROLE)) = "REJECTED
     " then delete ;';
7844 22122 +           %end;
```

```
7845 22123 +        end;
7846 22124 +     run;
7847 22125 +     quit;
7848 22126 +
7849 22127 +     filename X;
7850 22128 +     quit;
7851 22129 +%mend MakeDeltaCode2;
7852 22130 +
7853 22131 +%macro getInitialGClusterNumber(indata=, ninput=, nd
     ummy=0, div=100, ngc=);
7854 22132 +  %global &ngc;
7855 22133 +  data _null_;
7856 22134 +  %if &indata ne %then %do;
7857 22135 +      %let dsid = %sysfunc(open(&indata));
7858 22136 +           %let nvar = %sysfunc(attrn(&dsid, NVAR));
7859 22137 +      %let dsid = %sysfunc(close(&dsid));
7860 22138 +  %end;
7861 22139 +  %else %do;
7862 22140 +       %let nvar = %eval(&ninput+&ndummy); ;
7863 22141 +  %end;
7864 22142 +  %let numgc = %eval(&nvar/&div+2);
7865 22143 +  %let &ngc = &numgc;
7866 22144 +   run;
7867 22145 +  quit;
7868 22146 +%mend getInitialGClusterNumber;
7869 22147 +
7870 22148 +
7871 22149 +%macro MakeGobalConstellData(indata=, outlink=, outn
     ode=);
7872 22150 +data &outlink(drop = Selected);
7873 22151 +     set &indata;
7874 22152 +     LINKID = _N_;
7875 22153 +     label LINKID = "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_linkid, noquote))" ;
7876 22154 +run;
7877 22155 +data &outnode(keep=NODEID TYPE LABEL);
```

```
7878  22156 +    set &indata;
7879  22157 +    length TYPE $16;
7880  22158 +    rename VARIABLE = NODEID;
7881  22159 +    *label  CLUSNAME="Node ID";
7882  22160 +    label CLUSNAME= "%sysfunc(sasmsg(sashelp.dmine,
      rpt_varclus_label_nodeidvar, noquote))" ;
7883  22161 +    TYPE = "VARIABLE";
7884  22162 +    *label TYPE = "Node Type";
7885  22163 +    label TYPE =  "%sysfunc(sasmsg(sashelp.dmine, rp
      t_varclus_label_nodetype, noquote))" ;
7886  22164 +    run;
7887  22165 +data _tmp(keep=NODEID TYPE LABEL);
7888  22166 +    set &indata;
7889  22167 +    length TYPE $16;
7890  22168 +    rename CLUSNAME = NODEID;
7891  22169 +    label  CLUSNAME= "%sysfunc(sasmsg(sashelp.dmine,
       rpt_varclus_label_nodeidvar, noquote))" ;
7892  22170 +    TYPE = "GCLUSTER";
7893  22171 +    label TYPE = "%sysfunc(sasmsg(sashelp.dmine, rpt
      _varclus_label_nodetype, noquote))" ;
7894  22172 +    run;
7895  22173 +proc sort data=_tmp;
7896  22174 +    by NODEID;
7897  22175 +run;
7898  22176 +data _tmp;
7899  22177 +    set _tmp; by NODEID;
7900  22178 +    if first.NODEID then output;
7901  22179 +run;
7902  22180 +proc sort data=&outnode;
7903  22181 +    by NODEID;
7904  22182 +run;
7905  22183 +data  &outnode;
7906  22184 +    set _tmp &outnode;
7907  22185 +run;
7908  22186 +proc datasets lib = work nolist;
7909  22187 +     delete _tmp;
```

```
7910 22188 +run;
7911 22189 +quit;
7912 22190 +%mend MakeGobalConstellData;
7913 22191 +
7914 22192 +/* Make contellation plot data among GCLUSTERS */
7915 22193 +
7916 22194 +%Macro MakeGClusterConstData(indata=, inoutrsq=, out
     node=, outlink=);
7917 22195 +
7918 22196 +data &outlink(keep = _NAME_ _PARENT_ _LABEL_ LINKID)
     ;
7919 22197 +     set &indata;
7920 22198 +     LINKID = _N_;
7921 22199 +     if upcase(substr(strip(_NAME_),1, 2))="CL" then
      do;
7922 22200 +        _NAME_ = "ROOT"||upcase(substr(strip(_NAME_)
     ,5));
7923 22201 +     end;
7924 22202 +     if _PARENT_ ne " " and upcase(substr(strip(_PAR
     ENT_),1, 2))="CL" then do;
7925 22203 +        _PARENT_ = "ROOT"||upcase(substr(strip(_PARE
     NT_),5));
7926 22204 +     end;
7927 22205 +     if upcase(substr(strip(_LABEL_),1, 2))="CL" the
     n do;
7928 22206 +        _LABEL_ = "ROOT"||upcase(substr(strip(_LABEL
     _),5));
7929 22207 +    end;
7930 22208 +run;
7931 22209 +
7932 22210 +data _tmp_outrsquare;
7933 22211 +        set &inoutrsq;
7934 22212 +        if upcase(strip(TYPE)) = 'CLUSTERCOMP' then
     delete;
7935 22213 +run;
7936 22214 +
```

