```
 1 *------------------------------------------------------------
   --*
 2 User:                   u63452984
 3 Date:                   07 January 2024
 4 Time:                   05:54:50
 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:                   05:54:44
14 *------------------------------------------------------------
   --*
15 15241  proc freq data=EMWS4.VarClus2_VariableSet noprint;
16 15242  table ROLE*LEVEL/out=WORK.VarClus2META;
17 15243  run;
18 15244  proc print data=WORK.VarClus2META 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 15596  *------------------------------------------------------
   ---------*;
28 15597  * VarClus2: Generation of macros and macro variables
```

```
   ;
29 15598  * To see the code generated, set the EM_DEBUG macro
   variable to SOURCE or _ALL_;
30 15599  *-------------------------------------------------
   ---------*;
31
32 15600  %let EMEXCEPTIONSTRING=;
33 15601  *-------------------------------------------------
   ---------*;
34 15602  * TRAIN: VarClus2;
35 15603  *-------------------------------------------------
   ---------*;
36 15604  %let EM_ACTION = TRAIN;
37 15605  %let syscc = 0;
38 15606
39 15607  %macro main;
40 15608
41 15609     filename temp catalog 'sashelp.emexpl.variableclu
   stering_macros.source';
42 15610     %include temp;
43 15611     filename temp catalog 'sashelp.emexpl.variableclu
   stering_macros2.source';
44 15612     %include temp;
45 15613     filename temp;
46 15614
47 15615    %SetProperties;
48 15616
49 15617     %if %upcase(&EM_ACTION) = CREATE %then %do;
50 15618         filename temp catalog 'sashelp.emexpl.variabl
   eclustering_create.source';
51 15619         %include temp;
52 15620         filename temp;
53 15621         %create;
54 15622     %end;
55 15623     %else
56 15624     %if %upcase(&EM_ACTION) = TRAIN %then %do;
```

```
57 15625          filename temp catalog 'sashelp.emexpl.variab
   leclustering_train.source';
58 15626             %include temp;
59 15627             filename temp;
60 15628             %train;
61 15629     %end;
62 15630     %else
63 15631     %if %upcase(&EM_ACTION) = SCORE %then %do;
64 15632             filename temp catalog 'sashelp.emexpl.var
   iableclustering_score.source';
65 15633             %include temp;
66 15634             filename temp;
67 15635             %score;
68 15636     %end;
69 15637     %else
70 15638     %if %upcase(&EM_ACTION) = REPORT %then %do;
71 15639             filename temp catalog 'sashelp.emexpl.var
   iableclustering_report.source';
72 15640             %include temp;
73 15641             filename temp;
74 15642             %report;
75 15643     %end;
76 15644     /*
77 15645     %if %upcase(&EM_ACTION) = OPENTESTTABLE %then %do
   ;
78 15646         %put 'OPENING TABLE';
79 15647     %end;
80 15648     %if %upcase(&EM_ACTION) = CLOSETESTTABLE %then %d
   o;
81 15649         %put 'CLOSE TABLE';
82 15650     %end;
83 15651     */
84 15652  %mend main;
85 15653  %main;
86 NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMEXPL.V
   ARIABLECLUSTERING_MACROS.SOURCE.
```

```
 87 15654 +
 88 15655 +/* Initialize property macro variables */
 89 15656 +%macro SetProperties;
 90 15657 +   %em_checkmacro(name=EM_PROPERTY_MAXCLUS,       gl
    obal=Y, value=DEFAULT);
 91 15658 +   %em_checkmacro(name=EM_PROPERTY_HIDEVARIABLE,  gl
    obal=Y, value=Y);
 92 15659 +   %em_checkmacro(name=EM_PROPERTY_PRINTOPTION,   gl
    obal=Y, value=SHORT);
 93 15660 +   %em_checkmacro(name=EM_PROPERTY_CLUSSOURCE,    gl
    obal=Y, value=CORR);
 94 15661 +   %em_checkmacro(name=EM_PROPERTY_CLUSCOMP,      gl
    obal=Y, value=PRINCIPAL);
 95 15662 +   %em_checkmacro(name=EM_PROPERTY_CLUSHIERACHY,
      global=Y, value=Y);
 96 15663 +   %em_checkmacro(name=EM_PROPERTY_INCLUDECLASSVAR,
         global=Y, value=N);
 97 15664 +   %em_checkmacro(name=EM_PROPERTY_EXPORTEDCOMP,
      global=Y, value=CLUSTERCOMP);
 98 15665 +   %em_checkmacro(name=EM_PROPERTY_MAXEIGEN,
     global=Y, value=DEFAULT);
 99 15666 +   %em_checkmacro(name=EM_PROPERTY_PROPORTION,
    global=Y, value=DEFAULT);
100 15667 +   %em_checkmacro(name=EM_PROPERTY_PRINTOPTION,
     global=Y, value=SHORT);
101 15668 +   %em_checkmacro(name=EM_PROPERTY_TWOSTAGECLUS,
      global=Y, value=AUTO);
102 15669 +   %em_checkmacro(name=EM_PROPERTY_SUPPRESSSAMPWARN,
          global=Y, value=N);
103 15670 +
104 15671 +%mend SetProperties;
105 15672 +
106 15673 +%Macro MakeDummyVariables(indata=,
107 15674 +                         outvar=,
108 15675 +                         outdata=,
109 15676 +                         fileref=,
```

```
110  15677 +                          recreatecmeta=N, /* option
     al */
111  15678 +                          incmeta=,     /* optional
     */
112  15679 +                          outcmeta=,    /* optional
     */
113  15680 +                          ndummyvars=_ndummyvars
114  15681 +                          );
115  15682 +    %global &ndummyvars;
116  15683 +
117  15684 +    proc dmdb batch data=&indata out=_dmdbdat dmdbca
     t=_dmdbcat classout=_classout;;
118  15685 +       class
119  15686 +       %EM_BINARY_INPUT %EM_NOMINAL_INPUT %EM_ORDINA
     L_INPUT
120  15687 +       %EM_BINARY_REJECTED %EM_NOMINAL_REJECTED %EM_
     ORDINAL_REJECTED
121  15688 +    ;
122  15689 +    run;
123  15690 +    %let &ndummyvars = 0;
124  15691 +    data _null_;
125  15692 +    %let dsid = %sysfunc(open(work._classout));
126  15693 +      %let &ndummyvars = %sysfunc(attrn(&dsid, NOBS)
     );
127  15694 +    %let dsid = %sysfunc(close(&dsid));
128  15695 +    run;
129  15696 +
130  15697 +    proc dmzip data=_dmdbdat dmdbcat=_dmdbcat;
131  15698 +       input
132  15699 +       %EM_BINARY_INPUT %EM_NOMINAL_INPUT %EM_ORDINA
     L_INPUT
133  15700 +       %EM_BINARY_REJECTED %EM_NOMINAL_REJECTED %EM_
     ORDINAL_REJECTED
134  15701 +        / level=nominal stdize=no;
135  15702 +       make outvar = &outvar;
136  15703 +       score data = &indata out =&outdata;
```

```
137 15704 +       code  file=  "&fileref";
138 15705 +    run;
139 15706 +    %if &recreatecmeta eq Y %then %do;
140 15707 +    proc contents data =&outvar out=_tmpds(keep=NAME
     LABEL);
141 15708 +    data _tmpds;
142 15709 +        set _tmpds;
143 15710 +           ROLE = 'INPUT';
144 15711 +           LEVEL = 'INTERVAL';
145 15712 +           CREATOR='DMZIP';
146 15713 +           if NAME = '_TYPE_' then delete;
147 15714 +    run;
148 15715 +    data &outcmeta;
149 15716 +         set &incmeta _tmpds;
150 15717 +    run;
151 15718 +    %end;
152 15719 +    proc datasets lib=work nolist;
153 15720 +      delete  _dmdbdat _dmdbcat _classout
154 15721 +    %if &recreatecmeta eq Y %then %do;
155 15722 +    _tmpds
156 15723 +    %end;
157 15724 +    ;
158 15725 +    quit;
159 15726 +%Mend MakeDummyVariables;
160 15727 +
161 15728 +/*--- Determine Optimal Number of Cluster ----
162 15729 +%macro FindClusNum(statds=, groupds=, minvariation=)
    ;
163 15730 +   %global optnclus;
164 15731 +   data varclus_tmp(drop=_NAME_);
165 15732 +      set &statDs;
166 15733 +      where _type_ ='PROPOR';
167 15734 +   run;
168 15735 +   proc sort data=varclus_tmp;
169 15736 +      by _NCL_;
170 15737 +   run;
```

```
171  15738 +   proc transpose data=varclus_tmp out=varclus_tmp;
172  15739 +      by _NCL_;
173  15740 +      var %EM_INTERVAL_INPUT
174  15741 +      %if &EM_PROPERTY_INCLUDECLASSVAR eq Y %then %d
     o;
175  15742 +      %let dsid = %sysfunc(open(&EM_USER_OUTDUMMY));
176  15743 +      %let nvar = %sysfunc(attrn(&dsid, NVAR));
177  15744 +          %do i = 2 %to &nvar;
178  15745 +          %let varname = %sysfunc(varname(&dsid, &i)
     );
179  15746 +          &varname
180  15747 +          %end;
181  15748 +      %end;
182  15749 +   ;
183  15750 +   run;
184  15751 +
185  15752 +   %if &minVariation eq %then %do;
186  15753 +       %let minVariation = &EM_PROPERTY_MINVARIATION
     ;
187  15754 +   %end;
188  15755 +   %if ^(0<&minVariation<100) %then %do;
189  15756 +       %let minVariation = 90;
190  15757 +   %end;
191  15758 +
192  15759 +   data _null_;
193  15760 +      set varclus_tmp end=eof;
194  15761 +      by _NCL_;
195  15762 +      retain flag 0;
196  15763 +      if first._ncl_ then flag=0;
197  15764 +      if .<col1 < &minVariation then flag=1;
198  15765 +      if last._ncl_ and ^flag then do;
199  15766 +         call symput('OPTNCL', _ncl_);
200  15767 +         stop;
201  15768 +      end;
202  15769 +     if eof then call symput('OPTNCL', _ncl_);
203  15770 +   run;
```

```
204  15771 +
205  15772 +   %let optnclus = &OPTNCL;
206  15773 +
207  15774 +   data varclus_tmp(drop=_NCL_ _NAME_);
208  15775 +      set &statDs;
209  15776 +      where _type_ in('RSQUARED' 'GROUP') and _NCL_=
     &OPTNCL;
210  15777 +   run;
211  15778 +   proc sort data=varclus_tmp;
212  15779 +      by _TYPE_;
213  15780 +   run;
214  15781 +   proc transpose data=varclus_tmp out=varclus_tmp;
215  15782 +      by _TYPE_;
216  15783 +   run;
217  15784 +   proc sort data=varclus_tmp;
218  15785 +      by _name_ _type_;
219  15786 +   run;
220  15787 +
221  15788 +  proc transpose data=varclus_tmp out=&groupds;
222  15789 +      by _NAME_;
223  15790 +   run;
224  15791 +   proc sort data=&groupDs(rename=(col1=Cluster col2
     =Rsquare _NAME_=VARIABLE));
225  15792 +      by Cluster descending Rsquare;
226  15793 +      where Cluster ne 0;
227  15794 +   run;
228  15795 +   proc datasets lib=work nolist mt=(DATA VIEW);
229  15796 +      delete varclus_tmp;
230  15797 +   run;
231  15798 +   quit;
232  15799 +%mend findClusNum;
233  15800 +*/
234  15801 +
235  15802 +%macro getNclusfromTrain(inoutstat=, nc=);
236  15803 +%global &nc;
237  15804 +data _null_;
```

```
238 15805 +    set &inoutstat end=eof;
239 15806 +    if eof then do;
240 15807 +    call symput("&nc", _ncl_);
241 15808 +    end;
242 15809 +run;
243 15810 +%mend  getNclusfromTrain;
244 15811 +
245 15812 +%macro MakeDeltaCode(groupds=, outstatscore=, deltac
    odefile=);
246 15813 +
247 15814 +     *--- Build Code to Modify Metadata ---*;
248 15815 +     filename X "&deltacodefile";
249 15816 +     data _null_;
250 15817 +        FILE X;
251 15818 +        set &groupds end=eof;
252 15819 +        /*by Cluster;*/
253 15820 +         if _N_=1 then do;
254 15821 +           %if &EM_PROPERTY_INCLUDECLASSVAR eq Y %th
    en %do;
255 15822 +            put "if upcase(strip(ROLE)) ='INPUT' and
     upcase(strip(LEVEL)) ^='INTERVAL' then ROLE ='REJECTED' ;"
    ;
256 15823 +           %end;
257 15824 +           put "if upcase(strip(ROLE))='INPUT' and u
    pcase(strip(LEVEL))='INTERVAL' then do;";
258 15825 +           put "if upcase(strip(NAME)) in (";
259 15826 +        end;
260 15827 +        if Strip(upcase(Selected)) eq 'YES' then do;
261 15828 +           string = '"'!!trim(left(VARIABLE))!!'"';
262 15829 +           put string;
263 15830 +        end;
264 15831 +        if eof then do;
265 15832 +           put ') then ROLE="INPUT";';
266 15833 +           put 'else ROLE="REJECTED";';
267 15834 +           put 'end;';
268 15835 +
```

```
269  15836 +           %if %upcase(&EM_PROPERTY_HIDEVARIABLE) eq
      Y %then %do;
270  15837 +             put 'if upcase(strip(ROLE)) = "REJECTED
     " then delete ;';
271  15838 +           %end;
272  15839 +         end;
273  15840 +     run;
274  15841 +     quit;
275  15842 +
276  15843 +     filename X;
277  15844 +
278  15845 +     quit;
279  15846 +%mend MakeDeltaCode;
280  15847 +
281  15848 +%macro MakeVarClusCorrData(statds=, corrds=, corrplo
     tds= );
282  15849 +    %if ^%sysfunc(exist(&statds)) %then %do;
283  15850 +         %goto doendc;
284  15851 +    %end;
285  15852 +
286  15853 +    data &corrds(drop=_TYPE_ _NCL_) ;
287  15854 +       set &statds;
288  15855 +       where _type_ eq 'CORR' ;
289  15856 +    run ;
290  15857 +    proc sort data=&corrds;
291  15858 +       by _NAME_ ;
292  15859 +    run ;
293  15860 +    proc transpose data=&corrds out=&corrplotds name
     =_TMP_;
294  15861 +      BY _NAME_ ;
295  15862 +    run ;
296  15863 +    data &corrplotds;
297  15864 +       length _Y_ $100;
298  15865 +       set &corrplotDs;
299  15866 +       if _LABEL_ ne '' then _Y_=_LABEL_ ; else _Y_=
     _TMP_ ;
```

```
300 15867 +    run ;
301 15868 +    data varclus_match(rename=(_TMP_= _NAME_ _LABEL_
    =_X_));
302 15869 +       set &corrplotds;
303 15870 +       where _LABEL_ ne '' ;
304 15871 +       keep _TMP_ _LABEL_ ;
305 15872 +    run ;
306 15873 +    data _null_;
307 15874 +       nobs=0;
308 15875 +       dsid = open('varclus_match');
309 15876 +       if dsid then do;
310 15877 +          nobs = attrn(dsid, 'NOBS');
311 15878 +          dsid = close(dsid);
312 15879 +       end;
313 15880 +       call symput ('CORR_NOBS', nobs);
314 15881 +    run;
315 15882 +    %if &corr_nobs %then %do;
316 15883 +        proc sort data=varclus_match;
317 15884 +           by _name_;
318 15885 +        run ;
319 15886 +        proc sort data=&corrplotds;
320 15887 +           by _name_;
321 15888 +        run ;
322 15889 +        data &corrplotds(keep= _X_ _Y_ col1 rename=(
    col1=Correlation)) ;
323 15890 +           merge varclus_match &corrplotds;
324 15891 +           by _NAME_ ;
325 15892 +           if _X_ eq '' then _X_=_NAME_ ;
326 15893 +           label _X_ = "%sysfunc(sasmsg(sashelp.dmin
    e, rpt_varclus_label_variable, noquote))";
327 15894 +           label _Y_ = "%sysfunc(sasmsg(sashelp.dmin
    e, rpt_varclus_label_variable, noquote))";
328 15895 +           label col1 = "%sysfunc(sasmsg(sashelp.dmi
    ne, rpt_correlation_vlabel, noquote))";
329 15896 +
330 15897 +       run ;
```

```
331  15898 +    %end;
332  15899 +    %else %do;
333  15900 +        proc sort data=&corrplotds;
334  15901 +           by _name_;
335  15902 +        run ;
336  15903 +        data &corrplotds(keep= _NAME_ _Y_ col1 renam
     e=(_NAME_=_X_ col1=Correlation)) ;
337  15904 +           set &corrplotds;
338  15905 +           label _NAME_ = "%sysfunc(sasmsg(sashelp.d
     mine, rpt_varclus_label_variable, noquote))" ;
339  15906 +           label _Y_ = "%sysfunc(sasmsg(sashelp.dmin
     e, rpt_varclus_label_variable, noquote))" ;
340  15907 +           label col1 = "%sysfunc(sasmsg(sashelp.dmi
     ne, rpt_correlation_vlabel, noquote))";
341  15908 +
342  15909 +        run ;
343  15910 +    %end;
344  15911 +    proc sort data=&corrplotds;
345  15912 +       by _X_ _Y_;
346  15913 +    run ;
347  15914 +    proc datasets lib=work nolist mt=(DATA VIEW);
348  15915 +       delete varclus_match;
349  15916 +    run;
350  15917 +    quit;
351  15918 +
352  15919 +%doendc:
353  15920 +
354  15921 +%mend MakeVarClusCorrData;
355  15922 +
356  15923 +%macro MakeStatPlotData(statds= , outstatplotds=);
357  15924 +   %if %sysfunc(exist(&statds)) %then %do;
358  15925 +
359  15926 +       data varclus_tmp(drop=_NAME_ _NCL_) ;
360  15927 +          set &statDs;
361  15928 +          where _type_ in('MEAN', 'STD', 'N');
362  15929 +       run ;
```

```
363  15930 +       proc transpose data=varclus_tmp out=&outstatp
     lotds;
364  15931 +          id _TYPE_ ;
365  15932 +       run ;
366  15933 +       data &outstatplotds;
367  15934 +          set &outstatplotds(obs=1000);
368  15935 +          label _name_= "%sysfunc(sasmsg(sashelp.dmi
     ne, rpt_varclus_label_name, noquote))";
369  15936 +          label _label_="%sysfunc(sasmsg(sashelp.dmi
     ne, rpt_varclus_label_label, noquote))" ;
370  15937 +          if MEAN ne 0 then SCALEDSTD= STD / MEAN ;
371  15938 +          else SCALEDSTD= STD ;
372  15939 +          label SCALEDSTD = "%sysfunc(sasmsg(sashelp
     .dmine, rpt_varclus_label_scaledstd, noquote))";
373  15940 +       run ;
374  15941 +       proc sort data=&outstatplotds;
375  15942 +          by descending SCALEDSTD ;
376  15943 +       run ;
377  15944 +       proc datasets lib=work nolist mt=(DATA VIEW);
378  15945 +          delete varclus_tmp;
379  15946 +       run;
380  15947 +       quit;
381  15948 +    %end;
382  15949 +
383  15950 +%mend MakeStatPlotData;
384  15951 +
385  15952 +
386  15953 +%macro CreateScoreCode(indata=, ncluscomp=, fileref=
     );
387  15954 +     %EM_GETNAME(KEY=OUTSTATSCORE, type=DATA);
388  15955 +    data &EM_USER_OUTSTATSCORE;
389  15956 +          set &indata;
390  15957 +          if (_TYPE_ in ('SCORE' 'MEAN' 'STD') and _
     NCL_ = &ncluscomp ) or (_TYPE_ in ('MEAN' 'STD'));
391  15958 +          if _TYPE_ = 'MEAN' then _NAME_='MEAN';
392  15959 +          if _TYPE_ = 'STD' then _NAME_='STD';
```

```
393 15960 +          DROP _TYPE_ _NCL_;
394 15961 +     run;
395 15962 +
396 15963 +     filename _file_  "&fileRef";
397 15964 +
398 15965 +     data _null_;
399 15966 +        FILE _file_ MOD;
400 15967 +        put ' ';
401 15968 +        put '/*-----------------------------------
    ------------*/';
402 15969 +        put '/* ' "%sysfunc(sasmsg(sashelp.dmine, rp
    t_varclus_score_title_begin , noquote))" '*/';
403 15970 +        put '/*-----------------------------------
    ------------*/';
404 15971 +        put ' ';
405 15972 +        %let dsid = %sysfunc(open(&EM_USER_OUTSTATSC
    ORE));
406 15973 +
407 15974 +        %let nvar = %sysfunc(attrn(&dsid, NVAR));
408 15975 +        %let vn_name =%sysfunc(varnum(&dsid, _NAME_)
    );
409 15976 +
410 15977 +        %let k = 1;
411 15978 +        %do %while(^%sysfunc(fetch(&dsid)));
412 15979 +                %let _name = %sysfunc(getvarc(&dsid,
     &vn_name));
413 15980 +                %if &k > 2 %then %do;
414 15981 +                 %let cn = %eval(&k-2);
415 15982 +                 put "&_name = 0 ; /*---" "%sysfunc(
    sasmsg(sashelp.dmine, rpt_varclus_score_cluscompnum, noquot
    e, &cn))"  "------ */";
416 15983 +                %end;
417 15984 +                %let k = %eval(&k+1);
418 15985 +        %end;
419 15986 +
420 15987 +        %let rc = %sysfunc(rewind(&dsid));
```

```
421  15988 +
422  15989 +        %do i= 2 %to &nvar;
423  15990 +            %let _varname =  %sysfunc(varname(&dsid,
      &i));
424  15991 +            %do %while(^%sysfunc(fetch(&dsid)));
425  15992 +                %let _name = %sysfunc(getvarc(&dsid,
      &vn_name));
426  15993 +                %if &_name = MEAN %then
427  15994 +                %let _mean = %sysfunc(getvarn(&dsid,
      &i));
428  15995 +                %else %if &_name = STD %then
429  15996 +                %let _std = %sysfunc(getvarn(&dsid,
     &i));
430  15997 +                %else %do;
431  15998 +                      %let coeff =  %sysfunc(getvarn
     (&dsid, &i));
432  15999 +                      %let abscoeff = %sysfunc(abs(&
     coeff));
433  16000 +                         %if &abscoeff >  0 %then %
     do;
434  16001 +                          put "&_name = &_name+&coeff *