```
7937 22215 +proc freq data =_tmp_outrsquare noprint;
7938 22216 +         tables GCluster/out=_tmp_GCLUSFREQ(rename=(
     GCLUSTER=_NAME_));
7939 22217 +run;
7940 22218 +
7941 22219 +data &outnode(keep=_NAME_ TYPE LABEL);
7942 22220 +    set &outlink;
7943 22221 +    length TYPE $16;
7944 22222 +    length LABEL $100;
7945 22223 +    /*label CLUSNAME= "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_nodeidvar, noquote))" ;*/
7946 22224 +
7947 22225 +    if upcase(substr(strip(_NAME_),1, 2))='GC' then
     do;
7948 22226 +    TYPE = "GCLUSTER";
7949 22227 +    LABEL = "%sysfunc(sasmsg(sashelp.dmine, rpt_varc
     lus_label_gcluster, noquote)):"||_NAME_;
7950 22228 +    end;
7951 22229 +    else do;
7952 22230 +    TYPE= "ROOT";
7953 22231 +    LABEL= _NAME_;
7954 22232 +    end;
7955 22233 +    label TYPE  =  "%sysfunc(sasmsg(sashelp.dmine, r
     pt_varclus_label_nodetype, noquote))" ;
7956 22234 +    label LABEL =  "%sysfunc(sasmsg(sashelp.dmine, r
     pt_varclus_label_label, noquote))" ;
7957 22235 +    label _NAME_ =  "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_nodeidvar, noquote))" ;
7958 22236 +    run;
7959 22237 +
7960 22238 +proc sort data=&outnode;
7961 22239 +     by _NAME_;
7962 22240 +proc sort data=_tmp_GCLUSFREQ;
7963 22241 +     by _NAME_;
7964 22242 +run;
7965 22243 +
```

```
7966 22244 +data  &outnode;
7967 22245 +     merge &outnode _tmp_GCLUSFREQ; by _NAME_;
7968 22246 +     if COUNT=. then COUNT=1;
7969 22247 +run;
7970 22248 +
7971 22249 +proc datasets lib = work nolist;
7972 22250 +     delete _tmp_outrsquare _tmp_GCLUSFREQ;
7973 22251 +run;
7974 22252 +
7975 22253 +quit;
7976 22254 +%Mend MakeGClusterConstData;
7977 22255 +
7978 22256 +
7979 22257 +%macro CreateGClusterScoreCode(indata=,  globalclusi
     d=, fileref=);
7980 22258 +
7981 22259 +    data _gscoretmpds;
7982 22260 +          set &indata;
7983 22261 +          if (_TYPE_ in ('SCORE' 'MEAN' 'STD') and _
     NCL_ = 1 ) or (_TYPE_ in ('MEAN' 'STD'));
7984 22262 +          if _TYPE_ = 'MEAN' then _NAME_='MEAN';
7985 22263 +          if _TYPE_ = 'STD' then _NAME_='STD';
7986 22264 +          if _TYPE_ = 'SCORE' then _NAME_ = "GC"||st
     rip(&globalclusid);
7987 22265 +          DROP _TYPE_ _NCL_;
7988 22266 +     run;
7989 22267 +
7990 22268 +    /*  %let gscorefile =  %bquote(&EM_NODEDIR)&EM_D
     SEP.gclusterscore.sas;
7991 22269 +        GCluster Component &globalclusid ------ */
7992 22270 +
7993 22271 +    filename _file_  "&fileref";
7994 22272 +
7995 22273 +    data _null_;
7996 22274 +        %if &globalclusid eq 1 %then %do;
7997 22275 +          FILE _file_;
```

```
7998  22276 +        %end;
7999  22277 +        %else %do;
8000  22278 +          FILE _file_ MOD;
8001  22279 +        %end;
8002  22280 +
8003  22281 +        %let dsid = %sysfunc(open(work._gscoretmpds)
      );
8004  22282 +        %let nvar = %sysfunc(attrn(&dsid, NVAR));
8005  22283 +        %let vn_name =%sysfunc(varnum(&dsid, _NAME_)
      );
8006  22284 +
8007  22285 +        %let k = 1;
8008  22286 +        %do %while(^%sysfunc(fetch(&dsid)));
8009  22287 +                %let _name = %sysfunc(getvarc(&dsid,
       &vn_name));
8010  22288 +                %if &k > 2 %then %do;
8011  22289 +                put "&_name = 0 ; ";
8012  22290 +                %end;
8013  22291 +                %let k = %eval(&k+1);
8014  22292 +        %end;
8015  22293 +
8016  22294 +        %let rc = %sysfunc(rewind(&dsid));
8017  22295 +        %do i= 2 %to &nvar;
8018  22296 +            %let _varname =  %sysfunc(varname(&dsid,
       &i));
8019  22297 +            %do %while(^%sysfunc(fetch(&dsid)));
8020  22298 +                %let _name = %sysfunc(getvarc(&dsid,
       &vn_name));
8021  22299 +                %if &_name = MEAN %then
8022  22300 +                %let _mean = %sysfunc(getvarn(&dsid,
       &i));
8023  22301 +                %else %if &_name = STD %then
8024  22302 +                %let _std = %sysfunc(getvarn(&dsid,
      &i));
8025  22303 +                %else %do;
8026  22304 +                      %let coeff =  %sysfunc(getvarn
```

```
      (&dsid, &i));
8027  22305 +                     %let abscoeff = %sysfunc(abs(&
      coeff));
8028  22306 +                        %if &abscoeff >  0 %then %
      do;
8029  22307 +                         put "&_name = &_name+&coeff *
       (&_varname - &_mean)/&_std;";
8030  22308 +                            %end;
8031  22309 +                %end;
8032  22310 +             %end;
8033  22311 +             %let rc = %sysfunc(rewind(&dsid));
8034  22312 +
8035  22313 +          %end;
8036  22314 +
8037  22315 +         %let dsid= %sysfunc(close(&dsid));
8038  22316 +       run;
8039  22317 +
8040  22318 +
8041  22319 +       proc datasets lib=work nolist;
8042  22320 +            delete _gscoretmpds;
8043  22321 +       run;
8044  22322 +      quit;
8045  22323 +
8046  22324 +%mend CreateGClusterScoreCode;
8047  22325 +
8048  22326 +
8049  22327 +%macro MakeGClusterCorrelation(Indata=, ngcluster=,
      gscorecode=, outrsquare=);
8050  22328 +
8051  22329 +   %EM_REGISTER(KEY=GSCORE, TYPE=DATA);
8052  22330 +   %EM_GETNAME(KEY=GSCORE, TYPE=DATA);
8053  22331 +   %EM_REGISTER(KEY=GSCORESTAT, TYPE=DATA);
8054  22332 +   %EM_GETNAME(KEY=GSCORESTAT, TYPE=DATA);
8055  22333 +   %EM_REGISTER(KEY=GSCORETREE, TYPE=DATA);
8056  22334 +   %EM_GETNAME(KEY=GSCORETREE, TYPE=DATA);
8057  22335 +   %EM_REGISTER(KEY=GSCORECORR, TYPE=DATA);
```

```
8058  22336 +   %EM_GETNAME(KEY=GSCORECORR, TYPE=DATA);
8059  22337 +   %EM_REGISTER(KEY=GSCORECORRPLOT, TYPE=DATA);
8060  22338 +   %EM_GETNAME(KEY=GSCORECORRPLOT, TYPE=DATA);
8061  22339 +   %EM_REGISTER(KEY=GCLUSLINK, TYPE=DATA);
8062  22340 +   %EM_GETNAME(KEY=GCLUSLINK, TYPE=DATA);
8063  22341 +   %EM_REGISTER(KEY=GCLUSNODE, TYPE=DATA);
8064  22342 +   %EM_GETNAME(KEY=GCLUSNODE, TYPE=DATA);
8065  22343 +
8066  22344 +   filename gsfile "&gscorecode";
8067  22345 +
8068  22346 +   data &EM_USER_GSCORE;
8069  22347 +             set &indata;
8070  22348 +             %include  gsfile;
8071  22349 +        keep
8072  22350 +        %do i=1 %to &ngcluster;
8073  22351 +         %let gcvarname = GC&i;
8074  22352 +         &gcvarname
8075  22353 +        %end;
8076  22354 +     ;
8077  22355 +   run;
8078  22356 +
8079  22357 +   proc varclus data=&EM_USER_GSCORE outstat=&EM_USE
     R_GSCORESTAT outtree=&EM_USER_GSCORETREE
8080  22358 +    %if %upcase(&EM_PROPERTY_CLUSCOMP) eq CENTROID %
     then %do; centroid %end;
8081  22359 +    %if %upcase(&EM_PROPERTY_CLUSSOURCE) eq COV %the
     n %do; cov %end;
8082  22360 +    %if %upcase(&EM_PROPERTY_CLUSHIERACHY) eq Y %the
     n %do; hi %end;
8083  22361 +    noprint  ;
8084  22362 +     var
8085  22363 +       %do i=1 %to &ngcluster;
8086  22364 +         %let gcvarname = GC&i;
8087  22365 +         &gcvarname
8088  22366 +        %end;
8089  22367 +   ;
```