      (&_varname - &_mean)/&_std;";
435  16002 +                           %end;
436  16003 +                 %end;
437  16004 +             %end;
438  16005 +             %let rc = %sysfunc(rewind(&dsid));
439  16006 +          %end;
440  16007 +
441  16008 +        %let dsid= %sysfunc(close(&dsid));
442  16009 +       run;
443  16010 +%mend CreateScoreCode;
444  16011 +
445  16012 +
446  16013 +
447  16014 +/*-----------------------------------------------
     --------
```

```
448  16015 +    Instead of using %MakeRSquareData,
449  16016 +    %MakeVarClusResultTable at macro2.source is used
450  16017 + +-----------------------------------------------
     --------*/
451  16018 +
452  16019 +
453  16020 +%macro MakeRSquareData(indata=, inClusRSquare=, outd
     ata=, ncluster=);
454  16021 +
455  16022 +/* modifying  from ods rsquare = data */
456  16023 +
457  16024 +data &outdata(drop= ControlVar  NumberOfClusters Cur
     rentCluster);
458  16025 +    Length Cluster $16;
459  16026 +    length Variable $32;
460  16027 +    Length VariableLabel $64;
461  16028 +    set &indata; retain CurrentCluster;
462  16029 +    if NumberOfClusters ^= &ncluster then delete;
463  16030 +    if strip(Cluster) eq '' then Cluster = CurrentCl
     uster;
464  16031 +    CurrentCluster = Cluster;
465  16032 +   run;
466  16033 +proc sort data =&outdata ;
467  16034 +     by Cluster RsquareRatio;
468  16035 +run;
469  16036 +data _tmprsq(drop=index);
470  16037 +     set &outdata; by Cluster;
471  16038 +     if first.Cluster then do;
472  16039 +     index = strip(scan(Cluster,2 ));
473  16040 +     Variable = "Clus"||index;
474  16041 +     VariableLabel = "Cluster Component "||index;
475  16042 +     OwnCluster = 1;
476  16043 +     NextClosest = .;
477  16044 +     RsquareRatio = 0;
478  16045 +     output;
479  16046 +     end;
```

```
480  16047 +run;
481  16048 +
482  16049 +proc sort data = _tmprsq ;
483  16050 +     by Cluster RsquareRatio;
484  16051 +run;
485  16052 +data &outdata;
486  16053 +     set &outdata _tmprsq;
487  16054 +by Cluster;
488  16055 +run;
489  16056 +
490  16057 +
491  16058 +/* Just create the Selected variable with all YES */
492  16059 +
493  16060 +data &outdata;
494  16061 +    set &outdata; by cluster;
495  16062 +    length Selected $8;
496  16063 +    Selected = 'YES';
497  16064 +    label  OwnCluster = 'R-Sqaure with Cluster Compo
     nent';
498  16065 +    label  NextClosest = 'R-Sqaure with Next Cluster
      Component';
499  16066 +    rename OwnCluster = RSqWithClusterComp;
500  16067 +    rename NextClosest = RSqWithNextClusComp;
501  16068 +run;
502  16069 +
503  16070 +
504  16071 +/* Selected = Y/N will be done %score section -----
505  16072 +
506  16073 +%if &EM_PROPERTY_EXPORTEDCOMP ne CLUSTERCOMP %then %
     do;
507  16074 +data &outdata;
508  16075 +    set &outdata; by cluster;
509  16076 +    length Selected $8;
510  16077 +    if first.Cluster then Selected = 'YES';
511  16078 +    else Selected = 'NO';
512  16079 +    label  OwnCluster = 'R-Sqaure with Cluster Compo
```

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

```
543 16110 +    run;
544 16111 +data _clusRsq(drop=_NAME_ _LABEL_);
545 16112 +     set _clusRsq; by cluster;
546 16113 +     if last.Cluster then output;
547 16114 +     label  COL1 = 'R-Sqaure with Next Cluster Compo
    nent';
548 16115 +     rename COL1 = RSqWithNextClusComp;
549 16116 +     rename Cluster = Variable;
550 16117 +     label  Cluster = "Variable";
551 16118 +run;
552 16119 +
553 16120 +proc sort data =&outdata;
554 16121 +     by Variable;
555 16122 +run;
556 16123 +data &outdata;
557 16124 +    merge &outdata _clusRsq;
558 16125 +    by Variable;
559 16126 +run;
560 16127 +proc sort data =&outdata;
561 16128 +by Cluster  RsquareRatio;
562 16129 +run;
563 16130 +quit;
564 16131 +%end;
565 16132 +
566 16133 +proc datasets lib = work nolist;
567 16134 +     delete _tmprsq _clusRsq;
568 16135 +     run;
569 16136 +quit;
570 16137 +
571 16138 +%mend MakeRSquareData;
572 16139 +
573 16140 +
574 16141 +/*-----------------------------------------------
    ----------------------*/
575 16142 +
576 16143 +
```

```
577  16144 +
578  16145 +%macro ModifyCorr(indata=,
579  16146 +                  outdata=,
580  16147 +                  rsquare = Y
581  16148 +                  );
582  16149 +    data corr_tmp;
583  16150 +        set &indata;
584  16151 +    run;
585  16152 +     proc sql;
586  16153 +           update &indata
587  16154 +           set
588  16155 +     %let dsid = %sysfunc(open(work.corr_tmp));
589  16156 +     %let nvar = %sysfunc(attrn(&dsid, NVAR));
590  16157 +          %do i = 4 %to &nvar;
591  16158 +          %let _name = %sysfunc(varname(&dsid, &i));
592  16159 +             %if &rsquare eq Y %then %let _name_md =
      &_name.**2;
593  16160 +            %else  %let _name_md = &_name;
594  16161 +            %if &i < &nvar %then %do;
595  16162 +               &_name = 1- &_name_md ,
596  16163 +            %end;
597  16164 +            %else %do;
598  16165 +               &_name = &_name_md where _TYPE_ conta
     ins 'CORR' ;
599  16166 +            %end;
600  16167 +          %end;
601  16168 +      %let dsid= %sysfunc(close(&dsid));
602  16169 +
603  16170 +      select * from &indata;
604  16171 +      run;
605  16172 +      proc datasets lib = work nolist;
606  16173 +           delete corr_tmp;
607  16174 +      run;
608  16175 +      quit;
609  16176 +
610  16177 + %mend ModifyCorr;
```

```
611 16178 +
612 16179 + %macro MakeClusStructCorrData(indata=,outdata=, ncl
    uster=, Rsquare=N);
613 16180 +    data &outdata(drop= _NCL_  _TYPE_);
614 16181 +       set &indata;
615 16182 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
    E_) eq 'STRUCTUR') then delete;
616 16183 +       rename _NAME_ = Cluster;
617 16184 +       label _NAME_ = "%sysfunc(sasmsg(sashelp.dmine
    , rpt_varclus_label_clustername,  noquote))";
618 16185 +    run;
619 16186 +    %if &RSquare eq Y %then %do;
620 16187 +     data corr_tmp;
621 16188 +        set &outdata;
622 16189 +     run;
623 16190 +
624 16191 +     data &outdata(drop=i);
625 16192 +          set &outdata;
626 16193 +          %let dsid = %sysfunc(open(work.corr_tmp));
627 16194 +          %let nvar = %sysfunc(attrn(&dsid, NVAR));
628 16195 +          %do i = 2 %to &nvar;
629 16196 +            %let _name = %sysfunc(varname(&dsid, &i)
    );
630 16197 +            %let _name_md = &_name.**2;
631 16198 +                &_name = &_name_md;
632 16199 +          %end;
633 16200 +      %let dsid= %sysfunc(close(&dsid));
634 16201 +      run;
635 16202 +      proc datasets lib = work nolist;
636 16203 +           delete corr_tmp;
637 16204 +      run;
638 16205 +
639 16206 +    %end;
640 16207 +     quit;
641 16208 +%mend MakeClusStructCorrData;
642 16209 +
```

```
643 16210 +%macro MakeInterClusCorrData(indata=, outdata=, nclu
    ster=, RSquare=N, makeplotds=N, plotds=);
644 16211 +    data &outdata(drop= _NCL_  _TYPE_);
645 16212 +       set &indata;
646 16213 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
    E_) eq 'CCORR') then delete;
647 16214 +       rename _NAME_ = Cluster;
648 16215 +       label _NAME_ = "%sysfunc(sasmsg(sashelp.dmine
    , rpt_varclus_label_clustername,  noquote))";
649 16216 +    run;
650 16217 +    data corr_tmp;
651 16218 +        set &outdata;
652 16219 +    run;
653 16220 +
654 16221 +    %let dsid = %sysfunc(open(work.corr_tmp));
655 16222 +    %let nclus2= %eval(&ncluster+1);
656 16223 +    data &outdata;
657 16224 +         set &outdata;
658 16225 +          %do i = 2 %to &nclus2;
659 16226 +          %let i_1 = %eval(&i-1);
660 16227 +            %let _name = %sysfunc(varname(&dsid, &i)
    );
661 16228 +            %let _newName = Clus&i_1;
662 16229 +                rename &_name = &_newName; ;
663 16230 +                *label &_name ="Cluster &i_1";
664 16231 +                label &_name = "%sysfunc(sasmsg(sash
    elp.dmine, rpt_varclus_label_clusternum, noquote,  &i_1))";
665 16232 +          %end;
666 16233 +          keep Cluster
667 16234 +          %do i = 2 %to &nclus2;
668 16235 +                %let _name = %sysfunc(varname(&dsid,
     &i));
669 16236 +                &_name
670 16237 +          %end;
671 16238 +          ;
672 16239 +    %let dsid= %sysfunc(close(&dsid));
```

```
673  16240 +     run;
674  16241 +     quit;
675  16242 +
676  16243 +    %if &RSquare eq Y %then %do;
677  16244 +
678  16245 +       data corr_tmp;
679  16246 +        set &outdata;
680  16247 +     run;
681  16248 +
682  16249 +     data &outdata(drop=i);
683  16250 +          set &outdata;
684  16251 +          %let dsid = %sysfunc(open(work.corr_tmp));
685  16252 +          %let nvar = %sysfunc(attrn(&dsid, NVAR));
686  16253 +          %do i = 2 %to &nvar;
687  16254 +            %let _name = %sysfunc(varname(&dsid, &i)
     );
688  16255 +            %let _name_md = &_name.**2;
689  16256 +                &_name = &_name_md;
690  16257 +          %end;
691  16258 +      %let dsid= %sysfunc(close(&dsid));
692  16259 +      run;
693  16260 +    %end;
694  16261 +
695  16262 +    %if &makeplotds eq Y %then %do;
696  16263 +     proc transpose data = &outdata
697  16264 +          out=&plotds(drop=_LABEL_ rename=(_NAME_ =
     Y Cluster=X Col1= Correlation));
698  16265 +          by cluster;
699  16266 +     run;
700  16267 +     data &plotds;
701  16268 +          set  &plotds;
702  16269 +          label x="%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_label_cluster, noquote))";
703  16270 +          label Y="%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_label_cluster, noquote))";
704  16271 +     run;
```

```
705 16272 +     %end;
706 16273 +     proc datasets lib = work nolist;
707 16274 +           delete corr_tmp;
708 16275 +     run;
709 16276 +     quit;
710 16277 +%mend MakeInterClusCorrData;
711 16278 +
712 16279 +
713 16280 +%macro MakeClusConstellData(indata=, outlink=, outno
    de=);
714 16281 +
715 16282 +data &outlink(drop = Selected);
716 16283 +     set &indata;
717 16284 +     LINKID = _N_;
718 16285 +     label LINKID = "%sysfunc(sasmsg(sashelp.dmine,
    rpt_varclus_label_linkid,  noquote))";
719 16286 +     if strip(upcase(Cluster)) eq strip(upcase(Varia
    ble)) then Variable = ClosestCluster;
720 16287 +run;
721 16288 +data &outnode(keep=NODEID TYPE LABEL);
722 16289 +    set &indata;
723 16290 +    length TYPE $16;
724 16291 +    rename Variable = NODEID;
725 16292 +    label  Variable= "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_nodeidvar, noquote))";
726 16293 +    if strip(upcase(Cluster)) eq strip(upcase(Variab
    le))
727 16294 +    then  TYPE = "CLUSTER";
728 16295 +    else  TYPE="VARIABLE";
729 16296 +    label TYPE = "%sysfunc(sasmsg(sashelp.dmine, rpt
    _varclus_label_nodetype, noquote))";
730 16297 +run;
731 16298 +quit;
732 16299 +%mend MakeClusConstellData;
733 16300 +
734 16301 +
```

```
735 16302 +
736 16303 +%macro MakeClusConstellData(indata=, outlink=, outno
    de=);
737 16304 +
738 16305 +data &outlink(drop = Selected);
739 16306 +     set &indata;
740 16307 +     LINKID = _N_;
741 16308 +     label LINKID = "%sysfunc(sasmsg(sashelp.dmine,
    rpt_varclus_label_linkid,  noquote))";
742 16309 +     if strip(upcase(Cluster)) eq strip(upcase(Varia
    ble)) then Variable = ClosestCluster;
743 16310 +run;
744 16311 +data &outnode(keep=NODEID TYPE LABEL);
745 16312 +    set &indata;
746 16313 +    length TYPE $16;
747 16314 +    rename Variable = NODEID;
748 16315 +    label  Variable= "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_nodeidvar, noquote))";
749 16316 +    if strip(upcase(Cluster)) eq strip(upcase(Variab
    le))
750 16317 +    then  TYPE = "CLUSTER";
751 16318 +    else  TYPE="VARIABLE";
752 16319 +    label TYPE = "%sysfunc(sasmsg(sashelp.dmine, rpt
    _varclus_label_nodetype, noquote))";
753 16320 +run;
754 16321 +quit;
755 16322 +%mend MakeClusConstellData;
756 16323 +
757 16324 +
758 16325 +/*--- This will work only when inds is not a view da
    ta -------
759 16326 +
760 16327 +%macro getNVarNObs(inds=, nvar=, nobs=);
761 16328 +
762 16329 +    %global &nvar;
763 16330 +    %global &nobs;
```

```
764  16331 +    data _null_;
765  16332 +         dsid = open("&inds");
766  16333 +         nv = attrn(dsid, 'NVAR');
767  16334 +         no = attrn(dsid, 'NOBS');
768  16335 +         dsid = close(dsid);
769  16336 +         call symput("&nvar", nv);
770  16337 +         call symput("&nobs", no);
771  16338 +    run;
772  16339 +    quit;
773  16340 +%mend  getNVarNObs;
774  16341 +
775  16342 ++-------------------------------------------------
     ------------*/
776  16343 +
777  16344 +
778  16345 +%macro getNVar(inds=, nvar=);
779  16346 +    %global &nvar;
780  16347 +    data _null_;
781  16348 +         dsid = open("&inds");
782  16349 +         nv = attrn(dsid, 'NVAR');
783  16350 +         dsid = close(dsid);
784  16351 +         call symput("&nvar", nv);
785  16352 +    run;
786  16353 +    quit;
787  16354 +%mend  getNVar;
788  16355 +
789  16356 +
790  16357 +
791  16358 +%macro getNObs(inds=, nobs=);
792  16359 +    %global &nobs;
793  16360 +    data _null_;
794  16361 +        set &inds end=eof;
795  16362 +        if eof then call symput("&nobs", _N_);
796  16363 +    run;
797  16364 +    quit;
798  16365 +%mend  getNObs;
```

```
799  16366 +
800  16367 +%Macro CreateVarclusMeta(trainnum=);
801  16368 +   %EM_GETNAME(KEY=VARCLUSMETA, TYPE=DATA);
802  16369 +    data &EM_USER_VARCLUSMETA;
803  16370 +         length TrainNum 8.;
804  16371 +         length NewTrain $8;
805  16372 +         length NGCluster 8.;
806  16373 +         length ExportedComp $16;
807  16374 +         length HideVariable $8;
808  16375 +         TrainNum = &trainnum;
809  16376 +         NewTrain = "Y";
810  16377 +         ExportedComp = "&EM_PROPERTY_EXPORTEDCOMP";
811  16378 +         HideVariable = "&EM_PROPERTY_HIDEVARIABLE";
812  16379 +         NGCluster = 0; /* zero means no twostage */
813  16380 +   run;
814  16381 +   quit;
815  16382 +%mend CreateVarclusMeta;
816  NOTE: %INCLUDE (level 1) ending.
817  NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMEXPL.V
     ARIABLECLUSTERING_MACROS2.SOURCE.
818  16383 +
819  16384 +%macro MakeInterClusCorrData(indata=, outdata=, nclu
     ster=, globalclusid=, RSquare=N, makeplotds=N, plotds=);
820  16385 +    data &outdata(drop= _NCL_  _TYPE_);
821  16386 +       set &indata;
822  16387 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
     E_) eq 'CCORR') then delete;
823  16388 +       %if &globalclusid ne %then %do;
824  16389 +       _NAME_ = "GC&globalclusid._"||upcase(_NAME_);
825  16390 +       rename _NAME_ = Cluster;
826  16391 +       %end;
827  16392 +       %else %do;
828  16393 +        _NAME_ = upcase(_NAME_);
829  16394 +       rename _NAME_ = Cluster;
830  16395 +       %end;
831  16396 +       label _NAME_ = "%sysfunc(sasmsg(sashelp.dmine
```

```
    , rpt_varclus_label_clustername,  noquote))";
832 16397 +    run;
833 16398 +    data corr_tmp;
834 16399 +        set &outdata;
835 16400 +    run;
836 16401 +
837 16402 +    %let dsid = %sysfunc(open(work.corr_tmp));
838 16403 +    %let nclus2= %eval(&ncluster+1);
839 16404 +    data &outdata;
840 16405 +         set &outdata;
841 16406 +          %do i = 2 %to &nclus2;
842 16407 +          %let i_1 = %eval(&i-1);
843 16408 +            %let _name = %sysfunc(varname(&dsid, &i)
    );
844 16409 +            %if &globalclusid ne %then
845 16410 +                %do; %let _newName = GC&globalclusid
    ._CLUS&i_1;
846 16411 +                     rename &_name = &_newName;
847 16412 +                     *label &_name ="GC &globalclusi
    d : Cluster &i_1";
848 16413 +                     label &_name = "%sysfunc(sasmsg
    (sashelp.dmine, rpt_varclus_label_gc_clusternum, noquote,
    &globalclusid, &i_1))";
849 16414 +                %end;
850 16415 +            %else
851 16416 +                %do; %let _newName = CLUS&i_1;
852 16417 +                     rename &_name = &_newName;
853 16418 +                     *label &_name ="Cluster &i_1";
854 16419 +                     label &_name = "%sysfunc(sasmsg
    (sashelp.dmine, rpt_varclus_label_clusternum, noquote,  &i_
    1))";
855 16420 +                %end;
856 16421 +          %end;
857 16422 +          keep Cluster
858 16423 +          %do i = 2 %to &nclus2;
859 16424 +                %let _name = %sysfunc(varname(&dsid,
```

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

```
893 16458 +          label x="%sysfunc(sasmsg(sashelp.dmine, rp
    t_varclus_label_cluster, noquote))";
894 16459 +          label Y="%sysfunc(sasmsg(sashelp.dmine, rp
    t_varclus_label_cluster, noquote))";
895 16460 +          label  Correlation="%sysfunc(sasmsg(sashel
    p.dmine, rpt_correlation_vlabel, noquote))";
896 16461 +     run;
897 16462 +     %end;
898 16463 +     proc datasets lib = work nolist;
899 16464 +           delete corr_tmp;
900 16465 +     run;
901 16466 +     quit;
902 16467 +%mend MakeInterClusCorrData;
903 16468 +
904 16469 +%macro MakeOwnRSquare(indata=, outdata=, ncluster=,
    globalclusid=);
905 16470 +    data _tmpds(drop= _NCL_);
906 16471 +       set &indata;
907 16472 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
    E_) in ('GROUP','RSQUARED')) then delete;
908 16473 +       %if &globalclusid ne %then %do;
909 16474 +           _NAME_ = "GC&globalclusid.";
910 16475 +       rename _NAME_ = Cluster;
911 16476 +       %end;
912 16477 +       %else %do;
913 16478 +        _NAME_ = "CLUS";
914 16479 +        rename _NAME_ = Cluster;
915 16480 +       %end;
916 16481 +       label _NAME_ = "%sysfunc(sasmsg(sashelp.dmine
    , rpt_varclus_label_clustername,  noquote))";
917 16482 +    run;
918 16483 +    proc transpose data =_tmpds out =&outdata;
919 16484 +    run;
920 16485 +
921 16486 +    data &outdata(drop=COL1);
922 16487 +        %if &globalclusid ne %then %do;
```

```
923  16488 +        length GCluster $16;
924  16489 +        %end;
925  16490 +        length Cluster $32;
926  16491 +        length _NAME_ $32;
927  16492 +         set &outdata;
928  16493 +         _NAME_ = upcase(_NAME_);
929  16494 +         rename _NAME_=Variable;
930  16495 +         *label _NAME_="Variable";
931  16496 +         label _NAME_ = "%sysfunc(sasmsg(sashelp.dmi
     ne, rpt_varclus_label_variable, noquote))" ;
932  16497 +         label Cluster = "%sysfunc(sasmsg(sashelp.dm
     ine, rpt_varclus_label_cluster, noquote))" ;
933  16498 +         label GCluster = "%sysfunc(sasmsg(sashelp.d
     mine, rpt_varclus_label_gcluster, noquote))" ;
934  16499 +
935  16500 +         %if &globalclusid ne %then %do;
936  16501 +          GCluster = "GC&globalclusid";
937  16502 +          Cluster = "GC&globalclusid._CLUS"||strip(C