```
8090 22368 +   run;
8091 22369 +
8092 22370 +
8093 22371 +   %MakeVarClusCorrData(statds=&EM_USER_GSCORESTAT,
     corrds=&EM_USER_GSCORECORR, corrplotds=&EM_USER_GSCORECORRP
     LOT );
8094 22372 +   data &EM_USER_GSCORECORRPLOT ;
8095 22373 +        set &EM_USER_GSCORECORRPLOT;
8096 22374 +        rename _X_ = X;
8097 22375 +        rename _Y_ = Y;
8098 22376 +        label _X_ = "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_gcluster, noquote))" ;
8099 22377 +        label _Y_ = "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_gcluster, noquote))" ;
8100 22378 +   run;
8101 22379 +
8102 22380 +   %MakeGClusterConstData(indata=&EM_USER_GSCORETREE
     , inoutrsq=&outrsquare, outnode=&EM_USER_GCLUSNODE, outlink
     =&EM_USER_GCLUSLINK);
8103 22381 +
8104 22382 +   data &EM_USER_GSCORETREE;
8105 22383 +        length _NAME_ $32;
8106 22384 +        length _LABEL_ $100;
8107 22385 +        set &EM_USER_GSCORETREE(DROP=_LABEL_);
8108 22386 +        if upcase(substr(strip(_NAME_),1, 2))='GC' t
     hen do;
8109 22387 +         _LABEL_ = "%sysfunc(sasmsg(sashelp.dmine, r
     pt_varclus_label_gcluster, noquote)):"||_NAME_;
8110 22388 +        end;else do;
8111 22389 +         _LABEL_ = _NAME_;
8112 22390 +        end;
8113 22391 +       label _LABEL_ = "%sysfunc(sasmsg(sashelp.dmin
     e, rpt_varclus_label_variable, noquote))";
8114 22392 +
8115 22393 +   run;
8116 22394 +
```

```
8117  22395 +   quit;
8118  22396 +
8119  22397 +%mend MakeGClusterCorrelation;
8120  22398 +
8121  22399 +
8122  NOTE: %INCLUDE (level 1) ending.
8123  NOTE: Fileref TEMP has been deassigned.
8124  NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMEXPL.V
      ARIABLECLUSTERING_REPORT.SOURCE.
8125  22400 +%macro report;
8126  22401 +    %EM_GETNAME(key=VARCLUSMETA, type=DATA) ;
8127  22402 +
8128  22403 +    data _null_;
8129  22404 +         set &EM_USER_VARCLUSMETA;
8130  22405 +         if _N_=1 then
8131  22406 +         call symput('_trainnum', TrainNum);
8132  22407 +    run;
8133  22408 +   %if (&_trainnum = 1 ) or %upcase(&EM_PROPERTY_TWO
      STAGECLUS) = NO %then %do;
8134  22409 +           filename temp catalog 'sashelp.emexpl.var
      iableclustering_report1.source';
8135  22410 +           %include temp;
8136  22411 +           filename temp;
8137  22412 +           %report1;
8138  22413 +       %end;
8139  22414 +   %if (&_trainnum = 2 ) or %upcase(&EM_PROPERTY_TWO
      STAGECLUS) = YES %then %do;
8140  22415 +           filename temp catalog 'sashelp.emexpl.var
      iableclustering_report2.source';
8141  22416 +           %include temp;
8142  22417 +           filename temp;
8143  22418 +           %report2;
8144  22419 +    %end;
8145  22420 +
8146  22421 +   /* End of all runs, change the flag of newtrain t
      o N */
```

```
8147  22422 +    %EM_GETNAME(KEY=VARCLUSMETA, TYPE=DATA);
8148  22423 +    data &EM_USER_VARCLUSMETA;
8149  22424 +         set &EM_USER_VARCLUSMETA;
8150  22425 +         NewTrain = "N";
8151  22426 +    run;
8152  22427 +    quit;
8153  22428 +%mend report;
8154  22429 +
8155  22430 +
8156  NOTE: %INCLUDE (level 1) ending.
8157  NOTE: Fileref TEMP has been deassigned.
8158
8159  NOTE: Numeric values have been converted to character value
      s at the places given by: (Line):(Column).
8160        56786:166
8161  NOTE: There were 1 observations read from the data set EMWS
      4.VARCLUS_VARCLUSMETA.
8162  NOTE: DATA statement used (Total process time):
8163        real time           0.00 seconds
8164        user cpu time       0.00 seconds
8165        system cpu time     0.00 seconds
8166        memory              235225.43k
8167        OS Memory           245728.00k
8168        Timestamp           07/01/2024 05:54:30 AM
8169        Step Count                        1  Switch Count  0
8170        Page Faults                       0
8171        Page Reclaims                     62
8172        Page Swaps                        0
8173        Voluntary Context Switches        7
8174        Involuntary Context Switches      0
8175        Block Input Operations            288
8176        Block Output Operations           0
8177
8178
8179  NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMEXPL.V
      ARIABLECLUSTERING_REPORT1.SOURCE.
```

```
8180  22431 +%macro report1;
8181  22433 +   %EM_GETNAME(key=OUTRSQUARE, type=DATA) ;
8182  22434 +   %EM_GETNAME(key=OUTVARSEL, type=DATA) ;
8183  22435 +   %EM_GETNAME(key=CLUSFREQ, type=DATA) ;
8184  22437 +   data &EM_USER_OUTVARSEL;
8185  22438 +        set &EM_USER_OUTRSQUARE;
8186  22439 +        if upcase(strip(SELECTED)) = 'YES' then outp
      ut;
8187  22440 +   run;
8188  22442 +   data _tmp_outrsquare;
8189  22443 +        set &EM_USER_OUTRSQUARE;
8190  22444 +        if upcase(strip(TYPE)) = 'CLUSTERCOMP' then
      delete;
8191  22445 +   run;
8192  22447 +   proc freq data =_tmp_outrsquare noprint;
8193  22448 +        tables cluster/out=&EM_USER_CLUSFREQ sparse;
8194  22449 +   run;
8195  22451 +   proc datasets lib = work nolist;
8196  22452 +     delete _tmp_outrsquare;
8197  22453 +   run;
8198  22455 +   data &EM_USER_CLUSFREQ;
8199  22456 +       set &EM_USER_CLUSFREQ;
8200  22457 +       label
8201  22458 +           cluster = "%sysfunc(sasmsg(sashelp.dmine,
       rpt_varclus_label_cluster, NOQUOTE))"
8202  22459 +           count   = "%sysfunc(sasmsg(sashelp.dmine,
       rpt_varclus_label_freqcnt, NOQUOTE))"
8203  22460 +           percent = "%sysfunc(sasmsg(sashelp.dmine,
       rpt_varclus_label_freqpct, NOQUOTE))"
8204  22461 +       ;
8205  22462 +   run;
8206  22464 +   %EM_REPORT(key=OUTVARSEL, viewtype=DATA, block=MO
      DEL, autodisplay=Y, description=selectedvariables);
8207  22465 +   %EM_REPORT(key=CLUSFREQ, viewtype=DATA, block=MOD
      EL, autodisplay=Y, description=varfreqbycluster);
8208  22466 +   %EM_REPORT(key=OUTRSQUARE, viewtype=DATA, block=M
```

```
      ODEL, autodisplay=Y, description=varseltable);
8209  22468 +   %EM_REPORT(linkkey=OUTLINK, nodekey=OUTNODE, view
      type=Constellation,  autodisplay=Y, block=MODEL,  linkfrom=
      Cluster, linkto=Variable, linkid=LINKID, nodeid=NODEID, nod
      eshape=TYPE, description=clusterplot);
8210  22469 +   %EM_REPORT(KEY=OUTTREE, VIEWTYPE=DENDROGRAM,  aut
      odisplay=Y, block =MODEL, Name=NAME, PARENT=PARENT, HEIGHT=
      Proportion , TipText = Variable, description=dendrogram );
8211  22470 +   %EM_REPORT(key=OUTRSQUARE, view = 1, viewtype=SCA
      TTER, x=Cluster, y=RSqRatio, autodisplay=N, group=Selected,
       block=MODEL, description=rsqplot);
8212  22471 +   %EM_REPORT(view = 1, y=RsqWithOwnClusComp);
8213  22472 +   %EM_REPORT(view=1, y=RsqWithNextClusComp);
8214  22473 +   %EM_REPORT(key=OUTCORRPLOT, viewtype=HISTOGRAM, X
      =_X_,Y=_Y_, FREQ=correlation, autodisplay=N, block=MODEL, d
      escription=varcorr);
8215  22474 +   %EM_REPORT(key=OUTCLUSCORRPLOT, viewtype=HISTOGRA
      M, X=X,Y=Y, FREQ=Correlation, autodisplay=N, block=MODEL, d
      escription=cluscorr);
8216  22475 +   %EM_REPORT(key=OUTCLUSSTRUCT, view=2, viewtype=BA
      R, block=MODEL, x=Cluster, freq=_anynumeric_ , autodisplay=
      N,  description=corrstruct) ;
8217  22477 +   /*
8218  22478 +   %EM_REPORT(KEY=OUTTREE, VIEWTYPE=DENDROGRAM,  aut
      odisplay=N, block =Dendrogram, Name=_NAME_, Parent=_PARENT_
      , Height=_VAREXP_, TipText=Variable, description=totvar);
8219  22479 +   %EM_REPORT(key=OUTCORR, viewtype=DATA, block=MODE
      L,description=corrmatrix) ;
8220  22480 +   %EM_REPORT(key=OUTSTATPLOT, viewtype=DATA, block=
      MODEL, description=sumstat) ; */
8221  22483 +%mend report1;
8222  NOTE: %INCLUDE (level 1) ending.
8223  NOTE: Fileref TEMP has been deassigned.
8224
8225  NOTE: There were 7 observations read from the data set EMWS
      4.VARCLUS_OUTRSQUARE.
```