     OL1);
938  16503 +         %end;
939  16504 +         %else %do;
940  16505 +         Cluster = "CLUS"||strip(COL1);
941  16506 +        %end;
942  16507 +         rename COL2 = RSqWithOwnClusComp;
943  16508 +         *label COL2 = "R-Square With Own Cluster Co
     mponent";
944  16509 +         label COL2 = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_ownrsq, noquote))" ;
945  16510 +
946  16511 +   run;
947  16512 +   proc sort data =&outdata;
948  16513 +        by Cluster RSqWithOwnClusComp;
949  16514 +   run;
950  16515 +   proc datasets lib = work nolist;
951  16516 +           delete _tmpds;
952  16517 +   run;
```

```
953 16518 +   quit;
954 16519 +%mend MakeOwnRSquare;
955 16520 +
956 16521 +%macro MakeClusStructCorrData(indata=, outdata=, glo
    balclusid=, ncluster=, Rsquare=N);
957 16522 +    data &outdata(drop= _NCL_  _TYPE_);
958 16523 +       %if &globalclusid ne %then %do;
959 16524 +        length GCluster $16;
960 16525 +        %end;
961 16526 +       set &indata;
962 16527 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
    E_) eq 'STRUCTUR') then delete;
963 16528 +       %if &globalclusid ne %then %do;
964 16529 +         GCluster = "GC&globalclusid";
965 16530 +         _NAME_ = "GC&globalclusid._"||upcase(_NAME_
    );
966 16531 +         rename _NAME_ = Cluster;
967 16532 +         label _NAME_ = "%sysfunc(sasmsg(sashelp.dmi
    ne, rpt_varclus_label_cluster, noquote))" ;
968 16533 +         label GCluster = "%sysfunc(sasmsg(sashelp.d
    mine, rpt_varclus_label_gcluster, noquote))" ;
969 16534 +
970 16535 +       %end;
971 16536 +       %else %do;
972 16537 +         _NAME_ = upcase(_NAME_);
973 16538 +         rename _NAME_ = Cluster;
974 16539 +         label _NAME_ = "%sysfunc(sasmsg(sashelp.dmi
    ne, rpt_varclus_label_cluster, noquote))" ;
975 16540 +         label GCluster = "%sysfunc(sasmsg(sashelp.d
    mine, rpt_varclus_label_gcluster, noquote))" ;
976 16541 +
977 16542 +       %end;
978 16543 +     run;
979 16544 +    %if &RSquare eq Y %then %do;
980 16545 +     data corr_tmp;
981 16546 +          set &outdata;
```

```
 982  16547 +     run;
 983  16548 +     %let istart = 2;
 984  16549 +     %if &globalclusid ne %then %let istart = 3;
 985  16550 +     data &outdata;
 986  16551 +          set &outdata;
 987  16552 +          %let dsid = %sysfunc(open(work.corr_tmp));
 988  16553 +          %let nvar = %sysfunc(attrn(&dsid, NVAR));
 989  16554 +          %do i =&istart %to &nvar;
 990  16555 +            %let _name = %sysfunc(varname(&dsid, &i)
     );
 991  16556 +            %let _name_md = &_name.**2;
 992  16557 +                &_name = &_name_md;
 993  16558 +          %end;
 994  16559 +      %let dsid= %sysfunc(close(&dsid));
 995  16560 +      run;
 996  16561 +      proc datasets lib = work nolist;
 997  16562 +           delete corr_tmp;
 998  16563 +      run;
 999  16564 +    %end;
1000  16565 +     quit;
1001  16566 +%mend MakeClusStructCorrData;
1002  16567 +
1003  16568 +/*
1004  16569 +%MakeClusStructCorrData(indata=playpen._outstat, out
     data=_structrsq , ncluster=7, Rsquare=Y);
1005  16570 +*/
1006  16571 +
1007  16572 +%macro FindNextClosestClusByVar(indata=, outdata=, g
     lobalclusid=, ncluster=);
1008  16573 +
1009  16574 +     /* The indata should be the outdata
1010  16575 +        from %MakeClusStructCorrData(indata=, outdat
     a=, ); */
1011  16576 +
1012  16577 +     proc sort data =&indata out=_tmpclusRsq;
1013  16578 +     by cluster;
```

```
1014  16579 +     run;
1015  16580 +
1016  16581 +     proc transpose data =_tmpclusRsq out=_tmpclusRs
      q;
1017  16582 +      by cluster;
1018  16583 +     run;
1019  16584 +
1020  16585 +     proc sort data=_tmpclusRsq;
1021  16586 +        by _NAME_ COL1;
1022  16587 +     run;
1023  16588 +
1024  16589 +     data _tmpclusRsq;
1025  16590 +         length _NAME_ $32;
1026  16591 +         set _tmpclusRsq; by _NAME_;
1027  16592 +          _NAME_ = upcase(_NAME_);
1028  16593 +          %if &ncluster ne 1 %then %do;
1029  16594 +             if last._NAME_ then delete;
1030  16595 +          %end;
1031  16596 +          %else %do;
1032  16597 +             COL1 = 0;
1033  16598 +          %end;
1034  16599 +     run;
1035  16600 +     /* need to sort again */
1036  16601 +     proc sort data=_tmpclusRsq;
1037  16602 +        by _NAME_ COL1;
1038  16603 +     run;
1039  16604 +
1040  16605 +     data &outdata;
1041  16606 +         set _tmpclusRsq; by _NAME_;
1042  16607 +         Cluster = upcase(Cluster);
1043  16608 +         if last._NAME_ then output;
1044  16609 +         *label  COL1 = 'R-Sqaure with Next Cluster
      Component';
1045  16610 +         label COL1 = "%sysfunc(sasmsg(sashelp.dmine
      , rpt_varclus_label_nextrsq, noquote))" ;
1046  16611 +         rename COL1 = RSqWithNextClusComp;
```

```
1047 16612 +         Cluster = upcase(Cluster);
1048 16613 +         rename Cluster = ClosestCluster;
1049 16614 +         *label  Cluster = "Next Closest Cluster";
1050 16615 +         label Cluster = "%sysfunc(sasmsg(sashelp.dm
     ine, rpt_varclus_label_nextclus, noquote))" ;
1051 16616 +         rename _NAME_ = Variable;
1052 16617 +         label  _NAME_ = "%sysfunc(sasmsg(sashelp.dm
     ine, rpt_varclus_label_variable, noquote))" ;
1053 16618 +    run;
1054 16619 +
1055 16620 +
1056 16621 +    %if &globalclusid ne %then %do;
1057 16622 +    data &outdata;
1058 16623 +       length GCluster $16;
1059 16624 +       set &outdata;
1060 16625 +       GCluster = "GC&globalclusid";
1061 16626 +       run;
1062 16627 +    %end;
1063 16628 +    proc datasets lib = work nolist;
1064 16629 +           delete _tmpclusRsq;
1065 16630 +    run;
1066 16631 +    quit;
1067 16632 +%mend FindNextClosestClusByVar;
1068 16633 +
1069 16634 +
1070 16635 +%macro FindNextClosestClusByCluster(indata=, outdata
     =, globalclusid=, ncluster=);
1071 16636 +     /* The indata should be the outdata from %MakeI
     nterClusCorrData(indata=, outdata=, ); */
1072 16637 +     proc sort data =&indata out=_tmpclusRsq;
1073 16638 +     by cluster;
1074 16639 +     run;
1075 16640 +     proc transpose data =_tmpclusRsq out=_tmpclusRs
     q;
1076 16641 +     by cluster;
1077 16642 +     run;
```

```
1078 16643 +     proc sort data=_tmpclusRsq;
1079 16644 +        by _NAME_ col1;
1080 16645 +     run;
1081 16646 +     data _tmpclusRsq;
1082 16647 +         length _NAME_ $32;
1083 16648 +         set _tmpclusRsq; by _NAME_;
1084 16649 +          _NAME_ = upcase(_NAME_);
1085 16650 +          %if &ncluster ne 1 %then %do;
1086 16651 +             if last._NAME_ then delete;
1087 16652 +          %end;
1088 16653 +          %else %do;
1089 16654 +             COL1 = 0;
1090 16655 +          %end;
1091 16656 +     run;
1092 16657 +     data &outdata;
1093 16658 +         set _tmpclusRsq; by _NAME_;
1094 16659 +         Cluster = upcase(Cluster);
1095 16660 +         if last._NAME_ then output;
1096 16661 +         *label  COL1 = 'R-Sqaure with Next Cluster
     Component';
1097 16662 +         label COL1 = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_nextrsq, noquote))" ;
1098 16663 +         rename COL1 = RSqWithNextClusComp;
1099 16664 +         Cluster = upcase(Cluster);
1100 16665 +         rename Cluster = ClosestCluster;
1101 16666 +         *label  Cluster = "Next Closest Cluster";
1102 16667 +         label Cluster = "%sysfunc(sasmsg(sashelp.dm
     ine, rpt_varclus_label_nextclus, noquote))" ;
1103 16668 +         rename _NAME_ = Variable;
1104 16669 +         *label  _NAME_ = "Variable";
1105 16670 +         label  _NAME_ = "%sysfunc(sasmsg(sashelp.dm
     ine, rpt_varclus_label_variable, noquote))" ;
1106 16671 +
1107 16672 +    run;
1108 16673 +    %if &globalclusid ne %then %do;
1109 16674 +    data &outdata;
```

```
1110 16675 +       length GCluster $16;
1111 16676 +       set &outdata;
1112 16677 +       GCluster = "GC&globalclusid";
1113 16678 +       run;
1114 16679 +    %end;
1115 16680 +
1116 16681 +    proc datasets lib = work nolist;
1117 16682 +           delete _tmpclusRsq;
1118 16683 +    run;
1119 16684 +
1120 16685 +    quit;
1121 16686 +%mend FindNextClosestClusByCluster;
1122 16687 +
1123 16688 +%macro MakeVarClusResultTable(indata1=, indata2=, in
     data3=, outdata=, globalclusid=, ncluster=, selectedcomp=cl
     ustercomp);
1124 16689 +/*----
1125 16690 +  indata1=_ownRsq, indata2=_nextVarRsq, indata3=_nex
     tClusRSq,
1126 16691 +-----------*/
1127 16692 +
1128 16693 +proc sort data =&indata1;
1129 16694 +    by Variable;
1130 16695 +run;
1131 16696 +proc sort data =&indata2;
1132 16697 +    by Variable;
1133 16698 +run;
1134 16699 +data &outdata;
1135 16700 +    merge &indata1 &indata2;
1136 16701 +    by Variable;
1137 16702 +    length Type $16;
1138 16703 +    Type = 'Variable';
1139 16704 +    *label Type ='Type';
1140 16705 +    label Type = "%sysfunc(sasmsg(sashelp.dmine, rpt
     _varclus_label_type, noquote))" ;
1141 16706 +run;
```

```
1142 16707 +
1143 16708 +
1144 16709 +data &indata3;
1145 16710 +    set  &indata3 ;
1146 16711 +    length RSqWithOwnClusComp 8.;
1147 16712 +    Cluster = Variable;
1148 16713 +    RSqWithOwnClusComp = 1;
1149 16714 +    *label RSqWithOwnClusComp = "R-Square With Own C
     luster Component";
1150 16715 +    label RSqWithOwnClusComp = "%sysfunc(sasmsg(sash
     elp.dmine, rpt_varclus_label_ownrsq, noquote))" ;
1151 16716 +    length Type $16;
1152 16717 +    Type = 'ClusterComp';
1153 16718 +    label Type = "%sysfunc(sasmsg(sashelp.dmine, rpt
     _varclus_label_type, noquote))" ;
1154 16719 +
1155 16720 +;
1156 16721 +run;
1157 16722 +
1158 16723 +proc sort data=&outdata;
1159 16724 +    by Cluster;
1160 16725 +run;
1161 16726 +proc sort data =&indata3;
1162 16727 +    by Cluster;
1163 16728 +run;
1164 16729 +
1165 16730 +data &outdata;
1166 16731 +     set &outdata &indata3;
1167 16732 +     by Cluster;
1168 16733 +run;
1169 16734 +
1170 16735 +
1171 16736 +/* Create the Selected variable with all YES */
1172 16737 +
1173 16738 +data &outdata;
1174 16739 +     set &outdata;
```

```
1175 16740 +     length RsqRatio 8.;
1176 16741 +     length Selected $8;
1177 16742 +     *label RSqRatio = "1-R**2 Ratio";
1178 16743 +     label RSqRatio =  "%sysfunc(sasmsg(sashelp.dmin
     e, rpt_varclus_label_oneminusrsq, noquote))" ;
1179 16744 +     *label Selected = "Variable Selected";
1180 16745 +     label Selected = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_varselected, noquote))" ;
1181 16746 +     RsqRatio = (1-RSqWithOwnClusComp)/(1-RSqWithNex
     tClusComp);
1182 16747 +     Selected ='YES';
1183 16748 +     rename _LABEL_ = Label;
1184 16749 +     label _LABEL_ =  "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_label, noquote))" ;
1185 16750 +run;
1186 16751 +
1187 16752 +
1188 16753 +/*----  Selected = Y/N will be assigened at the %sco
     re                  -------+
1189 16754 +        Just create the Selected variable with all Y
     ES at the step above
1190 16755 + +-----------------------------------------------
     ----------------------------+
1191 16756 +
1192 16757 +proc sort data=&outdata;
1193 16758 +    by Cluster RsqRatio;
1194 16759 +run;
1195 16760 +
1196 16761 +%if &selectedcomp eq CLUSTERCOMP %then %do;
1197 16762 +data &outdata;
1198 16763 +     set &outdata; by Cluster;
1199 16764 +     length Selected $8;
1200 16765 +     label Selected = "Variable Selected";
1201 16766 +     if  first.Cluster then Selected ='Yes';
1202 16767 +     else Selected = 'No';
1203 16768 +    run;
```

```
1204 16769 +%end;
1205 16770 +%else %do;
1206 16771 +data &outdata(drop = _var _varchange);
1207 16772 +     set &outdata; retain _var 0; by Cluster;
1208 16773 +     length Selected $8;
1209 16774 +     label Selected = "Variable Selected";
1210 16775 +     if first.Cluster then  _varchange = 0;
1211 16776 +     else _varchange =1;
1212 16777 +     if _var ne _varchange then  Selected  = 'Yes';
1213 16778 +     else Selected = 'No';
1214 16779 +     if last.cluster then  _var = 0;
1215 16780 +     else _var = _varchange;
1216 16781 +run;
1217 16782 +%end;
1218 16783 +
1219 16784 +-------------------------------------------------
     ----------*/
1220 16785 +
1221 16786 +quit;
1222 16787 +%mend MakeVarClusResultTable;
1223 16788 +
1224 16789 +%Macro MakePlotDataFromCorrTable(indata=, outdata=,
     globalclusid=);
1225 16790 +     proc sort data =&indata;
1226 16791 +        by cluster;
1227 16792 +     run;
1228 16793 +     proc transpose data =&indata
1229 16794 +          out=&outdata(drop=_LABEL_ rename=(_NAME_ =
      Y Cluster=X Col1= Correlation));
1230 16795 +          by cluster;
1231 16796 +     run;
1232 16797 +     data &outdata;
1233 16798 +          set  &outdata;
1234 16799 +          label x= "%sysfunc(sasmsg(sashelp.dmine, r
     pt_varclus_label_cluster, noquote))" ;
1235 16800 +          label Y= "%sysfunc(sasmsg(sashelp.dmine, r
```

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

```
1266  16831 +          %let varname = %sysfunc(varname(&dsid, &i)
      );
1267  16832 +          &varname
1268  16833 +          %end;
1269  16834 +       %end;
1270  16835 +       ;
1271  16836 +      %if &freq ne  %then %do;
1272  16837 +          freq &freq;
1273  16838 +      %end;
1274  16839 +      %if &weight ne  %then %do;
1275  16840 +          weight &weight;
1276  16841 +      %end;
1277  16842 +
1278  16843 +      run;
1279  16844 +      %if &corrmatrix eq Y %then %do;
1280  16845 +      data  &outcorr (drop = _NCL_ _TYPE_);
1281  16846 +           set  &outstat ;
1282  16847 +           if _TYPE_ ='CORR' then output;
1283  16848 +      run;
1284  16849 +      %end;
1285  16850 +    %end;
1286  16851 +    %else %do;
1287  16852 +       proc corr data=&indata outp=&outstat noprint;
1288  16853 +            var
1289  16854 +       %let dsid = %sysfunc(open(&indata));
1290  16855 +       %let nvar = %sysfunc(attrn(&dsid, NVAR));
1291  16856 +          %do i = 1 %to &nvar;
1292  16857 +              %let _name = %sysfunc(varname(&dsid, &
      i));
1293  16858 +              %if &_name ne &target %then;
1294  16859 +              &_name
1295  16860 +           %end;
1296  16861 +        %let dsid= %sysfunc(close(&dsid));
1297  16862 +        ;
1298  16863 +        with &target;
1299  16864 +        run;
```

```
1300  16865 +    %end;
1301  16866 +    quit;
1302  16867 +%mend MakeCorrelation;
1303  16868 +
1304  16869 +
1305  16870 +%macro MakeCorrelationDistance(indata=,
1306  16871 +                                outdata=,
1307  16872 +                                rsquare = N
1308  16873 +                                );
1309  16874 +    data corr_tmp;
1310  16875 +        set &indata;
1311  16876 +        if _N_ = 1 then do;
1312  16877 +           output;
1313  16878 +           stop;
1314  16879 +         end;
1315  16880 +    run;
1316  16881 +    %if &outdata ne  %then %let  _outdata = &outdata
      ;
1317  16882 +    %else %let _outdata = &indata;
1318  16883 +
1319  16884 +    data &_outdata;
1320  16885 +         set &indata;
1321  16886 +
1322  16887 +         %let dsid = %sysfunc(open(work.corr_tmp));
1323  16888 +         %let nvar = %sysfunc(attrn(&dsid, NVAR));
1324  16889 +          %do i = 2 %to &nvar;
1325  16890 +              %let _name = %sysfunc(varname(&dsid, &
      i));
1326  16891 +              %if &rsquare eq Y %then %let _name_md
      = &_name.**2;
1327  16892 +              %else  %let _name_md = &_name;
1328  16893 +              &_name = 1- &_name_md;
1329  16894 +          %end;
1330  16895 +      %let dsid= %sysfunc(close(&dsid));
1331  16896 +      run;
1332  16897 +      proc datasets lib = work nolist;
```

```
1333 16898 +           delete corr_tmp;
1334 16899 +      run;
1335 16900 +      quit;
1336 16901 +  %mend MakeCorrelationDistance;
1337 16902 +
1338 16903 +
1339 16904 +%macro UpdateOutStatCorrToDistance(indata=, /* indat
     a should be a outstat from proc varclus */
1340 16905 +                                    rsquare = N
1341 16906 +                                    );
1342 16907 +    data corr_tmp;
1343 16908 +        set &indata;
1344 16909 +    run;
1345 16910 +    proc sql noprint;
1346 16911 +           update &indata
1347 16912 +           set
1348 16913 +     %let dsid = %sysfunc(open(work.corr_tmp));
1349 16914 +     %let nvar = %sysfunc(attrn(&dsid, NVAR));
1350 16915 +          %do i = 4 %to &nvar;
1351 16916 +          %let _name = %sysfunc(varname(&dsid, &i));
1352 16917 +             %if &rsquare eq Y %then %let _name_md =
      &_name.**2;
1353 16918 +            %else  %let _name_md = &_name;
1354 16919 +            %if &i < &nvar %then %do;
1355 16920 +               &_name = 1- &_name_md ,
1356 16921 +            %end;
1357 16922 +            %else %do;
1358 16923 +               &_name = &_name_md where _TYPE_ eq 'C
     ORR' ;
1359 16924 +            %end;
1360 16925 +          %end;
1361 16926 +      %let dsid= %sysfunc(close(&dsid));
1362 16927 +
1363 16928 +      select * from &indata;
1364 16929 +      run;
1365 16930 +     data &indata( drop = _NCL_);
```

```
1366 16931 +           set &indata;
1367 16932 +           if _TYPE_ not in ('CORR', 'STD', 'N', 'ME
     AN') then delete;
1368 16933 +           if _TYPE_ ='CORR' then _TYPE_ ='DISTANCE'
     ;
1369 16934 +     run;
1370 16935 +     data &indata(DROP = _NCL_);
1371 16936 +           set &indata;
1372 16937 +           if _TYPE_ = 'CORR' then _TYPE_ ='DISTANCE
     ';
1373 16938 +           if _TYPE_ not in ('DISTANCE', 'N', 'STD',
      'MEAN') then delete;
1374 16939 +           rename _NAME_ = _VAR_;
1375 16940 +     run;
1376 16941 +     proc datasets lib = work nolist;
1377 16942 +           delete corr_tmp;
1378 16943 +     run;
1379 16944 +     quit;
1380 16945 + %mend UpdateOutStatCorrToDistance;
1381 16946 +
1382 16947 +
1383 16948 +%macro HierClusWithCorr(indata= ,
1384 16949 +                        ncluster=,
1385 16950 +                        method = Ward,
1386 16951 +                        outtree = _outtree,
1387 16952 +                        idvar =_VAR_,
1388 16953 +                        outdata=,
1389 16954 +                        rescore = N,
1390 16955 +                        newncluster=
1391 16956 +                        );
1392 16957 +      %global &newncluster;
1393 16958 +      %if &rescore ne Y %then %do;
1394 16959 +      proc cluster data=&indata(type=Distance where=
     (upcase(strip(_TYPE_)) = "DISTANCE"))
1395 16960 +                   method=&method outtree=&outtree n
     oprint;
```

```
1396 16961 +           id &idvar;
1397 16962 +      run;
1398 16963 +      %end;
1399 16964 +      proc tree data=&outtree nclusters = &ncluster
     out=&outdata noprint;
1400 16965 +      run;
1401 16966 +      /* ----- Check some variables like CL1, CL5...
     , remove them ----*/
1402 16967 +      proc contents data =&indata out=_outcontent(ke
     ep=NAME) noprint;
1403 16968 +      run;
1404 16969 +      data _outcontent;
1405 16970 +          set _outcontent;
1406 16971 +          if NAME in ('_TYPE_' , '_VAR_') then delet
     e;
1407 16972 +          index = 1;
1408 16973 +          rename NAME = _NAME_;
1409 16974 +      run;
1410 16975 +      proc sort data=_outcontent;
1411 16976 +          by _NAME_;
1412 16977 +      run;
1413 16978 +      proc sort data =&outdata;
1414 16979 +           by _NAME_;
1415 16980 +      run;
1416 16981 +      data &outdata(drop=index);
1417 16982 +          merge &outdata _outcontent;
1418 16983 +          by _NAME_;
1419 16984 +          if index = . then delete;
1420 16985 +      run;
1421 16986 +      /*-----------------------------------------
     ---------------*/
1422 16987 +      data &outdata;
1423 16988 +           length CLUSNAME $16;
1424 16989 +           set &outdata;
1425 16990 +           if CLUSTER > &ncluster then delete;
1426 16991 +           CLUSNAME='GC'||strip(CLUSTER);
```

```
1427 16992 +           *label CLUSNAME = "Cluster Name";
1428 16993 +           label CLUSNAME  = "%sysfunc(sasmsg(sashel
     p.dmine, rpt_varclus_label_clustername, noquote))" ;
1429 16994 +           rename _NAME_ = VARIABLE ;
1430 16995 +           *label _NAME_ = "Variable";
1431 16996 +           *label CLUSTER = "Cluster";
1432 16997 +           label _NAME_ ="%sysfunc(sasmsg(sashelp.dm
     ine, rpt_varclus_label_variable, noquote))" ;
1433 16998 +           label CLUSTER ="%sysfunc(sasmsg(sashelp.d
     mine, rpt_varclus_label_cluster, noquote))" ;
1434 16999 +      run;
1435 17000 +      proc sort data=&outdata out=&outdata;
1436 17001 +           by CLUSTER;
1437 17002 +      run;
1438 17003 +      proc means data =&outdata noprint;
1439 17004 +           output out=_meanout;
1440 17005 +      run;
1441 17006 +      data _null_;
1442 17007 +            set _meanout;
1443 17008 +            if strip(_STAT_) eq 'MAX' then do;
1444 17009 +            call symput("&newncluster", CLUSTER);
1445 17010 +            stop;
1446 17011 +            end;
1447 17012 +      run;
1448 17013 +
1449 17014 +      proc datasets lib = work nolist;
1450 17015 +           delete _outcontent _meanout;
1451 17016 +      run;
1452 17017 +      quit;
1453 17018 +%mend  HierClusWithCorr;
1454 17019 +
1455 17020 +%macro CreateScoreCode2(indata=, ncluscomp=, globalc
     lusid=, fileref=);
1456 17021 +
1457 17022 +     data _tmpindata;
1458 17023 +          set &indata;
```

```
1459 17024 +          if (_TYPE_ in ('SCORE' 'MEAN' 'STD') and _
     NCL_ = &ncluscomp ) or (_TYPE_ in ('MEAN' 'STD'));
1460 17025 +          if _TYPE_ = 'MEAN' then _NAME_='MEAN';
1461 17026 +          if _TYPE_ = 'STD' then _NAME_='STD';
1462 17027 +          if _TYPE_ = 'SCORE' then _NAME_=upcase("GC
     &globalclusid._"||_NAME_);
1463 17028 +          DROP _TYPE_ _NCL_;
1464 17029 +     run;
1465 17030 +
1466 17031 +     filename _file_  "&fileRef";
1467 17032 +
1468 17033 +     data _null_;
1469 17034 +        FILE _file_ MOD;
1470 17035 +        put ' ';
1471 17036 +        put "/*-------------------------------------
     ------------*/";
1472 17037 +        put '/* ' "%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_score_title_gclus, noquote, &globalclusid))" '*/'
     ;
1473 17038 +        put "/*-------------------------------------
     ------------*/";
1474 17039 +        put ' ';
1475 17040 +        %let dsid = %sysfunc(open(work._tmpindata));
1476 17041 +
1477 17042 +        %let nvar = %sysfunc(attrn(&dsid, NVAR));
1478 17043 +        %let vn_name =%sysfunc(varnum(&dsid, _NAME_)
     );
1479 17044 +        %let k = 1;
1480 17045 +        %do %while(^%sysfunc(fetch(&dsid)));
1481 17046 +              %let _name = %sysfunc(getvarc(&dsid,
     &vn_name));
1482 17047 +              %if &k > 2 %then %do;
1483 17048 +               %let cn = %eval(&k-2);
1484 17049 +                put "&_name = 0 ; /*---" "%sysfunc(
     sasmsg(sashelp.dmine, rpt_varclus_score_gcluscompnum, noquo
     te, &globalclusid, &cn))" "------ */";
```

```
1485  17050 +                %end;
1486  17051 +                %let k = %eval(&k+1);
1487  17052 +
1488  17053 +        %end;
1489  17054 +        %let rc = %sysfunc(rewind(&dsid));
1490  17055 +
1491  17056 +        %do j= 2 %to &nvar;
1492  17057 +            %let _varname = %sysfunc(varname(&dsid,
      &j));
1493  17058 +            %do %while(^%sysfunc(fetch(&dsid)));
1494  17059 +                %let _name = %sysfunc(getvarc(&dsid,
       &vn_name));
1495  17060 +                %if &_name = MEAN %then
1496  17061 +                %let _mean = %sysfunc(getvarn(&dsid,
       &j));
1497  17062 +                %else %if &_name = STD %then
1498  17063 +                %let _std = %sysfunc(getvarn(&dsid,
      &j));
1499  17064 +                %else %do;
1500  17065 +                      %let coeff =  %sysfunc(getvarn
      (&dsid, &j));
1501  17066 +                      %let abscoeff = %sysfunc(abs(&
      coeff));
1502  17067 +                          %if &abscoeff >  0 %then %
      do;
1503  17068 +                       put "&_name = &_name+&coeff *
       (&_varname - &_mean)/&_std;";
1504  17069 +                           %end;
1505  17070 +                %end;
1506  17071 +             %end;
1507  17072 +             %let rc = %sysfunc(rewind(&dsid));
1508  17073 +         %end;
1509  17074 +
1510  17075 +        %let dsid= %sysfunc(close(&dsid));
1511  17076 +
1512  17077 +     run;
```

```
1513  17078 +
1514  17079 +     filename _file_;
1515  17080 +     proc datasets lib = work nolist;
1516  17081 +           delete _tmpindata;
1517  17082 +     run;
1518  17083 +     quit;
1519  17084 +%mend CreateScoreCode2;
1520  17085 +
1521  17086 +
1522  17087 +%macro MakeDeltaCode2(groupds=,deltacodefile=);
1523  17088 +
1524  17089 +     /*--- Build Code to Modify Metadata ---*/
1525  17090 +     filename X "&deltacodefile";
1526  17091 +     data _null_;
1527  17092 +        FILE X;
1528  17093 +        set &groupds end=eof;
1529  17094 +        if _N_=1 then do;
1530  17095 +           %if &EM_PROPERTY_INCLUDECLASSVAR eq Y %th
      en %do;
1531  17096 +            put "if upcase(strip(ROLE)) ='INPUT' and
       upcase(strip(LEVEL)) ^='INTERVAL' then ROLE ='REJECTED' ;"
      ;
1532  17097 +           %end;
1533  17098 +           put "if upcase(strip(ROLE))='INPUT' and u
      pcase(strip(LEVEL))='INTERVAL' then do;";
1534  17099 +           put "if upcase(strip(NAME)) in (";
1535  17100 +        end;
1536  17101 +        if Strip(upcase(Selected)) eq 'YES' then do;
1537  17102 +           string = '"'!!trim(left(VARIABLE))!!'"';
1538  17103 +           put string;
1539  17104 +        end;
1540  17105 +        if eof then do;
1541  17106 +           put ') then ROLE="INPUT";';
1542  17107 +           put 'else ROLE="REJECTED";';
1543  17108 +           put 'end;';
1544  17109 +
```

```
1545 17110 +           %if %upcase(&EM_PROPERTY_HIDEVARIABLE) eq
      Y %then %do;
1546 17111 +             put 'if upcase(strip(ROLE)) = "REJECTED
     " then delete ;';
1547 17112 +           %end;
1548 17113 +         end;
1549 17114 +     run;
1550 17115 +     quit;
1551 17116 +
1552 17117 +     filename X;
1553 17118 +     quit;
1554 17119 +%mend MakeDeltaCode2;
1555 17120 +
1556 17121 +%macro getInitialGClusterNumber(indata=, ninput=, nd
     ummy=0, div=100, ngc=);
1557 17122 +  %global &ngc;
1558 17123 +  data _null_;
1559 17124 +  %if &indata ne %then %do;
1560 17125 +      %let dsid = %sysfunc(open(&indata));
1561 17126 +           %let nvar = %sysfunc(attrn(&dsid, NVAR));
1562 17127 +      %let dsid = %sysfunc(close(&dsid));
1563 17128 +  %end;
1564 17129 +  %else %do;
1565 17130 +       %let nvar = %eval(&ninput+&ndummy); ;
1566 17131 +  %end;
1567 17132 +  %let numgc = %eval(&nvar/&div+2);
1568 17133 +  %let &ngc = &numgc;
1569 17134 +   run;
1570 17135 +  quit;
1571 17136 +%mend getInitialGClusterNumber;
1572 17137 +
1573 17138 +
1574 17139 +%macro MakeGobalConstellData(indata=, outlink=, outn
     ode=);
1575 17140 +data &outlink(drop = Selected);
1576 17141 +     set &indata;
```

```
1577 17142 +     LINKID = _N_;
1578 17143 +     label LINKID = "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_linkid, noquote))" ;
1579 17144 +run;
1580 17145 +data &outnode(keep=NODEID TYPE LABEL);
1581 17146 +    set &indata;
1582 17147 +    length TYPE $16;
1583 17148 +    rename VARIABLE = NODEID;
1584 17149 +    *label  CLUSNAME="Node ID";
1585 17150 +    label CLUSNAME= "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_nodeidvar, noquote))" ;
1586 17151 +    TYPE = "VARIABLE";
1587 17152 +    *label TYPE = "Node Type";
1588 17153 +    label TYPE =  "%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_label_nodetype, noquote))" ;
1589 17154 +    run;
1590 17155 +data _tmp(keep=NODEID TYPE LABEL);
1591 17156 +    set &indata;
1592 17157 +    length TYPE $16;
1593 17158 +    rename CLUSNAME = NODEID;
1594 17159 +    label  CLUSNAME= "%sysfunc(sasmsg(sashelp.dmine,
      rpt_varclus_label_nodeidvar, noquote))" ;
1595 17160 +    TYPE = "GCLUSTER";
1596 17161 +    label TYPE = "%sysfunc(sasmsg(sashelp.dmine, rpt
     _varclus_label_nodetype, noquote))" ;
1597 17162 +    run;
1598 17163 +proc sort data=_tmp;
1599 17164 +    by NODEID;
1600 17165 +run;
1601 17166 +data _tmp;
1602 17167 +    set _tmp; by NODEID;
1603 17168 +    if first.NODEID then output;
1604 17169 +run;
1605 17170 +proc sort data=&outnode;
1606 17171 +    by NODEID;
1607 17172 +run;
```

```
1608 17173 +data  &outnode;
1609 17174 +    set _tmp &outnode;
1610 17175 +run;
1611 17176 +proc datasets lib = work nolist;
1612 17177 +     delete _tmp;
1613 17178 +run;
1614 17179 +quit;
1615 17180 +%mend MakeGobalConstellData;
1616 17181 +
1617 17182 +/* Make contellation plot data among GCLUSTERS */
1618 17183 +
1619 17184 +%Macro MakeGClusterConstData(indata=, inoutrsq=, out
     node=, outlink=);
1620 17185 +
1621 17186 +data &outlink(keep = _NAME_ _PARENT_ _LABEL_ LINKID)
     ;
1622 17187 +     set &indata;
1623 17188 +     LINKID = _N_;
1624 17189 +     if upcase(substr(strip(_NAME_),1, 2))="CL" then
      do;
1625 17190 +        _NAME_ = "ROOT"||upcase(substr(strip(_NAME_)
     ,5));
1626 17191 +     end;
1627 17192 +     if _PARENT_ ne " " and upcase(substr(strip(_PAR
     ENT_),1, 2))="CL" then do;
1628 17193 +        _PARENT_ = "ROOT"||upcase(substr(strip(_PARE
     NT_),5));
1629 17194 +     end;
1630 17195 +     if upcase(substr(strip(_LABEL_),1, 2))="CL" the
     n do;
1631 17196 +        _LABEL_ = "ROOT"||upcase(substr(strip(_LABEL
     _),5));
1632 17197 +    end;
1633 17198 +run;
1634 17199 +
1635 17200 +data _tmp_outrsquare;
```

```
1636 17201 +        set &inoutrsq;
1637 17202 +        if upcase(strip(TYPE)) = 'CLUSTERCOMP' then
     delete;
1638 17203 +run;
1639 17204 +
1640 17205 +proc freq data =_tmp_outrsquare noprint;
1641 17206 +         tables GCluster/out=_tmp_GCLUSFREQ(rename=(
     GCLUSTER=_NAME_));
1642 17207 +run;
1643 17208 +
1644 17209 +data &outnode(keep=_NAME_ TYPE LABEL);
1645 17210 +    set &outlink;
1646 17211 +    length TYPE $16;
1647 17212 +    length LABEL $100;
1648 17213 +    /*label CLUSNAME= "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_nodeidvar, noquote))" ;*/
1649 17214 +
1650 17215 +    if upcase(substr(strip(_NAME_),1, 2))='GC' then
     do;
1651 17216 +    TYPE = "GCLUSTER";
1652 17217 +    LABEL = "%sysfunc(sasmsg(sashelp.dmine, rpt_varc
     lus_label_gcluster, noquote)):"||_NAME_;
1653 17218 +    end;
1654 17219 +    else do;
1655 17220 +    TYPE= "ROOT";
1656 17221 +    LABEL= _NAME_;
1657 17222 +    end;
1658 17223 +    label TYPE  =  "%sysfunc(sasmsg(sashelp.dmine, r
     pt_varclus_label_nodetype, noquote))" ;
1659 17224 +    label LABEL =  "%sysfunc(sasmsg(sashelp.dmine, r
     pt_varclus_label_label, noquote))" ;
1660 17225 +    label _NAME_ =  "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_nodeidvar, noquote))" ;
1661 17226 +    run;
1662 17227 +
1663 17228 +proc sort data=&outnode;
```

```
1664 17229 +     by _NAME_;
1665 17230 +proc sort data=_tmp_GCLUSFREQ;
1666 17231 +     by _NAME_;
1667 17232 +run;
1668 17233 +
1669 17234 +data  &outnode;
1670 17235 +     merge &outnode _tmp_GCLUSFREQ; by _NAME_;
1671 17236 +     if COUNT=. then COUNT=1;
1672 17237 +run;
1673 17238 +
1674 17239 +proc datasets lib = work nolist;
1675 17240 +     delete _tmp_outrsquare _tmp_GCLUSFREQ;
1676 17241 +run;
1677 17242 +
1678 17243 +quit;
1679 17244 +%Mend MakeGClusterConstData;
1680 17245 +
1681 17246 +
1682 17247 +%macro CreateGClusterScoreCode(indata=,  globalclusi
     d=, fileref=);
1683 17248 +
1684 17249 +    data _gscoretmpds;
1685 17250 +          set &indata;
1686 17251 +          if (_TYPE_ in ('SCORE' 'MEAN' 'STD') and _
     NCL_ = 1 ) or (_TYPE_ in ('MEAN' 'STD'));
1687 17252 +          if _TYPE_ = 'MEAN' then _NAME_='MEAN';
1688 17253 +          if _TYPE_ = 'STD' then _NAME_='STD';
1689 17254 +          if _TYPE_ = 'SCORE' then _NAME_ = "GC"||st
     rip(&globalclusid);
1690 17255 +          DROP _TYPE_ _NCL_;
1691 17256 +     run;
1692 17257 +
1693 17258 +    /*  %let gscorefile =  %bquote(&EM_NODEDIR)&EM_D
     SEP.gclusterscore.sas;
1694 17259 +        GCluster Component &globalclusid ------ */
1695 17260 +
```

```
1696 17261 +    filename _file_  "&fileref";
1697 17262 +
1698 17263 +    data _null_;
1699 17264 +        %if &globalclusid eq 1 %then %do;
1700 17265 +          FILE _file_;
1701 17266 +        %end;
1702 17267 +        %else %do;
1703 17268 +          FILE _file_ MOD;
1704 17269 +        %end;
1705 17270 +
1706 17271 +        %let dsid = %sysfunc(open(work._gscoretmpds)
     );
1707 17272 +        %let nvar = %sysfunc(attrn(&dsid, NVAR));
1708 17273 +        %let vn_name =%sysfunc(varnum(&dsid, _NAME_)
     );
1709 17274 +
1710 17275 +        %let k = 1;
1711 17276 +        %do %while(^%sysfunc(fetch(&dsid)));
1712 17277 +                %let _name = %sysfunc(getvarc(&dsid,
      &vn_name));
1713 17278 +                %if &k > 2 %then %do;
1714 17279 +                put "&_name = 0 ; ";
1715 17280 +                %end;
1716 17281 +                %let k = %eval(&k+1);
1717 17282 +        %end;
1718 17283 +
1719 17284 +        %let rc = %sysfunc(rewind(&dsid));
1720 17285 +        %do i= 2 %to &nvar;
1721 17286 +            %let _varname =  %sysfunc(varname(&dsid,
      &i));
1722 17287 +            %do %while(^%sysfunc(fetch(&dsid)));
1723 17288 +                %let _name = %sysfunc(getvarc(&dsid,
      &vn_name));
1724 17289 +                %if &_name = MEAN %then
1725 17290 +                %let _mean = %sysfunc(getvarn(&dsid,
      &i));
```

```
1726 17291 +                %else %if &_name = STD %then
1727 17292 +                %let _std = %sysfunc(getvarn(&dsid,
     &i));
1728 17293 +                %else %do;
1729 17294 +                      %let coeff =  %sysfunc(getvarn
     (&dsid, &i));
1730 17295 +                      %let abscoeff = %sysfunc(abs(&
     coeff));
1731 17296 +                         %if &abscoeff >  0 %then %
     do;
1732 17297 +                       put "&_name = &_name+&coeff *
      (&_varname - &_mean)/&_std;";
1733 17298 +                           %end;
1734 17299 +                 %end;
1735 17300 +             %end;
1736 17301 +             %let rc = %sysfunc(rewind(&dsid));
1737 17302 +
1738 17303 +          %end;
1739 17304 +
1740 17305 +         %let dsid= %sysfunc(close(&dsid));
1741 17306 +        run;
1742 17307 +
1743 17308 +
1744 17309 +        proc datasets lib=work nolist;
1745 17310 +             delete _gscoretmpds;
1746 17311 +        run;
1747 17312 +       quit;
1748 17313 +
1749 17314 +%mend CreateGClusterScoreCode;
1750 17315 +
1751 17316 +
1752 17317 +%macro MakeGClusterCorrelation(Indata=, ngcluster=,
     gscorecode=, outrsquare=);
1753 17318 +
1754 17319 +   %EM_REGISTER(KEY=GSCORE, TYPE=DATA);
1755 17320 +   %EM_GETNAME(KEY=GSCORE, TYPE=DATA);
```

```
1756 17321 +   %EM_REGISTER(KEY=GSCORESTAT, TYPE=DATA);
1757 17322 +   %EM_GETNAME(KEY=GSCORESTAT, TYPE=DATA);
1758 17323 +   %EM_REGISTER(KEY=GSCORETREE, TYPE=DATA);
1759 17324 +   %EM_GETNAME(KEY=GSCORETREE, TYPE=DATA);
1760 17325 +   %EM_REGISTER(KEY=GSCORECORR, TYPE=DATA);
1761 17326 +   %EM_GETNAME(KEY=GSCORECORR, TYPE=DATA);
1762 17327 +   %EM_REGISTER(KEY=GSCORECORRPLOT, TYPE=DATA);
1763 17328 +   %EM_GETNAME(KEY=GSCORECORRPLOT, TYPE=DATA);
1764 17329 +   %EM_REGISTER(KEY=GCLUSLINK, TYPE=DATA);
1765 17330 +   %EM_GETNAME(KEY=GCLUSLINK, TYPE=DATA);
1766 17331 +   %EM_REGISTER(KEY=GCLUSNODE, TYPE=DATA);
1767 17332 +   %EM_GETNAME(KEY=GCLUSNODE, TYPE=DATA);
1768 17333 +
1769 17334 +   filename gsfile "&gscorecode";
1770 17335 +
1771 17336 +   data &EM_USER_GSCORE;
1772 17337 +             set &indata;
1773 17338 +             %include  gsfile;
1774 17339 +        keep
1775 17340 +        %do i=1 %to &ngcluster;
1776 17341 +         %let gcvarname = GC&i;
1777 17342 +         &gcvarname
1778 17343 +        %end;
1779 17344 +     ;
1780 17345 +   run;
1781 17346 +
1782 17347 +   proc varclus data=&EM_USER_GSCORE outstat=&EM_USE
     R_GSCORESTAT outtree=&EM_USER_GSCORETREE
1783 17348 +    %if %upcase(&EM_PROPERTY_CLUSCOMP) eq CENTROID %
     then %do; centroid %end;
1784 17349 +    %if %upcase(&EM_PROPERTY_CLUSSOURCE) eq COV %the
     n %do; cov %end;
1785 17350 +    %if %upcase(&EM_PROPERTY_CLUSHIERACHY) eq Y %the
     n %do; hi %end;
1786 17351 +    noprint  ;
1787 17352 +     var
```

```
1788 17353 +       %do i=1 %to &ngcluster;
1789 17354 +         %let gcvarname = GC&i;
1790 17355 +         &gcvarname
1791 17356 +        %end;
1792 17357 +   ;
1793 17358 +   run;
1794 17359 +
1795 17360 +
1796 17361 +   %MakeVarClusCorrData(statds=&EM_USER_GSCORESTAT,
     corrds=&EM_USER_GSCORECORR, corrplotds=&EM_USER_GSCORECORRP
     LOT );
1797 17362 +   data &EM_USER_GSCORECORRPLOT ;
1798 17363 +        set &EM_USER_GSCORECORRPLOT;
1799 17364 +        rename _X_ = X;
1800 17365 +        rename _Y_ = Y;
1801 17366 +        label _X_ = "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_gcluster, noquote))" ;
1802 17367 +        label _Y_ = "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_gcluster, noquote))" ;
1803 17368 +   run;
1804 17369 +
1805 17370 +   %MakeGClusterConstData(indata=&EM_USER_GSCORETREE
     , inoutrsq=&outrsquare, outnode=&EM_USER_GCLUSNODE, outlink
     =&EM_USER_GCLUSLINK);
1806 17371 +
1807 17372 +   data &EM_USER_GSCORETREE;
1808 17373 +        length _NAME_ $32;
1809 17374 +        length _LABEL_ $100;
1810 17375 +        set &EM_USER_GSCORETREE(DROP=_LABEL_);
1811 17376 +        if upcase(substr(strip(_NAME_),1, 2))='GC' t
     hen do;
1812 17377 +         _LABEL_ = "%sysfunc(sasmsg(sashelp.dmine, r
     pt_varclus_label_gcluster, noquote)):"||_NAME_;
1813 17378 +        end;else do;
1814 17379 +         _LABEL_ = _NAME_;
1815 17380 +        end;
```

```
1816 17381 +       label _LABEL_ = "%sysfunc(sasmsg(sashelp.dmin
     e, rpt_varclus_label_variable, noquote))";
1817 17382 +
1818 17383 +   run;
1819 17384 +
1820 17385 +   quit;
1821 17386 +
1822 17387 +%mend MakeGClusterCorrelation;
1823 17388 +
1824 17389 +
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 17392 +%macro train;
1829 17394 +   filename temp catalog 'sashelp.emexpl.variableclu
     stering_train1.source';
1830 17395 +   %include temp;
1831 17396 +   filename temp catalog 'sashelp.emexpl.variableclu
     stering_train2.source';
1832 17397 +   %include temp;
1833 17398 +   filename temp catalog 'sashelp.emutil.em_copyfile
     .source';
1834 17399 +   %include temp;
1835 17400 +   filename temp;
1836 17402 +   %let VARCLUS_MAXNUMOBS = 100000;
1837 17403 +   %let VARCLUS_MAXNUMVAR = 200;
1838 17404 +   %let trainnum = 0;
1839 17405 +   %let error = 0;
1840 17407 +   %if &EM_IMPORT_DATA eq %then %do;
1841 17408 +       %let  EMEXCEPTIONSTRING = exception.server.IM
     PORT.NOTRAIN,1;
1842 17409 +       %let  error = 1;
1843 17410 +       %goto endtrain;
1844 17411 +   %end;
1845 17413 +    %if (%sysfunc(exist(&EM_IMPORT_DATA)) or %sysfun
```

```
     c(exist(&EM_IMPORT_DATA, VIEW))) < 1 %then %do;
1846 17414 +        %let  EMEXCEPTIONSTRING = exception.server.I
     MPORT.NOTRAIN,1;
1847 17415 +        %let  error = 1;
1848 17416 +        %goto endline;
1849 17417 +    %end;
1850 17419 +   %let _num_input_interval = %eval(&EM_NUM_INTERVAL
     _INPUT+&EM_NUM_INTERVAL_REJECTED);
1851 17420 +   %let _num_input_binary = %eval(&EM_NUM_BINARY_INP
     UT+&EM_NUM_BINARY_REJECTED);
1852 17421 +   %let _num_input_nominal = %eval(&EM_NUM_NOMINAL_I
     NPUT+&EM_NUM_NOMINAL_REJECTED);
1853 17422 +   %let _num_input_ordinal = %eval(&EM_NUM_ORDINAL_I
     NPUT+&EM_NUM_ORDINAL_REJECTED);
1854 17423 +   %let _num_input_class = %eval(&_num_input_binary+
     &_num_input_nominal+&_num_input_ordinal);
1855 17424 +   %let _num_input_total = %eval(&_num_input_interva
     l+&_num_input_class);
1856 17426 +   %if (&EM_PROPERTY_INCLUDECLASSVAR eq Y) %then %do
     ;
1857 17427 +        %if  &_num_input_total < 2 %then %do;
1858 17428 +            %let  EMEXCEPTIONSTRING = exception.serv
     er.METADATA.USEATLEAST2INPUTREJECT;
1859 17429 +            %let error = 2;
1860 17430 +            %goto endtrain;
1861 17431 +        %end;
1862 17432 +   %end;
1863 17433 +   %else %do;
1864 17434 +         %if  &_num_input_interval < 2 %then %do;
1865 17435 +         %let  EMEXCEPTIONSTRING = exception.server.