```
8226 NOTE: The data set EMWS4.VARCLUS_OUTVARSEL has 2 observatio
     ns and 9 variables.
8227 NOTE: DATA statement used (Total process time):
8228       real time           0.00 seconds
8229       user cpu time       0.00 seconds
8230       system cpu time     0.00 seconds
8231       memory              235225.43k
8232       OS Memory           245728.00k
8233       Timestamp           07/01/2024 05:54:30 AM
8234       Step Count                        1  Switch Count  0
8235       Page Faults                       0
8236       Page Reclaims                     127
8237       Page Swaps                        0
8238       Voluntary Context Switches        20
8239       Involuntary Context Switches      0
8240       Block Input Operations            0
8241       Block Output Operations           264
8242
8243
8244
8245 NOTE: There were 7 observations read from the data set EMWS
     4.VARCLUS_OUTRSQUARE.
8246 NOTE: The data set WORK._TMP_OUTRSQUARE has 5 observations
     and 9 variables.
8247 NOTE: DATA statement used (Total process time):
8248       real time           0.00 seconds
8249       user cpu time       0.00 seconds
8250       system cpu time     0.00 seconds
8251       memory              235225.43k
8252       OS Memory           245728.00k
8253       Timestamp           07/01/2024 05:54:30 AM
8254       Step Count                        1  Switch Count  0
8255       Page Faults                       0
8256       Page Reclaims                     127
8257       Page Swaps                        0
8258       Voluntary Context Switches        3
```

```
8259       Involuntary Context Switches      0
8260       Block Input Operations            0
8261       Block Output Operations           264
8262
8263
8264
8265 NOTE: There were 5 observations read from the data set WORK
     ._TMP_OUTRSQUARE.
8266 NOTE: The data set EMWS4.VARCLUS_CLUSFREQ has 2 observation
     s and 3 variables.
8267 NOTE: PROCEDURE FREQ used (Total process time):
8268       real time           0.00 seconds
8269       user cpu time       0.00 seconds
8270       system cpu time     0.00 seconds
8271       memory              235225.43k
8272       OS Memory           245728.00k
8273       Timestamp           07/01/2024 05:54:30 AM
8274       Step Count                        1  Switch Count  0
8275       Page Faults                       0
8276       Page Reclaims                     190
8277       Page Swaps                        0
8278       Voluntary Context Switches        15
8279       Involuntary Context Switches      0
8280       Block Input Operations            0
8281       Block Output Operations           528
8282
8283
8284
8285 NOTE: Deleting WORK._TMP_OUTRSQUARE (memtype=DATA).
8286
8287 NOTE: PROCEDURE DATASETS used (Total process time):
8288       real time           0.00 seconds
8289       user cpu time       0.00 seconds
8290       system cpu time     0.00 seconds
8291       memory              235225.43k
8292       OS Memory           245728.00k
```

```
8293        Timestamp           07/01/2024 05:54:30 AM
8294        Step Count                        1  Switch Count  0
8295        Page Faults                       0
8296        Page Reclaims                     51
8297        Page Swaps                        0
8298        Voluntary Context Switches        0
8299        Involuntary Context Switches      0
8300        Block Input Operations            0
8301        Block Output Operations           8
8302
8303
8304
8305  NOTE: There were 2 observations read from the data set EMWS
      4.VARCLUS_CLUSFREQ.
8306  NOTE: The data set EMWS4.VARCLUS_CLUSFREQ has 2 observation
      s and 3 variables.
8307  NOTE: DATA statement used (Total process time):
8308        real time           0.01 seconds
8309        user cpu time       0.00 seconds
8310        system cpu time     0.00 seconds
8311        memory              235225.43k
8312        OS Memory           245728.00k
8313        Timestamp           07/01/2024 05:54:30 AM
8314        Step Count                        1  Switch Count  0
8315        Page Faults                       0
8316        Page Reclaims                     471
8317        Page Swaps                        0
8318        Voluntary Context Switches        38
8319        Involuntary Context Switches      0
8320        Block Input Operations            288
8321        Block Output Operations           264
8322
8323
8324
8325  NOTE: The data set WORK.EM_USER_REPORT has 132 observations
       and 4 variables.
```

```
8326 NOTE: DATA statement used (Total process time):
8327       real time           0.02 seconds
8328       user cpu time       0.03 seconds
8329       system cpu time     0.01 seconds
8330       memory              235225.43k
8331       OS Memory           245728.00k
8332       Timestamp           07/01/2024 05:54:30 AM
8333       Step Count                        1  Switch Count  0
8334       Page Faults                       0
8335       Page Reclaims                     212
8336       Page Swaps                        0
8337       Voluntary Context Switches        0
8338       Involuntary Context Switches      0
8339       Block Input Operations            0
8340       Block Output Operations           264
8341
8342
8343
8344 NOTE: There were 132 observations read from the data set WO
     RK.EM_USER_REPORT.
8345 NOTE: The data set WORK.EM_USER_REPORT has 264 observations
      and 4 variables.
8346 NOTE: DATA statement used (Total process time):
8347       real time           0.02 seconds
8348       user cpu time       0.03 seconds
8349       system cpu time     0.00 seconds
8350       memory              235225.43k
8351       OS Memory           245728.00k
8352       Timestamp           07/01/2024 05:54:30 AM
8353       Step Count                        1  Switch Count  0
8354       Page Faults                       0
8355       Page Reclaims                     198
8356       Page Swaps                        0
8357       Voluntary Context Switches        0
8358       Involuntary Context Switches      0
8359       Block Input Operations            0
```