     METADATA.USEATLEAST2INPUTREJECT;
1866 17436 +         %let error = 2;
1867 17437 +         %goto endtrain;
1868 17438 +         %end;
1869 17439 +   %end;
1870 17441 +   %em_checkerror();
```

```
1871  17443 +   %if &EMEXCEPTIONSTRING ne %then %do;
1872  17444 +       %goto endtrain;
1873  17445 +   %end;
1874  17447 +   %if (&EM_PROPERTY_INCLUDECLASSVAR eq Y) and (&_nu
      m_input_class > 0) %then %do;
1875  17449 +        %EM_GETNAME(key=OUTDUMMY, type=DATA);
1876  17450 +        %MakeDummyVariables( indata=&EM_IMPORT_DATA,
1877  17451 +                             outvar=&EM_USER_OUTDUMM
      Y,
1878  17452 +                             outdata=_newtrainds,
1879  17453 +                             fileref=&EM_FILE_EMFLOW
      SCORECODE);
1880  17455 +       %if &EM_PROPERTY_SUPPRESSSAMPWARN eq N %then
      %do;
1881  17456 +          %getNObs(inds=_newtrainds, nobs=_varclus_
      nobs);
1882  17457 +           %if &_varclus_nobs > &VARCLUS_MAXNUMOBS
       %then %do;
1883  17458 +               proc datasets lib=work nolist;
1884  17459 +               delete  _newtrainds;
1885  17460 +               run;
1886  17461 +               quit;
1887  17462 +               %let  EMEXCEPTIONSTRING = exception.
      server.varclus.sample.warning;
1888  17463 +               %let error = 3;
1889  17464 +              %goto endtrain;
1890  17465 +           %end;
1891  17466 +        %end;
1892  17468 +        %if &EM_PROPERTY_TWOSTAGECLUS eq AUTO %then
      %do;
1893  17469 +            %getNVar(inds=_newtrainds, nvar=_nvar);
1894  17471 +           %if &_nvar > &VARCLUS_MAXNUMVAR %then %d
      o;
1895  17472 +               %let trainnum = 2;
1896  17473 +           %end;
1897  17474 +           %else %do;
```

```
1898 17475 +                %let trainnum = 1;
1899 17476 +             %end;
1900 17477 +          %end;
1901 17478 +     %end;
1902 17479 +     %else %do;
1903 17481 +           %if &EM_PROPERTY_SUPPRESSSAMPWARN eq N %th
     en %do;
1904 17483 +             %getNObs(inds=&EM_IMPORT_DATA, nobs=_var
     clus_nobs);
1905 17485 +             %if &_varclus_nobs > &VARCLUS_MAXNUMOBS
      %then %do;
1906 17487 +                  %let  EMEXCEPTIONSTRING = exception
     .server.varclus.sample.warning;
1907 17488 +                   %let error = 3;
1908 17489 +                  %goto endtrain;
1909 17490 +             %end;
1910 17491 +          %end;
1911 17494 +           %if &EM_PROPERTY_TWOSTAGECLUS eq AUTO %then
      %do;
1912 17495 +             /*%getNVar(inds=&EM_IMPORT_DATA, nvar=_n
     var);
1913 17496 +               %if &_nvar > &VARCLUS_MAXNUMVAR %then
     %do;*/
1914 17497 +             %if &EM_NUM_INTERVAL_INPUT > &VARCLUS_MA
     XNUMVAR %then %do;
1915 17498 +                 %let trainnum = 2;
1916 17499 +             %end;
1917 17500 +             %else %do;
1918 17501 +                 %let trainnum = 1;
1919 17502 +             %end;
1920 17503 +          %end;
1921 17505 +    %end;
1922 17507 +     %em_checkerror();
1923 17509 +     %if &EMEXCEPTIONSTRING ne %then %do;
1924 17510 +        %let error = 4;
1925 17511 +        %goto endtrain;
```

```
1926 17512 +    %end;
1927 17514 +    %CreateVarclusMeta(trainnum=&trainnum);
1928 17516 +    %if &error > 0 %then  %goto endtrain;
1929 17518 +       /* when only EM_PROPERTY_TWOSTAGECLUS eq AUTO
      &_triannum will be 1 or 2  */
1930 17520 +       %if (&trainnum = 1 ) or %upcase(&EM_PROPERTY_
     TWOSTAGECLUS) = NO %then %do;
1931 17521 +           %train1;
1932 17522 +       %end;
1933 17523 +       %if (&trainnum = 2 ) or %upcase(&EM_PROPERTY_
     TWOSTAGECLUS) = YES %then %do;
1934 17524 +           %train2;
1935 17525 +       %end;
1936 17527 +      filename _in  "&EM_FILE_EMFLOWSCORECODE";
1937 17528 +      filename _out "&EM_FILE_EMPUBLISHSCORECODE";
1938 17529 +      data _null_;
1939 17530 +         length line $20000;
1940 17531 +         file _out lrecl=20000;
1941 17532 +         fid=fopen("_in",'i',20000,'v');
1942 17533 +         if fid > 0  then do;
1943 17534 +            do while(^fread(fid));
1944 17535 +               rlen = frlen(fid);
1945 17536 +               rc= fget(fid,line,20000);
1946 17537 +               start = length(line)-length(left(line
     ))+1;
1947 17538 +               line=strip(line);
1948 17539 +               if line ne 'delete;' then do;
1949 17540 +                  put @start line;
1950 17541 +               end;
1951 17542 +            end;
1952 17543 +            if fid > 0  then rc=fclose(fid);
1953 17544 +         end;
1954 17545 +      run;
1955 17546 +      filename _in;
1956 17547 +      filename _out;
1957 17549 +   %endtrain:
```

```
1958  17552 +%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  17554 +%macro VarClus(indata=,
1963  17555 +              outstat=,
1964  17556 +              outtree=,
1965  17557 +              vars=,
1966  17558 +              freq=,
1967  17559 +              weight=,
1968  17560 +              includeclassvar=
1969  17561 +              );
1970  17563 +   %if &vars eq %then
1971  17564 +       %let vars = %EM_INTERVAL_INPUT %EM_INTERVAL_R
      EJECTED;
1972  17566 +        /*Add by ruzhan, May 28, 2013*/
1973  17567 +        ods graphics on;
1974  17568 +        %let odspath =;
1975  17569 +        data _null_;
1976  17570 +            path = pathname('WORK');
1977  17571 +            call symput('odspath', path);
1978  17572 +        run;
1979  17573 +        ods listing gpath="&odspath";
1980  17574 +        ods output DENDROGRAM=&outtree;
1981  17575 +       proc varclus data = &indata outstat= &outstat
1982  17576 +        %if %upcase(&EM_PROPERTY_CLUSCOMP) eq CENTRO
      ID %then %do; centroid %end;
1983  17577 +        %if %upcase(&EM_PROPERTY_CLUSSOURCE) eq COV
      %then %do; cov %end;
1984  17578 +        %if %upcase(&EM_PROPERTY_CLUSHIERACHY) eq Y
      %then %do; hi %end;
1985  17579 +        /*-----------------------
1986  17580 +        %if %upcase(&EM_PROPERTY_MAXCLUS) ne DEFAULT
       %then %do;
1987  17581 +            %let maxc = %sysevalf(&EM_PROPERTY_MAXCL
```

```
     US, int);
1988 17582 +            %if &maxc > 1 %then %do;     maxc=&maxc
     %end;
1989 17583 +       %end;
1990 17584 +       -------------------------*/
1991 17585 +       %if (&EM_PROPERTY_MAXCLUS ne . )  and (%upcas
     e(&EM_PROPERTY_MAXCLUS) ne DEFAULT) %then %do;
1992 17586 +            maxc = &EM_PROPERTY_MAXCLUS
1993 17587 +       %end;
1994 17589 +       /*-----------------------------------------
     ------------------------------------------------------
1995 17590 +        (%upcase(&EM_PROPERTY_MAXCLUS) ne DEFAULT) i
     s needed because of diagram conversion from Em12.1 to 12.3
1996 17591 +        ----------------------------------------
     ------------------------------------------------------*
     /
1997 17593 +       %if (%upcase(&EM_PROPERTY_CLUSCOMP) ne CENTRO
     ID) and
1998 17594 +           ( ( &EM_PROPERTY_MAXEIGEN ne . )  and (%u
     pcase(&EM_PROPERTY_MAXEIGEN) ne DEFAULT) ) %then %do;
1999 17595 +            maxeigen = &EM_PROPERTY_MAXEIGEN
2000 17596 +       %end;
2001 17598 +       %if &EM_PROPERTY_PROPORTION ne 0  and (%upcas
     e(&EM_PROPERTY_PROPORTION) ne DEFAULT) %then %do;
2002 17599 +            proportion = &EM_PROPERTY_PROPORTION
2003 17600 +       %end;
2004 17601 +       %if %upcase(&EM_PROPERTY_PRINTOPTION) eq SUMM
     ARY %then %do;
2005 17602 +               summary
2006 17603 +       %end;
2007 17605 +       %if %upcase(&EM_PROPERTY_PRINTOPTION) eq SHOR
     T %then %do;
2008 17606 +           short
2009 17607 +       %end;
2010 17608 +        %if %upcase(&EM_PROPERTY_PRINTOPTION) eq ALL
      %then %do;
```

```
2011 17609 +           corr trace simple
2012 17610 +        %end;
2013 17611 +        %if %upcase(&EM_PROPERTY_PRINTOPTION) eq NONE
      %then %do;
2014 17612 +          noprint
2015 17613 +        %end;
2016 17614 +         /*Comment out by ruzhan, May 28, 2013*/
2017 17615 +        /*       outtree=&outtree*/
2018 17616 +        ;
2019 17617 +         var %EM_INTERVAL_INPUT %EM_INTERVAL_REJECTED
2020 17619 +       %if &includeclassvar eq Y %then %do;
2021 17620 +          %let dsid = %sysfunc(open(&EM_USER_OUTDUMM
      Y));
2022 17621 +          %if &dsid > 0 %then %do;
2023 17622 +             %let nvar = %sysfunc(attrn(&dsid, NVAR)
      );
2024 17623 +              %do i = 2 %to &nvar;
2025 17624 +                  %let varname = %sysfunc(varname(&d
      sid, &i));
2026 17625 +                   &varname
2027 17626 +              %end;
2028 17628 +          %end;
2029 17629 +       %end;
2030 17631 +       ;
2031 17632 +       /*------------------------
2032 17633 +       %if &freq ne  %then %do;
2033 17634 +          freq &freq;
2034 17635 +       %end;
2035 17636 +       %if &weight ne  %then %do;
2036 17637 +          weight &weight;
2037 17638 +       %end;
2038 17639 +       ----------------------------*/
2039 17640 +       %if %EM_FREQ ne %then %do;
2040 17641 +           Freq %EM_FREQ;
2041 17642 +       %end;
2042 17643 +       ;
```

```
2043 17644 +    run;
2044 17645 +    ods graphics off;
2045 17646 +    %endline:
2046 17647 +    quit;
2047 17648 +%mend VarClus;
2048 17650 +%macro Train1;
2049 17652 +   %EM_GETNAME(key=OUTCORRPLOT, type=DATA) ;
2050 17653 +   %EM_GETNAME(key=OUTCORR, type=DATA) ;
2051 17654 +   %EM_GETNAME(key=OUTSTAT, type=DATA) ;
2052 17655 +   %EM_GETNAME(key=OUTGROUP, type=DATA) ;
2053 17656 +   %EM_GETNAME(key=OUTSTATPLOT, type=DATA) ;
2054 17657 +   %EM_GETNAME(key=OUTTREE, type=DATA) ;
2055 17658 +   %EM_GETNAME(key=OUTRSQUARE, type=DATA) ;
2056 17659 +   %EM_GETNAME(key=OUTCLUSSTRUCT, type=DATA) ;
2057 17660 +   %EM_GETNAME(key=OUTCLUSCORR, type=DATA) ;
2058 17661 +   %EM_GETNAME(key=OUTCLUSCORRPLOT, type=DATA) ;
2059 17662 +   %EM_GETNAME(key=OUTCLUSRSQUARE, type=DATA) ;
2060 17663 +   %EM_GETNAME(key=OUTLINK, type=DATA) ;
2061 17664 +   %EM_GETNAME(key=OUTNODE, type=DATA) ;
2062 17666 +   /*
2063 17667 +     %let VARCLUS_MAXNUMOBS = 100;
2064 17668 +     %let VARCLUS_MAXNUMVAR = 200;
2065 17669 +   */
2066 17671 +   /* Train
2067 17673 +   %if &EM_IMPORT_DATA eq %then %do;
2068 17674 +       %let  EMEXCEPTIONSTRING = exception.server.IM
     PORT.NOTRAIN,1;
2069 17675 +       %goto endtrain1;
2070 17676 +   %end;
2071 17678 +   %if  &EM_NUM_INTERVAL_INPUT < 2 %then %do;
2072 17679 +        %let  EMEXCEPTIONSTRING = exception.server.M
     ETADATA.USEATLEAST2INPUTREJECT;
2073 17680 +        %goto endtrain1;
2074 17681 +   %end;
2075 17682 +   */
2076 17683 +   %if &EM_PROPERTY_INCLUDECLASSVAR eq Y %then %do;
```

```
2077 17684 +       /* moved this to pretrain ------------
2078 17686 +        %EM_GETNAME(key=OUTDUMMY, type=DATA);
2079 17687 +        %MakeDummyVariables( indata=&EM_IMPORT_DATA,
2080 17688 +                             outvar=&EM_USER_OUTDUMM
     Y,
2081 17689 +                             outdata=_newtrainds,
2082 17690 +                             fileref=&EM_FILE_EMFLOW
     SCORECODE);
2083 17692 +        %if &EM_PROPERTY_SUPPRESSSAMPWARN eq N %then
      %do;
2084 17693 +            %getNVarNObs(inds=_newtrainds, nvar=_nva
     r, nobs=_nobs);
2085 17694 +            %if &_nobs > &VARCLUS_MAXNUMOBS ne %then
      %do;
2086 17695 +                %let  EMEXCEPTIONSTRING = exception.
     server.varclus.sample.warning
2087 17696 +                %goto endtrain;
2088 17697 +            %end;
2089 17698 +        %end;
2090 17700 +        +--------------------------------------*/
2091 17702 +        %if (%sysfunc(exist(_newtrainds)) or %sysfun
     c(exist(_newtrainds, VIEW))) < 1 %then %do;
2092 17703 +        /*when there is no class var and &EM_PROPERT
     Y_INCLUDECLASSVAR eq Y
2093 17704 +          There is no _newtrainds */
2094 17705 +         %VarClus( indata=&EM_IMPORT_DATA,
2095 17706 +                  outstat=&EM_USER_OUTSTAT,
2096 17707 +                  outtree = &EM_USER_OUTTREE,
2097 17708 +                  includeclassvar=N
2098 17709 +                 );
2099 17710 +        %end;
2100 17711 +        %else %do;
2101 17712 +        %VarClus( indata=_newtrainds,
2102 17713 +                  outstat=&EM_USER_OUTSTAT,
2103 17714 +                  outtree = &EM_USER_OUTTREE,
2104 17715 +                  includeclassvar=Y
```

```
2105 17716 +                );
2106 17717 +         proc datasets lib=work nolist;
2107 17718 +              delete  _newtrainds;
2108 17719 +         run;
2109 17720 +         %end;
2110 17721 +    %end;
2111 17722 +    %else %do;
2112 17724 +        /*----- moved this to pretrain ------------
2113 17725 +        %if &EM_PROPERTY_SUPPRESSSAMPWARN eq N %then
      %do;
2114 17726 +            %getNVarNObs(inds=&EM_IMPORT_DATA, nvar=
     _nvar, nobs=_nobs);
2115 17727 +            %if &_nobs > &VARCLUS_MAXNUMOBS ne %then
      %do;
2116 17728 +                %let  EMEXCEPTIONSTRING = exception
     .server.varclus.sample.warning;
2117 17729 +                %goto endtrain;
2118 17730 +            %end;
2119 17731 +        %end;
2120 17732 +        +--------------------------------------*/
2121 17734 +        %VarClus(indata=&EM_IMPORT_DATA,
2122 17735 +                 outstat=&EM_USER_OUTSTAT,
2123 17736 +                 outtree =&EM_USER_OUTTREE,
2124 17737 +                 includeclassvar=N
2125 17738 +               );
2126 17740 +   %end;
2127 17742 +   %em_checkerror();
2128 17744 +   %if &EMEXCEPTIONSTRING ne %then %do;
2129 17745 +       %goto endtrain1;
2130 17746 +   %end;
2131 17747 +    /*
2132 17748 +    %FindClusNum(statds=&EM_USER_OUTSTAT, groupds=&E
     M_USER_OUTGROUP, minvariation=&EM_PROPERTY_MINVARIATION);
2133 17749 +    */
2134 17751 +    %getNclusfromTrain(inoutstat=&EM_USER_OUTSTAT, n
     c=_nclus);
```

```
2135 17753 +    %let gid=;
2136 17755 +    /*--- To make the main result table ------------
     ----------*/
2137 17757 +    %MakeInterClusCorrData(indata=&EM_USER_OUTSTAT,
     globalclusid=&gid, outdata=&EM_USER_OUTCLUSCORR, ncluster=&
     _nclus, RSquare=Y);
2138 17758 +    %FindNextClosestClusByCluster(indata=&EM_USER_OU
     TCLUSCORR, outdata=_nextClusRSq, globalclusid=&gid, ncluste
     r=&_nclus);
2139 17759 +    %MakeOwnRSquare(indata=&EM_USER_OUTSTAT, globalc
     lusid=&gid, outdata=_ownRsq, ncluster=&_nclus);
2140 17760 +    %MakeClusStructCorrData(indata=&EM_USER_OUTSTAT,
      globalclusid=&gid, outdata=&EM_USER_OUTCLUSSTRUCT , nclust
     er=&_nclus, Rsquare=Y);
2141 17761 +    %FindNextClosestClusByVar(indata=&EM_USER_OUTCLU
     SSTRUCT, outdata=_nextVarRSq, globalclusid=&gid, ncluster=&
     _nclus);
2142 17762 +    %MakeVarClusResultTable(indata1=_ownRsq, indata2
     =_nextVarRsq, indata3=_nextClusRSq, outdata=&EM_USER_OUTRSQ
     UARE, globalclusid=&gid, ncluster=&_nclus, selectedcomp=&EM
     _PROPERTY_EXPORTEDCOMP);
2143 17764 +   /* Inter Cluster Correlation Plot -------------
     ---------*/
2144 17766 +    %MakeInterClusCorrData(indata=&EM_USER_OUTSTAT,
     globalclusid=&gid, outdata=&EM_USER_OUTCLUSCORRPLOT, nclust
     er=&_nclus, RSquare=N);
2145 17767 +    %MakePlotDataFromCorrTable(indata=&EM_USER_OUTCL
     USCORRPLOT, outdata=&EM_USER_OUTCLUSCORRPLOT, globalclusid=
     &gid);
2146 17769 +   /* Create some other data sets for results ----*/
2147 17771 +   %MakeStatPlotData(statds=&EM_USER_OUTSTAT , outst
     atplotds=&EM_USER_OUTSTATPLOT);
2148 17772 +   %MakeVarClusCorrData(statds=&EM_USER_OUTSTAT, cor
     rds=&EM_USER_OUTCORR, corrplotds=&EM_USER_OUTCORRPLOT);
2149 17773 +   %MakeClusConstellData(indata=&EM_USER_OUTRSQUARE,
      outlink=&EM_USER_OUTLINK, outnode=&EM_USER_OUTNODE);
```

```
2150 17775 +  /*---- move this to Score action ------------
2151 17776 +   %MakeDeltaCode(groupds=&EM_USER_OUTRSQUARE, Delta
     CodeFile=&EM_FILE_CDELTA_TRAIN);
2152 17777 +   */
2153 17779 +   /* create score code at training time */
2154 17780 +   %CreateScoreCode(indata=&EM_USER_OUTSTAT, nclusco
     mp=&_nclus, fileref=&EM_FILE_EMFLOWSCORECODE);
2155 17781 +   %endtrain1:
2156 17782 +%mend Train1;
2157 NOTE: %INCLUDE (level 1) ending.
2158 NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMEXPL.V
     ARIABLECLUSTERING_TRAIN2.SOURCE.
2159 17785 +%macro VarClus2( indata=,
2160 17786 +                 outdata=,
2161 17787 +                 outstat =outstat,
2162 17788 +                 outrsquare=outrsquare,
2163 17789 +                 outtree = outtree,
2164 17790 +                 outclusstruct = outclusstruct,
2165 17791 +                 outcluscorr =outcluscorr,
2166 17792 +                 outcluscorrplot =outcluscorrplot,
2167 17793 +                 outclusrsquare =outclusrsquare,
2168 17794 +                 outstatplot = outstatplot,
2169 17795 +                 outcorr = outcorr,
2170 17796 +                 outcorrplot = outcorrplot,
2171 17797 +                 clusdata=,
2172 17798 +                 outnode=outnode,
2173 17799 +                 outlink=outlink,
2174 17800 +                 ngroup =1  );
2175 17801 +         %if (%sysfunc(exist(&indata)) or %sysfunc(e
     xist(&indata, VIEW))) < 1 %then %do;
2176 17802 +              %let  EMEXCEPTIONSTRING = exception.se
     rver.IMPORT.NOTRAIN,1;
2177 17803 +              %goto endline;
2178 17804 +          %end;
2179 17806 +        %global numglobalcluster;
2180 17807 +        %let  numglobalcluster = &ngroup;
```

```
2181 17808 +        %let  gscorefile =  %bquote(&EM_NODEDIR)&EM_
     DSEP.gclusterscore.sas;
2182 17809 +        %do vci =1 %to &ngroup;
2183 17810 +           data tmp_clusdata;
2184 17811 +                set &clusdata;
2185 17812 +                if CLUSTER ne &vci then delete;
2186 17813 +           run;
2187 17814 +           %let outstatds = &outstat.&vci;
2188 17815 +           %let outrsquareds = &outrsquare.&vci;
2189 17816 +           %let outtreeds = &outtree.&vci;
2190 17817 +           %let outclusstuctds = &outclusstruct.&vci
     ;
2191 17818 +           %let outcluscorrds = &outcluscorr.&vci;
2192 17819 +           %let outcluscorrplotds = &outcluscorrplot
     .&vci;
2193 17820 +           %let outclusrsquareds = &outclussquare.&v
     ci;
2194 17821 +           %let outstatplotds = &outstatplot.&vci;
2195 17822 +           %let outcorrds = &outcorr.&vci;
2196 17823 +           %let outcorrplotds = &outcorrplot.&vci;
2197 17824 +           %let outnodeds = &outnode.&vci;
2198 17825 +           %let outlinkds = &outlink.&vci;
2199 17827 +           %EM_REGISTER(KEY=&outstatds, TYPE=DATA);
2200 17828 +           %EM_GETNAME(KEY=&outstatds, TYPE=DATA);
2201 17829 +           %let emuseroutstat = &EM_USER_&outstatds;
2202 17831 +           %EM_REGISTER(KEY=&outtreeds, TYPE=DATA);
2203 17832 +           %EM_GETNAME(KEY=&outtreeds, TYPE=DATA);
2204 17833 +           %let emuserouttree = &EM_USER_&outtreeds;
2205 17835 +           %EM_REGISTER(KEY=&outcluscorrds, TYPE=DAT
     A);
2206 17836 +           %EM_GETNAME(KEY=&outcluscorrds, TYPE=DATA
     );
2207 17837 +           %let emuseroutcluscorr = &EM_USER_&outclu
     scorrds;
2208 17839 +           %EM_REGISTER(KEY=&outclusstuctds, TYPE=DA
     TA);
```

```
2209  17840 +           %EM_GETNAME(KEY=&outclusstuctds, TYPE=DAT
      A);
2210  17841 +           %let emuseroutclusstruct = &EM_USER_&outc
      lusstuctds;
2211  17843 +           %EM_REGISTER(KEY=&outrsquareds, TYPE=DATA
      );
2212  17844 +           %EM_GETNAME(KEY=&outrsquareds, TYPE=DATA)
      ;
2213  17845 +           %let emuseroutrsquare = &EM_USER_&outrsqu
      areds;
2214  17847 +           %EM_REGISTER(KEY=&outcluscorrplotds, TYPE
      =DATA);
2215  17848 +           %EM_GETNAME(KEY=&outcluscorrplotds, TYPE=
      DATA);
2216  17849 +           %let emuseroutcluscorrplot = &EM_USER_&ou
      tcluscorrplotds;
2217  17851 +           %EM_REGISTER(KEY=&outstatplotds, TYPE=DAT
      A);
2218  17852 +           %EM_GETNAME(KEY=&outstatplotds, TYPE=DATA
      );
2219  17853 +           %let emuseroutstatplot = &EM_USER_&outsta
      tplotds;
2220  17855 +           %EM_REGISTER(KEY=&outcorrds, TYPE=DATA);
2221  17856 +           %EM_GETNAME(KEY=&outcorrds, TYPE=DATA);
2222  17857 +           %let emuseroutcorr = &EM_USER_&outcorrds;
2223  17859 +           %EM_REGISTER(KEY=&outcorrplotds, TYPE=DAT
      A);
2224  17860 +           %EM_GETNAME(KEY=&outcorrplotds, TYPE=DATA
      );
2225  17861 +           %let emuseroutcorrplot = &EM_USER_&outcor
      rplotds;
2226  17863 +           %EM_REGISTER(KEY=&outnodeds, TYPE=DATA);
2227  17864 +           %EM_GETNAME(KEY=&outnodeds, TYPE=DATA);
2228  17865 +           %let emuseroutnode = &EM_USER_&outnodeds;
2229  17867 +           %EM_REGISTER(KEY=&outlinkds, TYPE=DATA);
2230  17868 +           %EM_GETNAME(KEY=&outlinkds, TYPE=DATA);
```

```
2231 17869 +           %let emuseroutlink = &EM_USER_&outlinkds;
2232 17871 +           /*Add by ruzhan, May 28, 2013*/
2233 17872 +           ods graphics on;
2234 17873 +            %let odspath =;
2235 17874 +            data _null_;
2236 17875 +                path = pathname('WORK');
2237 17876 +                call symput('odspath', path);
2238 17877 +            run;
2239 17878 +            ods listing gpath="&odspath";
2240 17880 +           ods output DENDROGRAM=&emuserouttree;
2241 17882 +           proc varclus data=&indata outstat=&emuser
     outstat
2242 17883 +            %if %upcase(&EM_PROPERTY_CLUSCOMP) eq CE
     NTROID %then %do; centroid %end;
2243 17884 +            %if %upcase(&EM_PROPERTY_CLUSSOURCE) eq
     COV %then %do; cov %end;
2244 17885 +            %if %upcase(&EM_PROPERTY_CLUSHIERACHY) e
     q Y %then %do; hi %end;
2245 17886 +            %if (&EM_PROPERTY_MAXCLUS ne .)  and (%u
     pcase(&EM_PROPERTY_MAXCLUS) ne DEFAULT) %then %do;
2246 17887 +                  maxc=&EM_PROPERTY_MAXCLUS
2247 17888 +           %end;
2248 17890 +            %if (%upcase(&EM_PROPERTY_CLUSCOMP) ne C
     ENTROID) and
2249 17891 +                ( (&EM_PROPERTY_MAXEIGEN ne .)  and
     (%upcase(&EM_PROPERTY_MAXEIGEN) ne DEFAULT)) %then %do;
2250 17892 +            maxeigen = &EM_PROPERTY_MAXEIGEN
2251 17893 +            %end;
2252 17895 +           %if (&EM_PROPERTY_PROPORTION ne 0)  and (
     %upcase(&EM_PROPERTY_PROPORTION) ne DEFAULT) %then %do;
2253 17896 +                proportion = &EM_PROPERTY_PROPORTION
2254 17897 +           %end;
2255 17898 +           %if %upcase(&EM_PROPERTY_PRINTOPTION) eq
     SUMMARY %then %do;
2256 17899 +               summary
2257 17900 +           %end;
```

```
2258 17901 +           %if %upcase(&EM_PROPERTY_PRINTOPTION) eq
     SHORT %then %do;
2259 17902 +               short
2260 17903 +           %end;
2261 17904 +            %if %upcase(&EM_PROPERTY_PRINTOPTION) eq
      ALL %then %do;
2262 17905 +               corr trace simple
2263 17906 +           %end;
2264 17907 +           %if %upcase(&EM_PROPERTY_PRINTOPTION) eq
     NONE %then %do;
2265 17908 +               noprint
2266 17909 +           %end;
2267 17910 +            /*Comment out by ruzhan, May 28, 2013*/
2268 17911 +              /*outtree=&emuserouttree*/
2269 17912 +           ;
2270 17913 +           var
2271 17914 +           %let dsid=%sysfunc(open(work.tmp_clusdata
     ));
2272 17915 +           %let vn_name =%sysfunc(varnum(&dsid, VARI
     ABLE));
2273 17916 +                %do %while(^%sysfunc(fetch(&dsid)));
2274 17917 +                    %let _varname= %sysfunc(getvarc(
     &dsid, &vn_name));
2275 17918 +                    &_varname
2276 17919 +                %end;
2277 17920 +            %let dsid = %sysfunc(close(&dsid));
2278 17921 +           ;
2279 17922 +           %if %EM_FREQ ne %then %do;
2280 17923 +           Freq %EM_FREQ;
2281 17924 +           %end;
2282 17925 +           run;
2283 17926 +            ods graphics off;
2284 17927 +            %getNclusfromTrain(inoutstat=&emuserouts
     tat, nc=_nclus);
2285 17929 +            %let gid=&vci;
2286 17930 +            /*--- To make the main result table ----
```

```
     ------------------*/
2287 17931 +           %MakeInterClusCorrData(indata=&emuserout
     stat, globalclusid=&gid, outdata=&emuseroutcluscorr, nclust
     er=&_nclus, RSquare=Y);
2288 17932 +           %FindNextClosestClusByCluster(indata=&em
     useroutcluscorr, outdata=_nextClusRSq, globalclusid=&gid, n
     cluster=&_nclus);
2289 17933 +           %MakeOwnRSquare(indata=&emuseroutstat, g
     lobalclusid=&gid, outdata=_ownRsq, ncluster=&_nclus);
2290 17934 +           %MakeClusStructCorrData(indata=&emuserou
     tstat, globalclusid=&gid, outdata=&emuseroutclusstruct , nc
     luster=&_nclus, Rsquare=Y);
2291 17935 +           %FindNextClosestClusByVar(indata=&emuser
     outclusstruct, outdata=_nextVarRSq, globalclusid=&gid, nclu
     ster=&_nclus);
2292 17936 +           %MakeVarClusResultTable(indata1=_ownRsq,
      indata2=_nextVarRsq, indata3=_nextClusRSq, outdata=&emuser
     outrsquare, globalclusid=&gid, ncluster=&_nclus, selectedco
     mp=&EM_PROPERTY_EXPORTEDCOMP);
2293 17938 +           /* Inter Cluster Correlation Plot ------
     ------------------*/
2294 17940 +           %MakeInterClusCorrData(indata=&emuserout
     stat, globalclusid=&gid, outdata=&emuseroutcluscorrplot, nc
     luster=&_nclus, RSquare=N);
2295 17941 +           %MakePlotDataFromCorrTable(indata=&emuse
     routcluscorrplot, outdata=&emuseroutcluscorrplot, globalclu
     sid=&gid);
2296 17943 +          /* Create some other data sets for result
     s ---------------*/
2297 17945 +          %MakeStatPlotData(statds=&emuseroutstat ,
      outstatplotds=&emuseroutstatplot);
2298 17946 +          %MakeVarClusCorrData(statds=&emuseroutsta
     t, corrds=&emuseroutcorr, corrplotds=&emuseroutcorrplot);
2299 17947 +          %MakeClusConstellData(indata=&emuseroutrs
     quare, outlink=&emuseroutlink, outnode=&emuseroutnode);
2300 17949 +          %CreateScoreCode2(indata=&emuseroutstat,
```

```
     ncluscomp=&_nclus, globalclusid=&gid, fileref=&EM_FILE_EMFL
     OWSCORECODE);
2301 17951 +          %CreateGClusterScoreCode(indata=&emuserou
     tstat,  globalclusid=&gid, fileref=&gscorefile);
2302 17953 +          %if &vci = 1 %then %do;
2303 17954 +          data &EM_USER_OUTRSQUARE;
2304 17955 +               set &emuseroutrsquare;
2305 17956 +          run;
2306 17957 +          %end;
2307 17958 +          %else %do;
2308 17959 +          data &EM_USER_OUTRSQUARE;
2309 17960 +               set &EM_USER_OUTRSQUARE &emuseroutrs
     quare;
2310 17961 +          run;
2311 17962 +          %end;
2312 17963 +          quit;
2313 17964 +       %end;
2314 17966 +       /*---- move this to Score action ------------
2315 17967 +       %MakeDeltaCode2(groupds=&EM_USER_OUTRSQUARE,
     DeltaCodeFile=&EM_FILE_CDELTA_TRAIN);
2316 17968 +       */
2317 17970 +       /* Make Score Gcluster component & correlatio
     n  */
2318 17971 +       %MakeGClusterCorrelation(Indata=&indata, ngcl
     uster=&ngroup, gscorecode=&gscorefile, outrsquare=&EM_USER_
     OUTRSQUARE);
2319 17973 +       quit;
2320 17974 +       %endline:
2321 17975 +%mend VarClus2;
2322 17978 +%macro Train2;
2323 17980 +   %EM_GETNAME(key=OUTCORRPLOT, type=DATA) ;
2324 17981 +   %EM_GETNAME(key=OUTCORR, type=DATA) ;
2325 17982 +   %EM_GETNAME(key=OUTSTAT, type=DATA) ;
2326 17983 +   %EM_GETNAME(key=OUTGROUP, type=DATA) ;
2327 17984 +   %EM_GETNAME(key=OUTSTATPLOT, type=DATA) ;
2328 17985 +   %EM_GETNAME(key=OUTTREE, type=DATA) ;
```

```
2329 17986 +   %EM_GETNAME(key=OUTRSQUARE, type=DATA) ;
2330 17987 +   %EM_GETNAME(key=OUTCLUSSTRUCT, type=DATA) ;
2331 17988 +   %EM_GETNAME(key=OUTCLUSCORR, type=DATA) ;
2332 17989 +   %EM_GETNAME(key=OUTCLUSCORRPLOT, type=DATA) ;
2333 17990 +   %EM_GETNAME(key=OUTCLUSRSQUARE, type=DATA) ;
2334 17991 +   %EM_GETNAME(key=GOUTSTAT, type=DATA) ;
2335 17992 +   %EM_GETNAME(key=GOUTTREE, type=DATA) ;
2336 17993 +   %EM_GETNAME(key=GOUTCORR, type=DATA) ;
2337 17994 +   %EM_GETNAME(key=GOUTGROUP, type=DATA) ;
2338 17995 +   %EM_GETNAME(key=GOUTNODE, type=DATA) ;
2339 17996 +   %EM_GETNAME(key=GOUTLINK, type=DATA) ;
2340 17997 +   %EM_GETNAME(key=VARCLUSMETA, type=DATA) ;
2341 18001 +    /* Train */
2342 18003 +   /* moved to pretrain ------------
2343 18005 +   %if &EM_IMPORT_DATA eq %then %do;
2344 18006 +       %let  EMEXCEPTIONSTRING = exception.server.IM
     PORT.NOTRAIN,1;
2345 18007 +       %goto endtrain2;
2346 18008 +   %end;
2347 18010 +   %if  &EM_NUM_INTERVAL_INPUT < 2 %then %do;
2348 18011 +        %let  EMEXCEPTIONSTRING = exception.server.M
     ETADATA.USEATLEAST2INPUTREJECT;
2349 18012 +        %goto endtrain2;
2350 18013 +   %end;
2351 18014 +   +-----------------------------------------*/
2352 18016 +    %let _num_input_interval = %eval(&EM_NUM_INTERVA
     L_INPUT+&EM_NUM_INTERVAL_REJECTED);
2353 18019 +    %if &EM_PROPERTY_INCLUDECLASSVAR eq Y %then %do;
2354 18021 +        /*----- moved to pretrain ------------
2355 18023 +        %EM_GETNAME(key=OUTDUMMY, type=DATA);
2356 18024 +        %MakeDummyVariables( indata=&EM_IMPORT_DATA,
2357 18025 +                             outvar=&EM_USER_OUTDUMM
     Y,
2358 18026 +                             outdata=_newtrainds,
2359 18027 +                             fileref=&EM_FILE_EMFLOW
     SCORECODE,
```

```
2360 18028 +                                ndummyvars=_ndummyvars
2361 18029 +                              );
2362 18031 +        %if &EM_PROPERTY_SUPPRESSSAMPWARN eq N %then
      %do;
2363 18032 +           %getNVarNObs(inds=_newtrainds, nvar=_nva
     r, nobs=_nobs);
2364 18033 +           %if &_nobs > &VARCLUS_MAXNUMOBS  %then %
     do;
2365 18034 +              %let  EMEXCEPTIONSTRING = exception.
     server.varclus.sample.warning
2366 18035 +              %goto endtrain;
2367 18036 +           %end;
2368 18037 +        %end;
2369 18039 +        +--------------------------------------*/
2370 18040 +        %if (%sysfunc(exist(_newtrainds)) or %sysfun
     c(exist(_newtrainds, VIEW))) < 1 %then %do;
2371 18041 +        /*when there is no class var and &EM_PROPERT
     Y_INCLUDECLASSVAR eq Y
2372 18042 +          There is no _newtrainds */
2373 18043 +        %MakeCorrelation(    indata=&EM_IMPORT_DATA,
2374 18044 +                             outstat=&EM_USER_GOUTST
     AT,
2375 18045 +                             corrmatrix=Y,
2376 18046 +                             outcorr=&EM_USER_GOUTCO
     RR,
2377 18047 +                             includeclassvar=N,
2378 18048 +                             freq =%EM_FREQ
2379 18049 +                           );
2380 18050 +        %getInitialGClusterNumber(ninput=&_num_input
     _interval, ngc=_ngc);
2381 18052 +        %end;
2382 18053 +        %else %do;
2383 18054 +         %MakeCorrelation(   indata=_newtrainds,
2384 18055 +                             outstat=&EM_USER_GOUTST
     AT,
2385 18056 +                             corrmatrix=Y,
```

```
2386 18057 +                              outcorr=&EM_USER_GOUTCO
     RR,
2387 18058 +                              includeclassvar=Y,
2388 18059 +                              freq =%EM_FREQ
2389 18060 +                            );
2390 18061 +        %getInitialGClusterNumber(ninput=&_num_input
     _interval, ndummy=&_ndummyvars, ngc=_ngc);
2391 18063 +        %end;
2392 18065 +    %end;
2393 18066 +    %else %do;
2394 18068 +        /* moved to pretrain -------------------
2395 18070 +         %if &EM_PROPERTY_SUPPRESSSAMPWARN eq N %the
     n %do;
2396 18071 +            %getNVarNObs(inds=&EM_IMPORT_DATA, nvar=
     _nvar, nobs=_nobs);
2397 18072 +            %if &_nobs > &VARCLUS_MAXNUMOBS  %then %
     do;
2398 18073 +                %let  EMEXCEPTIONSTRING = exception.
     server.varclus.sample.warning
2399 18074 +                %goto endtrain;
2400 18075 +            %end;
2401 18076 +        %end;
2402 18077 +         +---------------------------------------
     */
2403 18079 +        %MakeCorrelation( indata=&EM_IMPORT_DATA,
2404 18080 +                          outstat=&EM_USER_GOUTSTAT,
2405 18081 +                          corrmatrix=Y,
2406 18082 +                          outcorr=&EM_USER_GOUTCORR,
2407 18083 +                          includeclassvar=N,
2408 18084 +                          freq =%EM_FREQ
2409 18085 +                          );
2410 18086 +        %getInitialGClusterNumber(ninput=&_num_input
     _interval, ngc=_ngc);
2411 18087 +    %end;
2412 18089 +   %UpdateOutStatCorrToDistance(indata=&EM_USER_GOUT
     STAT, /* indata should be a outstat from proc varclus */
```

```
2413 18090 +                                rsquare = N
2414 18091 +                                );
2415 18092 +   %HierClusWithCorr( indata=&EM_USER_GOUTSTAT ,
2416 18093 +                     ncluster=&_ngc,
2417 18094 +                     method = Ward,
2418 18095 +                     outtree =&EM_USER_GOUTTREE,
2419 18096 +                     idvar = _VAR_,
2420 18097 +                     outdata=&EM_USER_GOUTGROUP,
2421 18098 +                     rescore = N,
2422 18099 +                     newncluster=_newnclus
2423 18100 +                       );
2424 18102 +   /* store the number of global cluster */
2425 18104 +   data &EM_USER_VARCLUSMETA;
2426 18105 +        set &EM_USER_VARCLUSMETA;
2427 18106 +        NGCluster = &_newnclus;
2428 18107 +   run;
2429 18109 +   %MakeGobalConstellData(indata=&EM_USER_GOUTGROUP,
      outlink=&EM_USER_GOUTLINK, outnode=&EM_USER_GOUTNODE);
2430 18111 +   %if &EM_PROPERTY_INCLUDECLASSVAR eq Y %then %do;
2431 18112 +        %if (%sysfunc(exist(_newtrainds)) or %sysfun
      c(exist(_newtrainds, VIEW))) < 1 %then %do;
2432 18113 +           %VarClus2(indata=&EM_IMPORT_DATA,
2433 18114 +                clusdata=&EM_USER_GOUTGROUP,
2434 18115 +                ngroup=&_newnclus);
2435 18116 +         %end;
2436 18117 +         %else %do;
2437 18118 +            %VarClus2(indata=_newtrainds,
2438 18119 +                clusdata=&EM_USER_GOUTGROUP,
2439 18120 +                ngroup=&_newnclus);
2440 18122 +         %end;
2441 18124 +   %end;
2442 18125 +   %else %do;
2443 18126 +       %VarClus2(indata=&EM_IMPORT_DATA,
2444 18127 +                clusdata=&EM_USER_GOUTGROUP,
2445 18128 +                ngroup=&_newnclus);
2446 18129 +   %end;
```

```
2447 18131 +   %em_checkerror();
2448 18132 +   %if &EMEXCEPTIONSTRING ne %then %do;
2449 18133 +       %put &em_codebar;
2450 18134 +       %put Error: Variable clustering failed.;
2451 18135 +       %put &em_codebar;
2452 18136 +       %goto endtrain2;
2453 18137 +   %end;
2454 18138 +   /*
2455 18139 +    %FindClusNum(statds=&EM_USER_OUTSTAT, groupds=&E
     M_USER_OUTGROUP, minvariation=&EM_PROPERTY_MINVARIATION);
2456 18140 +   */
2457 18141 +   %endtrain2:
2458 18142 +%mend Train2;
2459 NOTE: %INCLUDE (level 1) ending.
2460 NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMUTIL.E
     M_COPYFILE.SOURCE.
2461 18144 +%macro em_copyfile(infref=, outfref=, append=N);
2462 18145 +
2463 18146 +   %if %sysfunc(fileref(&infref))=0 and %sysfunc(fil
     eref(&outfref))<=0 %then %do;
2464 18147 +
2465 18148 +       %let omode =;
2466 18149 +       %if &append=Y %then
2467 18150 +           %let omode = MOD;
2468 18151 +       data _null_;
2469 18152 +          length line $20000;
2470 18153 +          file &outfref &omode lrecl=20000;
2471 18154 +          fid=fopen("&infref",'i',20000,'v');
2472 18155 +          if fid > 0  then do;
2473 18156 +             do while(^fread(fid));
2474 18157 +                rlen = frlen(fid);
2475 18158 +                rc= fget(fid,line,20000);
2476 18159 +                start = length(line)-length(left(lin
     e))+1;
2477 18160 +                line=strip(line);
2478 18161 +                put @start line;
```

```
2479 18162 +             end;
2480 18163 +             if fid > 0  then rc=fclose(fid);
2481 18164 +          end;
2482 18165 +       run;
2483 18166 +
2484 18167 +   %end;
2485 18168 +%mend em_copyfile;
2486 NOTE: %INCLUDE (level 1) ending.
2487 NOTE: Fileref TEMP has been deassigned.
2488
2489 18169  %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              24713.31k
2495       OS Memory           34936.00k
2496       Timestamp           07/01/2024 05:54:44 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           0.10 seconds
2512       user cpu time       0.05 seconds
2513       system cpu time     0.06 seconds
```

```
2514        memory              159533.12k
2515        OS Memory           169592.00k
2516        Timestamp           07/01/2024 05:54:44 AM
2517        Step Count                        1  Switch Count  11
2518        Page Faults                       0
2519        Page Reclaims                     33749
2520        Page Swaps                        0
2521        Voluntary Context Switches        32
2522        Involuntary Context Switches      0
2523        Block Input Operations            0
2524        Block Output Operations           88
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
      8 variables.
2529 NOTE: The data set WORK._CLASSOUT has 61 observations and 9
      variables.
2530 NOTE: PROCEDURE DMDB used (Total process time):
2531        real time           0.11 seconds
2532        user cpu time       0.05 seconds
2533        system cpu time     0.07 seconds
2534        memory              159533.12k
2535        OS Memory           169592.00k
2536        Timestamp           07/01/2024 05:54:44 AM
2537        Step Count                        1  Switch Count  6
2538        Page Faults                       0
2539        Page Reclaims                     33928
2540        Page Swaps                        0
2541        Voluntary Context Switches        37
2542        Involuntary Context Switches      0
2543        Block Input Operations            0
2544        Block Output Operations           2392
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              159533.12k
2553       OS Memory           169592.00k
2554       Timestamp           07/01/2024 05:54:44 AM
2555       Step Count                        1  Switch Count  0
2556       Page Faults                       0
2557       Page Reclaims                     63
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 8 variable(s)
     .