```
8360       Block Output Operations           264
8361
8362
8363
8364 NOTE: There were 264 observations read from the data set WO
     RK.EM_USER_REPORT.
8365 NOTE: The data set WORK.EM_USER_REPORT has 396 observations
      and 4 variables.
8366 NOTE: DATA statement used (Total process time):
8367       real time           0.02 seconds
8368       user cpu time       0.02 seconds
8369       system cpu time     0.00 seconds
8370       memory              235225.43k
8371       OS Memory           245728.00k
8372       Timestamp           07/01/2024 05:54:30 AM
8373       Step Count                        1  Switch Count  0
8374       Page Faults                       0
8375       Page Reclaims                     196
8376       Page Swaps                        0
8377       Voluntary Context Switches        0
8378       Involuntary Context Switches      0
8379       Block Input Operations            0
8380       Block Output Operations           520
8381
8382
8383
8384 NOTE: There were 396 observations read from the data set WO
     RK.EM_USER_REPORT.
8385 NOTE: The data set WORK.EM_USER_REPORT has 528 observations
      and 4 variables.
8386 NOTE: DATA statement used (Total process time):
8387       real time           0.02 seconds
8388       user cpu time       0.03 seconds
8389       system cpu time     0.00 seconds
8390       memory              235225.43k
8391       OS Memory           245728.00k
```

```
8392        Timestamp           07/01/2024 05:54:30 AM
8393        Step Count                        1  Switch Count  0
8394        Page Faults                       0
8395        Page Reclaims                     196
8396        Page Swaps                        0
8397        Voluntary Context Switches        0
8398        Involuntary Context Switches      0
8399        Block Input Operations            0
8400        Block Output Operations           520
8401
8402
8403
8404  NOTE: There were 528 observations read from the data set WO
      RK.EM_USER_REPORT.
8405  NOTE: The data set WORK.EM_USER_REPORT has 660 observations
       and 4 variables.
8406  NOTE: DATA statement used (Total process time):
8407        real time           0.02 seconds
8408        user cpu time       0.03 seconds
8409        system cpu time     0.00 seconds
8410        memory              235225.43k
8411        OS Memory           245728.00k
8412        Timestamp           07/01/2024 05:54:30 AM
8413        Step Count                        1  Switch Count  0
8414        Page Faults                       0
8415        Page Reclaims                     196
8416        Page Swaps                        0
8417        Voluntary Context Switches        0
8418        Involuntary Context Switches      0
8419        Block Input Operations            0
8420        Block Output Operations           776
8421
8422
8423
8424  NOTE: There were 660 observations read from the data set WO
      RK.EM_USER_REPORT.
```

```
8425 NOTE: The data set WORK.EM_USER_REPORT has 793 observations
      and 4 variables.
8426 NOTE: DATA statement used (Total process time):
8427       real time           0.02 seconds
8428       user cpu time       0.03 seconds
8429       system cpu time     0.00 seconds
8430       memory              235225.43k
8431       OS Memory           245728.00k
8432       Timestamp           07/01/2024 05:54:30 AM
8433       Step Count                        1  Switch Count  0
8434       Page Faults                       0
8435       Page Reclaims                     228
8436       Page Swaps                        0
8437       Voluntary Context Switches        0
8438       Involuntary Context Switches      0
8439       Block Input Operations            0
8440       Block Output Operations           776
8441
8442
8443
8444 NOTE: There were 793 observations read from the data set WO
     RK.EM_USER_REPORT.
8445 NOTE: The data set WORK.EM_USER_REPORT has 926 observations
      and 4 variables.
8446 NOTE: DATA statement used (Total process time):
8447       real time           0.02 seconds
8448       user cpu time       0.02 seconds
8449       system cpu time     0.01 seconds
8450       memory              235225.43k
8451       OS Memory           245728.00k
8452       Timestamp           07/01/2024 05:54:30 AM
8453       Step Count                        1  Switch Count  0
8454       Page Faults                       0
8455       Page Reclaims                     228
8456       Page Swaps                        0
8457       Voluntary Context Switches        0
```

```
8458        Involuntary Context Switches      0
8459        Block Input Operations            0
8460        Block Output Operations           1032
8461
8462
8463
8464  NOTE: There were 926 observations read from the data set WO
      RK.EM_USER_REPORT.
8465  NOTE: The data set WORK.EM_USER_REPORT has 1059 observation
      s and 4 variables.
8466  NOTE: DATA statement used (Total process time):
8467        real time           0.02 seconds
8468        user cpu time       0.03 seconds
8469        system cpu time     0.00 seconds
8470        memory              235225.43k
8471        OS Memory           245728.00k
8472        Timestamp           07/01/2024 05:54:30 AM
8473        Step Count                        1  Switch Count  0
8474        Page Faults                       0
8475        Page Reclaims                     260
8476        Page Swaps                        0
8477        Voluntary Context Switches        0
8478        Involuntary Context Switches      0
8479        Block Input Operations            0
8480        Block Output Operations           1032
8481
8482
8483
8484  NOTE: There were 1059 observations read from the data set W
      ORK.EM_USER_REPORT.
8485  NOTE: The data set WORK.EM_USER_REPORT has 1192 observation
      s and 4 variables.
8486  NOTE: DATA statement used (Total process time):
8487        real time           0.02 seconds
8488        user cpu time       0.03 seconds
8489        system cpu time     0.00 seconds
```