2567 NOTE: The PROC statement has finished with return code 0.
2568 NOTE: This is the INPUT statement.
2569 NOTE: 8 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: 8 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=29
2579 NOTE: The total number of variables is 8 with dimensionalit
     y 61.
```

```
2580 NOTE: The data set EMWS4.VARCLUS2_OUTDUMMY has 3 observatio
     ns and 62 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           0.10 seconds
2584       user cpu time       0.04 seconds
2585       system cpu time     0.06 seconds
2586       memory              160464.71k
2587       OS Memory           170880.00k
2588       Timestamp           07/01/2024 05:54:44 AM
2589       Step Count                        1  Switch Count  13
2590       Page Faults                       0
2591       Page Reclaims                     33503
2592       Page Swaps                        0
2593       Voluntary Context Switches        30
2594       Involuntary Context Switches      0
2595       Block Input Operations            0
2596       Block Output Operations           31496
2597
2598 NOTE: This is the SCORE statement.
2599 NOTE: The data set WORK._NEWTRAINDS has 24999 observations
     and 76 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/VarClus2/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           0.13 seconds
2610       user cpu time       0.04 seconds
```

```
2611       system cpu time     0.07 seconds
2612       memory              160464.71k
2613       OS Memory           170880.00k
2614       Timestamp           07/01/2024 05:54:44 AM
2615       Step Count                        1  Switch Count  8
2616       Page Faults                       0
2617       Page Reclaims                     34097
2618       Page Swaps                        0
2619       Voluntary Context Switches        70
2620       Involuntary Context Switches      0
2621       Block Input Operations            0
2622       Block Output Operations           31800
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              160464.71k
2634       OS Memory           170880.00k
2635       Timestamp           07/01/2024 05:54:44 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     0.01 seconds
2654       memory              160464.71k
2655       OS Memory           170880.00k
2656       Timestamp           07/01/2024 05:54:44 AM
2657       Step Count                        1  Switch Count  0
2658       Page Faults                       0
2659       Page Reclaims                     284
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              160464.71k
2675       OS Memory           170880.00k
2676       Timestamp           07/01/2024 05:54:44 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 18170  %let SYSCC = 0;
2689 NOTE: PROCEDURE DISPLAY used (Total process time):
2690        real time           0.00 seconds
2691        user cpu time       0.00 seconds
2692        system cpu time     0.00 seconds
2693        memory              160464.71k
2694        OS Memory           170880.00k
2695        Timestamp           07/01/2024 05:54:44 AM
2696        Step Count                        1  Switch Count  1
2697        Page Faults                       0
2698        Page Reclaims                     59
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.VARCLUS2_VARCLUSMETA has 1 observa
     tions 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              160464.71k
2713        OS Memory           170880.00k
```

```
2714        Timestamp           07/01/2024 05:54:45 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              160464.71k
2731        OS Memory           170880.00k
2732        Timestamp           07/01/2024 05:54:45 AM
2733        Step Count                        1  Switch Count  0
2734        Page Faults                       0
2735        Page Reclaims                     32
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: Clustering algorithm converged.
2747  NOTE: Clustering algorithm converged.
2748  NOTE: Clustering algorithm converged.
2749  NOTE: Clustering algorithm converged.
```

```
2750 NOTE: Clustering algorithm converged.
2751 NOTE: Clustering algorithm converged.
2752 NOTE: Clustering algorithm converged.
2753 NOTE: Clustering algorithm converged.
2754 NOTE: Clustering algorithm converged.
2755 NOTE: Clustering algorithm converged.
2756 NOTE: Clustering algorithm converged.
2757 NOTE: Clustering algorithm converged.
2758 NOTE: Clustering algorithm converged.
2759 NOTE: Clustering algorithm converged.
2760 NOTE: Clustering algorithm converged.
2761 NOTE: Clustering algorithm converged.
2762 NOTE: Clustering algorithm converged.
2763 NOTE: Clustering algorithm converged.
2764 NOTE: Clustering algorithm converged.
2765 NOTE: Clustering algorithm converged.
2766 NOTE: Clustering algorithm converged.
2767 NOTE: Clustering algorithm converged.
2768 NOTE: Clustering algorithm converged.
2769 NOTE: Clustering algorithm converged.
2770 NOTE: Clustering algorithm converged.
2771 NOTE: Clustering algorithm converged.
2772 NOTE: The data set EMWS4.VARCLUS2_OUTTREE has 112 observati
     ons and 3 variables.
2773 NOTE: The data set EMWS4.VARCLUS2_OUTSTAT has 1422 observat
     ions and 64 variables.
2774 NOTE: The PROCEDURE VARCLUS printed pages 2-5.
2775 NOTE: PROCEDURE VARCLUS used (Total process time):
2776       real time           2.19 seconds
2777       user cpu time       0.10 seconds
2778       system cpu time     0.03 seconds
2779       memory              160464.71k
2780       OS Memory           170880.00k
2781       Timestamp           07/01/2024 05:54:47 AM
2782       Step Count                        1  Switch Count  0
2783       Page Faults                       0
```

```
2784       Page Reclaims                     4310
2785       Page Swaps                        0
2786       Voluntary Context Switches        1554
2787       Involuntary Context Switches      1
2788       Block Input Operations            288
2789       Block Output Operations           2824
2790
2791
2792
2793  NOTE: Deleting WORK._NEWTRAINDS (memtype=DATA).
2794
2795  NOTE: PROCEDURE DATASETS used (Total process time):
2796       real time           0.00 seconds
2797       user cpu time       0.00 seconds
2798       system cpu time     0.01 seconds
2799       memory              160464.71k
2800       OS Memory           170880.00k
2801       Timestamp           07/01/2024 05:54:47 AM
2802       Step Count                        1  Switch Count  0
2803       Page Faults                       0
2804       Page Reclaims                     49
2805       Page Swaps                        0
2806       Voluntary Context Switches        0
2807       Involuntary Context Switches      0
2808       Block Input Operations            0
2809       Block Output Operations           8
2810
2811
2812
2813  18171  %let SYSCC = 0;
2814  NOTE: PROCEDURE DISPLAY used (Total process time):
2815       real time           0.00 seconds
2816       user cpu time       0.00 seconds
2817       system cpu time     0.00 seconds
2818       memory              160464.71k
2819       OS Memory           170880.00k
```

```
2820       Timestamp           07/01/2024 05:54:47 AM
2821       Step Count                        1  Switch Count  1
2822       Page Faults                       0
2823       Page Reclaims                     62
2824       Page Swaps                        0
2825       Voluntary Context Switches        1
2826       Involuntary Context Switches      0
2827       Block Input Operations            0
2828       Block Output Operations           0
2829
2830
2831
2832 NOTE: Numeric values have been converted to character value
      s at the places given by: (Line):(Column).
2833       5:214
2834 NOTE: There were 1422 observations read from the data set E
      MWS4.VARCLUS2_OUTSTAT.
2835 NOTE: DATA statement used (Total process time):
2836       real time           0.00 seconds
2837       user cpu time       0.00 seconds
2838       system cpu time     0.00 seconds
2839       memory              160464.71k
2840       OS Memory           170880.00k
2841       Timestamp           07/01/2024 05:54:47 AM
2842       Step Count                        1  Switch Count  0
2843       Page Faults                       0
2844       Page Reclaims                     198
2845       Page Swaps                        0
2846       Voluntary Context Switches        15
2847       Involuntary Context Switches      0
2848       Block Input Operations            1568
2849       Block Output Operations           0
2850
2851
2852
2853 NOTE: Numeric values have been converted to character value
```

```
     s at the places given by: (Line):(Column).
2854       7:15
2855 NOTE: Character values have been converted to numeric value
     s at the places given by: (Line):(Column).
2856       7:9
2857 NOTE: There were 1422 observations read from the data set E
     MWS4.VARCLUS2_OUTSTAT.
2858 NOTE: The data set EMWS4.VARCLUS2_OUTCLUSCORR has 28 observ
     ations and 62 variables.
2859 NOTE: DATA statement used (Total process time):
2860       real time           0.00 seconds
2861       user cpu time       0.00 seconds
2862       system cpu time     0.00 seconds
2863       memory              160464.71k
2864       OS Memory           170880.00k
2865       Timestamp           07/01/2024 05:54:47 AM
2866       Step Count                        1  Switch Count  0
2867       Page Faults                       0
2868       Page Reclaims                     392
2869       Page Swaps                        0
2870       Voluntary Context Switches        17
2871       Involuntary Context Switches      0
2872       Block Input Operations            0
2873       Block Output Operations           264
2874
2875
2876
2877 NOTE: There were 28 observations read from the data set EMW
     S4.VARCLUS2_OUTCLUSCORR.
2878 NOTE: The data set WORK.CORR_TMP has 28 observations and 62
      variables.
2879 NOTE: DATA statement used (Total process time):
2880       real time           0.00 seconds
2881       user cpu time       0.00 seconds
2882       system cpu time     0.00 seconds
2883       memory              160464.71k
```

```
2884        OS Memory           170880.00k
2885        Timestamp           07/01/2024 05:54:47 AM
2886        Step Count                      1  Switch Count  0
2887        Page Faults                     0
2888        Page Reclaims                   126
2889        Page Swaps                      0
2890        Voluntary Context Switches      9
2891        Involuntary Context Switches    0
2892        Block Input Operations          288
2893        Block Output Operations         264
2894
2895
2896
2897 NOTE: There were 28 observations read from the data set EMW
     S4.VARCLUS2_OUTCLUSCORR.
2898 NOTE: The data set EMWS4.VARCLUS2_OUTCLUSCORR has 28 observ
     ations and 29 variables.
2899 NOTE: DATA statement used (Total process time):
2900        real time           0.03 seconds
2901        user cpu time       0.03 seconds
2902        system cpu time     0.01 seconds
2903        memory              160464.71k
2904        OS Memory           170880.00k
2905        Timestamp           07/01/2024 05:54:47 AM
2906        Step Count                      1  Switch Count  0
2907        Page Faults                     0
2908        Page Reclaims                   3340
2909        Page Swaps                      0
2910        Voluntary Context Switches      30
2911        Involuntary Context Switches    0
2912        Block Input Operations          0
2913        Block Output Operations         264
2914
2915
2916
2917 NOTE: There were 28 observations read from the data set EMW
```

```
      S4.VARCLUS2_OUTCLUSCORR.
2918  NOTE: The data set WORK.CORR_TMP has 28 observations and 29
       variables.
2919  NOTE: DATA statement used (Total process time):
2920        real time           0.00 seconds
2921        user cpu time       0.00 seconds
2922        system cpu time     0.00 seconds
2923        memory              160464.71k
2924        OS Memory           170880.00k
2925        Timestamp           07/01/2024 05:54:47 AM
2926        Step Count                        1  Switch Count  0
2927        Page Faults                       0
2928        Page Reclaims                     126
2929        Page Swaps                        0
2930        Voluntary Context Switches        9
2931        Involuntary Context Switches      0
2932        Block Input Operations            288
2933        Block Output Operations           264
2934
2935
2936
2937  NOTE: There were 28 observations read from the data set EMW
      S4.VARCLUS2_OUTCLUSCORR.
2938  NOTE: The data set EMWS4.VARCLUS2_OUTCLUSCORR has 28 observ
      ations and 29 variables.
2939  NOTE: DATA statement used (Total process time):
2940        real time           0.01 seconds
2941        user cpu time       0.00 seconds
2942        system cpu time     0.00 seconds
2943        memory              160464.71k
2944        OS Memory           170880.00k
2945        Timestamp           07/01/2024 05:54:47 AM
2946        Step Count                        1  Switch Count  0
2947        Page Faults                       0
2948        Page Reclaims                     168
2949        Page Swaps                        0
```

```
2950       Voluntary Context Switches        30
2951       Involuntary Context Switches      0
2952       Block Input Operations            0
2953       Block Output Operations           264
2954
2955
2956
2957 NOTE: Deleting WORK.CORR_TMP (memtype=DATA).
2958
2959 NOTE: PROCEDURE DATASETS used (Total process time):
2960       real time           0.00 seconds
2961       user cpu time       0.00 seconds
2962       system cpu time     0.00 seconds
2963       memory              160464.71k
2964       OS Memory           170880.00k
2965       Timestamp           07/01/2024 05:54:47 AM
2966       Step Count                        1  Switch Count  0
2967       Page Faults                       0
2968       Page Reclaims                     49
2969       Page Swaps                        0
2970       Voluntary Context Switches        0
2971       Involuntary Context Switches      0
2972       Block Input Operations            0
2973       Block Output Operations           8
2974
2975
2976
2977 NOTE: There were 28 observations read from the data set EMW
     S4.VARCLUS2_OUTCLUSCORR.
2978 NOTE: The data set WORK._TMPCLUSRSQ has 28 observations and
      29 variables.
2979 NOTE: PROCEDURE SORT used (Total process time):
2980       real time           0.00 seconds
2981       user cpu time       0.00 seconds
2982       system cpu time     0.00 seconds
2983       memory              160464.71k
```

```
2984        OS Memory           170880.00k
2985        Timestamp           07/01/2024 05:54:47 AM
2986        Step Count                      1  Switch Count  0
2987        Page Faults                     0
2988        Page Reclaims                   169
2989        Page Swaps                      0
2990        Voluntary Context Switches      11
2991        Involuntary Context Switches    0
2992        Block Input Operations          288
2993        Block Output Operations         272
2994
2995
2996
2997 NOTE: There were 28 observations read from the data set WOR
     K._TMPCLUSRSQ.
2998 NOTE: The data set WORK._TMPCLUSRSQ has 784 observations an
     d 4 variables.
2999 NOTE: PROCEDURE TRANSPOSE used (Total process time):
3000        real time           0.00 seconds
3001        user cpu time       0.00 seconds
3002        system cpu time     0.00 seconds
3003        memory              160464.71k
3004        OS Memory           170880.00k
3005        Timestamp           07/01/2024 05:54:47 AM
3006        Step Count                      1  Switch Count  0
3007        Page Faults                     0
3008        Page Reclaims                   243
3009        Page Swaps                      0
3010        Voluntary Context Switches      0
3011        Involuntary Context Switches    0
3012        Block Input Operations          0
3013        Block Output Operations         528
3014
3015
3016
3017 NOTE: There were 784 observations read from the data set WO
```

```
RK._TMPCLUSRSQ.
3018 NOTE: The data set WORK._TMPCLUSRSQ has 784 observations an
     d 4 variables.
3019 NOTE: PROCEDURE SORT used (Total process time):
3020       real time           0.00 seconds
3021       user cpu time       0.01 seconds
3022       system cpu time     0.00 seconds
3023       memory              160464.71k
3024       OS Memory           170880.00k
3025       Timestamp           07/01/2024 05:54:47 AM
3026       Step Count                        1  Switch Count  0
3027       Page Faults                       0
3028       Page Reclaims                     106
3029       Page Swaps                        0
3030       Voluntary Context Switches        0
3031       Involuntary Context Switches      0
3032       Block Input Operations            0
3033       Block Output Operations           264
3034
3035
3036
3037 NOTE: There were 784 observations read from the data set WO
     RK._TMPCLUSRSQ.
3038 NOTE: The data set WORK._TMPCLUSRSQ has 756 observations an
     d 4 variables.
3039 NOTE: DATA statement used (Total process time):
3040       real time           0.00 seconds
3041       user cpu time       0.00 seconds
3042       system cpu time     0.00 seconds
3043       memory              160464.71k
3044       OS Memory           170880.00k
3045       Timestamp           07/01/2024 05:54:47 AM
3046       Step Count                        1  Switch Count  0
3047       Page Faults                       0
3048       Page Reclaims                     116
3049       Page Swaps                        0
```

```
3050       Voluntary Context Switches        0
3051       Involuntary Context Switches      0
3052       Block Input Operations            0
3053       Block Output Operations           264
3054
3055
3056
3057 NOTE: There were 756 observations read from the data set WO
     RK._TMPCLUSRSQ.
3058 NOTE: The data set WORK._NEXTCLUSRSQ has 28 observations an
     d 4 variables.
3059 NOTE: DATA statement used (Total process time):
3060       real time           0.00 seconds
3061       user cpu time       0.00 seconds
3062       system cpu time     0.01 seconds
3063       memory              160464.71k
3064       OS Memory           170880.00k
3065       Timestamp           07/01/2024 05:54:47 AM
3066       Step Count                        1  Switch Count  0
3067       Page Faults                       0
3068       Page Reclaims                     474
3069       Page Swaps                        0
3070       Voluntary Context Switches        0
3071       Involuntary Context Switches      0
3072       Block Input Operations            0
3073       Block Output Operations           264
3074
3075
3076
3077 NOTE: Deleting WORK._TMPCLUSRSQ (memtype=DATA).
3078
3079 NOTE: PROCEDURE DATASETS used (Total process time):
3080       real time           0.00 seconds
3081       user cpu time       0.00 seconds
3082       system cpu time     0.00 seconds
3083       memory              160464.71k
```

```
3084        OS Memory           170880.00k
3085        Timestamp           07/01/2024 05:54:47 AM
3086        Step Count                       1  Switch Count  0
3087        Page Faults                      0
3088        Page Reclaims                    49
3089        Page Swaps                       0
3090        Voluntary Context Switches       0
3091        Involuntary Context Switches     0
3092        Block Input Operations           0
3093        Block Output Operations          8
3094
3095
3096
3097 NOTE: Numeric values have been converted to character value
     s at the places given by: (Line):(Column).
3098        2:77
3099 NOTE: Character values have been converted to numeric value
     s at the places given by: (Line):(Column).
3100        2:71
3101 NOTE: There were 1422 observations read from the data set E
     MWS4.VARCLUS2_OUTSTAT.
3102 NOTE: The data set WORK._TMPDS has 2 observations and 63 va
     riables.
3103 NOTE: DATA statement used (Total process time):
3104        real time           0.00 seconds
3105        user cpu time       0.00 seconds
3106        system cpu time     0.00 seconds
3107        memory              160464.71k
3108        OS Memory           170880.00k
3109        Timestamp           07/01/2024 05:54:47 AM
3110        Step Count                       1  Switch Count  0
3111        Page Faults                      0
3112        Page Reclaims                    396
3113        Page Swaps                       0
3114        Voluntary Context Switches       4
3115        Involuntary Context Switches     0
```

```
3116       Block Input Operations            0
3117       Block Output Operations           264
3118
3119
3120
3121 NOTE: There were 2 observations read from the data set WORK
     ._TMPDS.
3122 NOTE: The data set WORK._OWNRSQ has 61 observations and 4 v
     ariables.
3123 NOTE: PROCEDURE TRANSPOSE used (Total process time):
3124       real time           0.00 seconds
3125       user cpu time       0.00 seconds
3126       system cpu time     0.00 seconds
3127       memory              160464.71k
3128       OS Memory           170880.00k
3129       Timestamp           07/01/2024 05:54:47 AM
3130       Step Count                        1  Switch Count  0
3131       Page Faults                       0
3132       Page Reclaims                     227
3133       Page Swaps                        0
3134       Voluntary Context Switches        0
3135       Involuntary Context Switches      0
3136       Block Input Operations            0
3137       Block Output Operations           528
3138
3139
3140
3141 NOTE: Numeric values have been converted to character value
     s at the places given by: (Line):(Column).
3142       12:35
3143 NOTE: Variable GCluster is uninitialized.
3144 NOTE: There were 61 observations read from the data set WOR
     K._OWNRSQ.
3145 NOTE: The data set WORK._OWNRSQ has 61 observations and 4 v
     ariables.
3146 NOTE: DATA statement used (Total process time):
```

```
3147        real time           0.00 seconds
3148        user cpu time       0.01 seconds
3149        system cpu time     0.00 seconds
3150        memory              160464.71k
3151        OS Memory           170880.00k
3152        Timestamp           07/01/2024 05:54:47 AM
3153        Step Count                        1  Switch Count  0
3154        Page Faults                       0
3155        Page Reclaims                     583
3156        Page Swaps                        0
3157        Voluntary Context Switches        0
3158        Involuntary Context Switches      0
3159        Block Input Operations            0
3160        Block Output Operations           264
3161
3162
3163
3164 NOTE: There were 61 observations read from the data set WOR
     K._OWNRSQ.
3165 NOTE: The data set WORK._OWNRSQ has 61 observations and 4 v
     ariables.
3166 NOTE: PROCEDURE SORT used (Total process time):
3167        real time           0.00 seconds
3168        user cpu time       0.00 seconds
3169        system cpu time     0.00 seconds
3170        memory              160464.71k
3171        OS Memory           170880.00k
3172        Timestamp           07/01/2024 05:54:47 AM
3173        Step Count                        1  Switch Count  0
3174        Page Faults                       0
3175        Page Reclaims                     115
3176        Page Swaps                        0
3177        Voluntary Context Switches        0
3178        Involuntary Context Switches      0
3179        Block Input Operations            0
3180        Block Output Operations           264
```

```
3181
3182
3183
3184 NOTE: Deleting WORK._TMPDS (memtype=DATA).
3185
3186 NOTE: PROCEDURE DATASETS used (Total process time):
3187       real time           0.00 seconds
3188       user cpu time       0.00 seconds
3189       system cpu time     0.00 seconds
3190       memory              160464.71k
3191       OS Memory           170880.00k
3192       Timestamp           07/01/2024 05:54:47 AM
3193       Step Count                        1  Switch Count  0
3194       Page Faults                       0
3195       Page Reclaims                     51
3196       Page Swaps                        0
3197       Voluntary Context Switches        0
3198       Involuntary Context Switches      0
3199       Block Input Operations            0
3200       Block Output Operations           8
3201
3202
3203
3204 NOTE: Numeric values have been converted to character value
     s at the places given by: (Line):(Column).
3205       4:40
3206 NOTE: Character values have been converted to numeric value
     s at the places given by: (Line):(Column).
3207       4:34
3208 NOTE: Variable GCluster is uninitialized.
3209 NOTE: There were 1422 observations read from the data set E
     MWS4.VARCLUS2_OUTSTAT.
3210 NOTE: The data set EMWS4.VARCLUS2_OUTCLUSSTRUCT has 28 obse
     rvations and 62 variables.
3211 NOTE: DATA statement used (Total process time):
3212       real time           0.01 seconds
```

```
3213       user cpu time       0.00 seconds
3214       system cpu time     0.00 seconds
3215       memory              160464.71k
3216       OS Memory           170880.00k
3217       Timestamp           07/01/2024 05:54:47 AM
3218       Step Count                        1  Switch Count  0
3219       Page Faults                       0
3220       Page Reclaims                     477
3221       Page Swaps                        0
3222       Voluntary Context Switches        16
3223       Involuntary Context Switches      0
3224       Block Input Operations            0
3225       Block Output Operations           264
3226
3227
3228
3229 NOTE: There were 28 observations read from the data set EMW
     S4.VARCLUS2_OUTCLUSSTRUCT.
3230 NOTE: The data set WORK.CORR_TMP has 28 observations and 62
      variables.