```
8490        memory              235225.43k
8491        OS Memory           245728.00k
8492        Timestamp           07/01/2024 05:54:30 AM
8493        Step Count                        1  Switch Count  0
8494        Page Faults                       0
8495        Page Reclaims                     260
8496        Page Swaps                        0
8497        Voluntary Context Switches        0
8498        Involuntary Context Switches      0
8499        Block Input Operations            0
8500        Block Output Operations           1288
8501
8502
8503
8504  NOTE: There were 1192 observations read from the data set W
      ORK.EM_USER_REPORT.
8505  NOTE: The data set WORK.EM_USER_REPORT has 1325 observation
      s and 4 variables.
8506  NOTE: DATA statement used (Total process time):
8507        real time           0.02 seconds
8508        user cpu time       0.03 seconds
8509        system cpu time     0.00 seconds
8510        memory              235225.43k
8511        OS Memory           245728.00k
8512        Timestamp           07/01/2024 05:54:30 AM
8513        Step Count                        1  Switch Count  0
8514        Page Faults                       0
8515        Page Reclaims                     292
8516        Page Swaps                        0
8517        Voluntary Context Switches        0
8518        Involuntary Context Switches      0
8519        Block Input Operations            0
8520        Block Output Operations           1288
8521
8522
8523
```

```
8524 NOTE: There were 1325 observations read from the data set W
     ORK.EM_USER_REPORT.
8525 NOTE: The data set WORK.EM_USER_REPORT has 1457 observation
     s and 4 variables.
8526 NOTE: DATA statement used (Total process time):
8527       real time           0.02 seconds
8528       user cpu time       0.03 seconds
8529       system cpu time     0.00 seconds
8530       memory              235225.43k
8531       OS Memory           245728.00k
8532       Timestamp           07/01/2024 05:54:30 AM
8533       Step Count                       1  Switch Count  0
8534       Page Faults                      0
8535       Page Reclaims                    323
8536       Page Swaps                       0
8537       Voluntary Context Switches       0
8538       Involuntary Context Switches     0
8539       Block Input Operations           0
8540       Block Output Operations          1544
8541
8542
8543
8544 NOTE: There were 1 observations read from the data set EMWS
     4.VARCLUS_VARCLUSMETA.
8545 NOTE: The data set EMWS4.VARCLUS_VARCLUSMETA has 1 observat
     ions and 5 variables.
8546 NOTE: DATA statement used (Total process time):
8547       real time           0.01 seconds
8548       user cpu time       0.00 seconds
8549       system cpu time     0.00 seconds
8550       memory              235225.43k
8551       OS Memory           245728.00k
8552       Timestamp           07/01/2024 05:54:30 AM
8553       Step Count                       1  Switch Count  0
8554       Page Faults                      0
8555       Page Reclaims                    127
```

```
8556        Page Swaps                          0
8557        Voluntary Context Switches          31
8558        Involuntary Context Switches        0
8559        Block Input Operations              0
8560        Block Output Operations             264
8561
8562
8563  22486  *-------------------------------------------------
      ---------*;
8564  22487  * End REPORT: VarClus;
8565  22488  *-------------------------------------------------
      ---------*;
8566  22489
8567
8568  22490  /* Reset EM Options */
8569  22491  options formchar="|----|+|---+=|-/\<>*";
8570  22492  options nocenter ls=256 ps=10000;
8571  22493  goptions reset=all device=GIF NODISPLAY;
8572
8573  22494  proc sort data=WORK.EM_USER_REPORT;
8574  22495  by ID VIEW;
8575  22496  run;
8576
8577  NOTE: There were 1457 observations read from the data set W
      ORK.EM_USER_REPORT.
8578  NOTE: The data set WORK.EM_USER_REPORT has 1457 observation
      s and 4 variables.
8579  NOTE: PROCEDURE SORT used (Total process time):
8580        real time           0.00 seconds
8581        user cpu time       0.00 seconds
8582        system cpu time     0.00 seconds
8583        memory              235225.43k
8584        OS Memory           245728.00k
8585        Timestamp           07/01/2024 05:54:30 AM
8586        Step Count                        1  Switch Count  0
8587        Page Faults                       0
```

```
8588        Page Reclaims                     358
8589        Page Swaps                        0
8590        Voluntary Context Switches        0
8591        Involuntary Context Switches      0
8592        Block Input Operations            0
8593        Block Output Operations           1544
8594
8595
```