3231 NOTE: DATA statement used (Total process time):
3232       real time           0.00 seconds
3233       user cpu time       0.00 seconds
3234       system cpu time     0.00 seconds
3235       memory              160464.71k
3236       OS Memory           170880.00k
3237       Timestamp           07/01/2024 05:54:47 AM
3238       Step Count                        1  Switch Count  0
3239       Page Faults                       0
3240       Page Reclaims                     123
3241       Page Swaps                        0
3242       Voluntary Context Switches        9
3243       Involuntary Context Switches      0
3244       Block Input Operations            288
3245       Block Output Operations           264
3246
```

```
3247
3248
3249 NOTE: There were 28 observations read from the data set EMW
     S4.VARCLUS2_OUTCLUSSTRUCT.
3250 NOTE: The data set EMWS4.VARCLUS2_OUTCLUSSTRUCT has 28 obse
     rvations and 62 variables.
3251 NOTE: DATA statement used (Total process time):
3252       real time           0.01 seconds
3253       user cpu time       0.01 seconds
3254       system cpu time     0.01 seconds
3255       memory              160464.71k
3256       OS Memory           170880.00k
3257       Timestamp           07/01/2024 05:54:47 AM
3258       Step Count                        1  Switch Count  0
3259       Page Faults                       0
3260       Page Reclaims                     157
3261       Page Swaps                        0
3262       Voluntary Context Switches        32
3263       Involuntary Context Switches      0
3264       Block Input Operations            0
3265       Block Output Operations           264
3266
3267
3268
3269 NOTE: Deleting WORK.CORR_TMP (memtype=DATA).
3270
3271 NOTE: PROCEDURE DATASETS used (Total process time):
3272       real time           0.00 seconds
3273       user cpu time       0.00 seconds
3274       system cpu time     0.00 seconds
3275       memory              160464.71k
3276       OS Memory           170880.00k
3277       Timestamp           07/01/2024 05:54:47 AM
3278       Step Count                        1  Switch Count  0
3279       Page Faults                       0
3280       Page Reclaims                     49
```

```
3281        Page Swaps                          0
3282        Voluntary Context Switches          0
3283        Involuntary Context Switches        0
3284        Block Input Operations              0
3285        Block Output Operations             8
3286
3287
3288
3289 NOTE: There were 28 observations read from the data set EMW
     S4.VARCLUS2_OUTCLUSSTRUCT.
3290 NOTE: The data set WORK._TMPCLUSRSQ has 28 observations and
      62 variables.
3291 NOTE: PROCEDURE SORT used (Total process time):
3292        real time           0.00 seconds
3293        user cpu time       0.00 seconds
3294        system cpu time     0.00 seconds
3295        memory              160464.71k
3296        OS Memory           170880.00k
3297        Timestamp           07/01/2024 05:54:47 AM
3298        Step Count                        1  Switch Count  0
3299        Page Faults                       0
3300        Page Reclaims                     143
3301        Page Swaps                        0
3302        Voluntary Context Switches        10
3303        Involuntary Context Switches      0
3304        Block Input Operations            288
3305        Block Output Operations           272
3306
3307
3308
3309 NOTE: There were 28 observations read from the data set WOR
     K._TMPCLUSRSQ.
3310 NOTE: The data set WORK._TMPCLUSRSQ has 1708 observations a
     nd 4 variables.
3311 NOTE: PROCEDURE TRANSPOSE used (Total process time):
3312        real time           0.00 seconds
```

```
3313       user cpu time       0.00 seconds
3314       system cpu time     0.00 seconds
3315       memory              160464.71k
3316       OS Memory           170880.00k
3317       Timestamp           07/01/2024 05:54:47 AM
3318       Step Count                        1  Switch Count  0
3319       Page Faults                       0
3320       Page Reclaims                     201
3321       Page Swaps                        0
3322       Voluntary Context Switches        0
3323       Involuntary Context Switches      0
3324       Block Input Operations            0
3325       Block Output Operations           784
3326
3327
3328
3329 NOTE: There were 1708 observations read from the data set W
     ORK._TMPCLUSRSQ.
3330 NOTE: The data set WORK._TMPCLUSRSQ has 1708 observations a
     nd 4 variables.
3331 NOTE: PROCEDURE SORT used (Total process time):
3332       real time           0.00 seconds
3333       user cpu time       0.01 seconds
3334       system cpu time     0.00 seconds
3335       memory              160464.71k
3336       OS Memory           170880.00k
3337       Timestamp           07/01/2024 05:54:47 AM
3338       Step Count                        1  Switch Count  0
3339       Page Faults                       0
3340       Page Reclaims                     135
3341       Page Swaps                        0
3342       Voluntary Context Switches        0
3343       Involuntary Context Switches      0
3344       Block Input Operations            0
3345       Block Output Operations           520
3346
```

```
3347
3348
3349 NOTE: There were 1708 observations read from the data set W
     ORK._TMPCLUSRSQ.
3350 NOTE: The data set WORK._TMPCLUSRSQ has 1647 observations a
     nd 4 variables.
3351 NOTE: DATA statement used (Total process time):
3352       real time           0.00 seconds
3353       user cpu time       0.00 seconds
3354       system cpu time     0.00 seconds
3355       memory              160464.71k
3356       OS Memory           170880.00k
3357       Timestamp           07/01/2024 05:54:47 AM
3358       Step Count                        1  Switch Count  0
3359       Page Faults                       0
3360       Page Reclaims                     101
3361       Page Swaps                        0
3362       Voluntary Context Switches        0
3363       Involuntary Context Switches      0
3364       Block Input Operations            0
3365       Block Output Operations           528
3366
3367
3368
3369 NOTE: There were 1647 observations read from the data set W
     ORK._TMPCLUSRSQ.
3370 NOTE: The data set WORK._TMPCLUSRSQ has 1647 observations a
     nd 4 variables.
3371 NOTE: PROCEDURE SORT used (Total process time):
3372       real time           0.00 seconds
3373       user cpu time       0.00 seconds
3374       system cpu time     0.00 seconds
3375       memory              160464.71k
3376       OS Memory           170880.00k
3377       Timestamp           07/01/2024 05:54:47 AM
3378       Step Count                        1  Switch Count  0
```

```
3379       Page Faults                          0
3380       Page Reclaims                        126
3381       Page Swaps                           0
3382       Voluntary Context Switches           0
3383       Involuntary Context Switches         0
3384       Block Input Operations               0
3385       Block Output Operations              520
3386
3387
3388
3389 NOTE: There were 1647 observations read from the data set W
     ORK._TMPCLUSRSQ.
3390 NOTE: The data set WORK._NEXTVARRSQ has 61 observations and
      4 variables.
3391 NOTE: DATA statement used (Total process time):
3392       real time            0.00 seconds
3393       user cpu time        0.01 seconds
3394       system cpu time      0.00 seconds
3395       memory               160464.71k
3396       OS Memory            170880.00k
3397       Timestamp            07/01/2024 05:54:47 AM
3398       Step Count                        1  Switch Count  0
3399       Page Faults                       0
3400       Page Reclaims                     471
3401       Page Swaps                        0
3402       Voluntary Context Switches        0
3403       Involuntary Context Switches      0
3404       Block Input Operations            0
3405       Block Output Operations           264
3406
3407
3408
3409 NOTE: Deleting WORK._TMPCLUSRSQ (memtype=DATA).
3410
3411 NOTE: PROCEDURE DATASETS used (Total process time):
3412       real time            0.00 seconds
```

```
3413       user cpu time       0.00 seconds
3414       system cpu time     0.00 seconds
3415       memory              160464.71k
3416       OS Memory           170880.00k
3417       Timestamp           07/01/2024 05:54:47 AM
3418       Step Count                        1  Switch Count  0
3419       Page Faults                       0
3420       Page Reclaims                     49
3421       Page Swaps                        0
3422       Voluntary Context Switches        0
3423       Involuntary Context Switches      0
3424       Block Input Operations            0
3425       Block Output Operations           8
3426
3427
3428
3429 NOTE: There were 61 observations read from the data set WOR
     K._OWNRSQ.
3430 NOTE: The data set WORK._OWNRSQ has 61 observations and 4 v
     ariables.
3431 NOTE: PROCEDURE SORT used (Total process time):
3432       real time           0.00 seconds
3433       user cpu time       0.00 seconds
3434       system cpu time     0.00 seconds
3435       memory              160464.71k
3436       OS Memory           170880.00k
3437       Timestamp           07/01/2024 05:54:47 AM
3438       Step Count                        1  Switch Count  0
3439       Page Faults                       0
3440       Page Reclaims                     115
3441       Page Swaps                        0
3442       Voluntary Context Switches        0
3443       Involuntary Context Switches      0
3444       Block Input Operations            0
3445       Block Output Operations           264
3446
```

```
3447
3448
3449 NOTE: There were 61 observations read from the data set WOR
     K._NEXTVARRSQ.
3450 NOTE: The data set WORK._NEXTVARRSQ has 61 observations and
      4 variables.
3451 NOTE: PROCEDURE SORT used (Total process time):
3452       real time           0.00 seconds
3453       user cpu time       0.00 seconds
3454       system cpu time     0.00 seconds
3455       memory              160464.71k
3456       OS Memory           170880.00k
3457       Timestamp           07/01/2024 05:54:47 AM
3458       Step Count                        1  Switch Count  0
3459       Page Faults                       0
3460       Page Reclaims                     116
3461       Page Swaps                        0
3462       Voluntary Context Switches        0
3463       Involuntary Context Switches      0
3464       Block Input Operations            0
3465       Block Output Operations           264
3466
3467
3468
3469 NOTE: There were 61 observations read from the data set WOR
     K._OWNRSQ.
3470 NOTE: There were 61 observations read from the data set WOR
     K._NEXTVARRSQ.
3471 NOTE: The data set EMWS4.VARCLUS2_OUTRSQUARE has 61 observa
     tions and 7 variables.
3472 NOTE: DATA statement used (Total process time):
3473       real time           0.00 seconds
3474       user cpu time       0.00 seconds
3475       system cpu time     0.00 seconds
3476       memory              160464.71k
3477       OS Memory           170880.00k
```

```
3478        Timestamp           07/01/2024 05:54:47 AM
3479        Step Count                        1  Switch Count  0
3480        Page Faults                       0
3481        Page Reclaims                     286
3482        Page Swaps                        0
3483        Voluntary Context Switches        16
3484        Involuntary Context Switches      0
3485        Block Input Operations            0
3486        Block Output Operations           264
3487
3488
3489
3490 NOTE: There were 28 observations read from the data set WOR
     K._NEXTCLUSRSQ.
3491 NOTE: The data set WORK._NEXTCLUSRSQ has 28 observations an
     d 7 variables.
3492 NOTE: DATA statement used (Total process time):
3493        real time           0.00 seconds
3494        user cpu time       0.00 seconds
3495        system cpu time     0.00 seconds
3496        memory              160464.71k
3497        OS Memory           170880.00k
3498        Timestamp           07/01/2024 05:54:47 AM
3499        Step Count                        1  Switch Count  0
3500        Page Faults                       0
3501        Page Reclaims                     358
3502        Page Swaps                        0
3503        Voluntary Context Switches        0
3504        Involuntary Context Switches      0
3505        Block Input Operations            0
3506        Block Output Operations           264
3507
3508
3509
3510 NOTE: There were 61 observations read from the data set EMW
     S4.VARCLUS2_OUTRSQUARE.
```

```
3511 NOTE: The data set EMWS4.VARCLUS2_OUTRSQUARE has 61 observa
     tions and 7 variables.
3512 NOTE: PROCEDURE SORT used (Total process time):
3513       real time           0.01 seconds
3514       user cpu time       0.01 seconds
3515       system cpu time     0.00 seconds
3516       memory              160464.71k
3517       OS Memory           170880.00k
3518       Timestamp           07/01/2024 05:54:47 AM
3519       Step Count                        1  Switch Count  0
3520       Page Faults                       0
3521       Page Reclaims                     119
3522       Page Swaps                        0
3523       Voluntary Context Switches        37
3524       Involuntary Context Switches      0
3525       Block Input Operations            288
3526       Block Output Operations           264
3527
3528
3529
3530 NOTE: There were 28 observations read from the data set WOR
     K._NEXTCLUSRSQ.
3531 NOTE: The data set WORK._NEXTCLUSRSQ has 28 observations an
     d 7 variables.
3532 NOTE: PROCEDURE SORT used (Total process time):
3533       real time           0.00 seconds
3534       user cpu time       0.00 seconds
3535       system cpu time     0.00 seconds
3536       memory              160464.71k
3537       OS Memory           170880.00k
3538       Timestamp           07/01/2024 05:54:47 AM
3539       Step Count                        1  Switch Count  0
3540       Page Faults                       0
3541       Page Reclaims                     117
3542       Page Swaps                        0
3543       Voluntary Context Switches        0
```

```
3544        Involuntary Context Switches      0
3545        Block Input Operations            0
3546        Block Output Operations           264
3547
3548
3549
3550  NOTE: There were 61 observations read from the data set EMW
      S4.VARCLUS2_OUTRSQUARE.
3551  NOTE: There were 28 observations read from the data set WOR
      K._NEXTCLUSRSQ.
3552  NOTE: The data set EMWS4.VARCLUS2_OUTRSQUARE has 89 observa
      tions and 7 variables.
3553  NOTE: DATA statement used (Total process time):
3554        real time           0.01 seconds
3555        user cpu time       0.00 seconds
3556        system cpu time     0.00 seconds
3557        memory              160464.71k
3558        OS Memory           170880.00k
3559        Timestamp           07/01/2024 05:54:47 AM
3560        Step Count                        1  Switch Count  0
3561        Page Faults                       0
3562        Page Reclaims                     165
3563        Page Swaps                        0
3564        Voluntary Context Switches        38
3565        Involuntary Context Switches      2
3566        Block Input Operations            288
3567        Block Output Operations           264
3568
3569
3570
3571  NOTE: There were 89 observations read from the data set EMW
      S4.VARCLUS2_OUTRSQUARE.
3572  NOTE: The data set EMWS4.VARCLUS2_OUTRSQUARE has 89 observa
      tions and 9 variables.
3573  NOTE: DATA statement used (Total process time):
3574        real time           0.01 seconds
```

```
3575       user cpu time       0.00 seconds
3576       system cpu time     0.01 seconds
3577       memory              160464.71k
3578       OS Memory           170880.00k
3579       Timestamp           07/01/2024 05:54:47 AM
3580       Step Count                        1  Switch Count  0
3581       Page Faults                       0
3582       Page Reclaims                     469
3583       Page Swaps                        0
3584       Voluntary Context Switches        40
3585       Involuntary Context Switches      0
3586       Block Input Operations            288
3587       Block Output Operations           264
3588
3589
3590
3591  NOTE: Numeric values have been converted to character value
      s at the places given by: (Line):(Column).
3592       50511:90
3593  NOTE: Character values have been converted to numeric value
      s at the places given by: (Line):(Column).
3594       50511:84
3595  NOTE: There were 1422 observations read from the data set E
      MWS4.VARCLUS2_OUTSTAT.
3596  NOTE: The data set EMWS4.VARCLUS2_OUTCLUSCORRPLOT has 28 ob
      servations and 62 variables.
3597  NOTE: DATA statement used (Total process time):
3598       real time           0.00 seconds
3599       user cpu time       0.01 seconds
3600       system cpu time     0.00 seconds
3601       memory              160464.71k
3602       OS Memory           170880.00k
3603       Timestamp           07/01/2024 05:54:47 AM
3604       Step Count                        1  Switch Count  0
3605       Page Faults                       0
3606       Page Reclaims                     366
```

```
3607        Page Swaps                          0
3608        Voluntary Context Switches          18
3609        Involuntary Context Switches        0
3610        Block Input Operations              0
3611        Block Output Operations             264
3612
3613
3614
3615  NOTE: There were 28 observations read from the data set EMW
      S4.VARCLUS2_OUTCLUSCORRPLOT.
3616  NOTE: The data set WORK.CORR_TMP has 28 observations and 62
       variables.
3617  NOTE: DATA statement used (Total process time):
3618        real time           0.00 seconds
3619        user cpu time       0.00 seconds
3620        system cpu time     0.00 seconds
3621        memory              160464.71k
3622        OS Memory           170880.00k
3623        Timestamp           07/01/2024 05:54:47 AM
3624        Step Count                          1  Switch Count  0
3625        Page Faults                         0
3626        Page Reclaims                       123
3627        Page Swaps                          0
3628        Voluntary Context Switches          10
3629        Involuntary Context Switches        0
3630        Block Input Operations              288
3631        Block Output Operations             264
3632
3633
3634
3635  NOTE: There were 28 observations read from the data set EMW
      S4.VARCLUS2_OUTCLUSCORRPLOT.
3636  NOTE: The data set EMWS4.VARCLUS2_OUTCLUSCORRPLOT has 28 ob
      servations and 29 variables.
3637  NOTE: DATA statement used (Total process time):
3638        real time           0.03 seconds
```

```
3639        user cpu time       0.02 seconds
3640        system cpu time     0.01 seconds
3641        memory              160464.71k
3642        OS Memory           170880.00k
3643        Timestamp           07/01/2024 05:54:47 AM
3644        Step Count                        1  Switch Count  0
3645        Page Faults                       0
3646        Page Reclaims                     3319
3647        Page Swaps                        0
3648        Voluntary Context Switches        30
3649        Involuntary Context Switches      0
3650        Block Input Operations            0
3651        Block Output Operations           264
3652
3653
3654
3655 NOTE: Deleting WORK.CORR_TMP (memtype=DATA).
3656
3657 NOTE: PROCEDURE DATASETS used (Total process time):
3658        real time           0.00 seconds
3659        user cpu time       0.00 seconds
3660        system cpu time     0.00 seconds
3661        memory              160464.71k
3662        OS Memory           170880.00k
3663        Timestamp           07/01/2024 05:54:47 AM
3664        Step Count                        1  Switch Count  0
3665        Page Faults                       0
3666        Page Reclaims                     48
3667        Page Swaps                        0
3668        Voluntary Context Switches        0
3669        Involuntary Context Switches      0
3670        Block Input Operations            0
3671        Block Output Operations           8
3672
3673
3674
```

```
3675 NOTE: There were 28 observations read from the data set EMW
     S4.VARCLUS2_OUTCLUSCORRPLOT.
3676 NOTE: The data set EMWS4.VARCLUS2_OUTCLUSCORRPLOT has 28 ob
     servations and 29 variables.
3677 NOTE: PROCEDURE SORT used (Total process time):
3678       real time           0.01 seconds
3679       user cpu time       0.00 seconds
3680       system cpu time     0.00 seconds
3681       memory              160464.71k
3682       OS Memory           170880.00k
3683       Timestamp           07/01/2024 05:54:47 AM
3684       Step Count                        1  Switch Count  0
3685       Page Faults                       0
3686       Page Reclaims                     115
3687       Page Swaps                        0
3688       Voluntary Context Switches        36
3689       Involuntary Context Switches      0
3690       Block Input Operations            288
3691       Block Output Operations           264
3692
3693
3694
3695 NOTE: There were 28 observations read from the data set EMW
     S4.VARCLUS2_OUTCLUSCORRPLOT.
3696 NOTE: The data set EMWS4.VARCLUS2_OUTCLUSCORRPLOT has 784 o
     bservations and 3 variables.
3697 NOTE: PROCEDURE TRANSPOSE used (Total process time):
3698       real time           0.01 seconds
3699       user cpu time       0.00 seconds
3700       system cpu time     0.00 seconds
3701       memory              160464.71k
3702       OS Memory           170880.00k
3703       Timestamp           07/01/2024 05:54:47 AM
3704       Step Count                        1  Switch Count  0
3705       Page Faults                       0
3706       Page Reclaims                     218
```

```
3707        Page Swaps                          0
3708        Voluntary Context Switches          39
3709        Involuntary Context Switches        0
3710        Block Input Operations              288
3711        Block Output Operations             528
3712
3713
3714
3715  NOTE: There were 784 observations read from the data set EM
      WS4.VARCLUS2_OUTCLUSCORRPLOT.
3716  NOTE: The data set EMWS4.VARCLUS2_OUTCLUSCORRPLOT has 784 o
      bservations and 3 variables.
3717  NOTE: DATA statement used (Total process time):
3718        real time           0.01 seconds
3719        user cpu time       0.00 seconds
3720        system cpu time     0.00 seconds
3721        memory              160464.71k
3722        OS Memory           170880.00k
3723        Timestamp           07/01/2024 05:54:47 AM
3724        Step Count                          1  Switch Count  0
3725        Page Faults                         0
3726        Page Reclaims                       464
3727        Page Swaps                          0
3728        Voluntary Context Switches          40
3729        Involuntary Context Switches        0
3730        Block Input Operations              288
3731        Block Output Operations             264
3732
3733
3734
3735  NOTE: There were 3 observations read from the data set EMWS
      4.VARCLUS2_OUTSTAT.
3736        WHERE _type_ in ('MEAN', 'N', 'STD');
3737  NOTE: The data set WORK.VARCLUS_TMP has 3 observations and
      62 variables.
3738  NOTE: DATA statement used (Total process time):
```

```
3739        real time           0.00 seconds
3740        user cpu time       0.00 seconds
3741        system cpu time     0.01 seconds
3742        memory              160464.71k
3743        OS Memory           170880.00k
3744        Timestamp           07/01/2024 05:54:47 AM
3745        Step Count                        1  Switch Count  0
3746        Page Faults                       0
3747        Page Reclaims                     256
3748        Page Swaps                        0
3749        Voluntary Context Switches        5
3750        Involuntary Context Switches      0
3751        Block Input Operations            0
3752        Block Output Operations           264
3753
3754
3755
3756 NOTE: There were 3 observations read from the data set WORK
     .VARCLUS_TMP.
3757 NOTE: The data set EMWS4.VARCLUS2_OUTSTATPLOT has 61 observ
     ations and 5 variables.
3758 NOTE: PROCEDURE TRANSPOSE used (Total process time):
3759        real time           0.00 seconds
3760        user cpu time       0.00 seconds
3761        system cpu time     0.00 seconds
3762        memory              160464.71k
3763        OS Memory           170880.00k
3764        Timestamp           07/01/2024 05:54:47 AM
3765        Step Count                        1  Switch Count  0
3766        Page Faults                       0
3767        Page Reclaims                     185
3768        Page Swaps                        0
3769        Voluntary Context Switches        15
3770        Involuntary Context Switches      0
3771        Block Input Operations            0
3772        Block Output Operations           528
```

```
3773
3774
3775
3776 NOTE: There were 61 observations read from the data set EMW
     S4.VARCLUS2_OUTSTATPLOT.
3777 NOTE: The data set EMWS4.VARCLUS2_OUTSTATPLOT has 61 observ
     ations and 6 variables.
3778 NOTE: DATA statement used (Total process time):
3779       real time           0.01 seconds
3780       user cpu time       0.00 seconds
3781       system cpu time     0.00 seconds
3782       memory              160464.71k
3783       OS Memory           170880.00k
3784       Timestamp           07/01/2024 05:54:47 AM
3785       Step Count                        1  Switch Count  0
3786       Page Faults                       0
3787       Page Reclaims                     469
3788       Page Swaps                        0
3789       Voluntary Context Switches        37
3790       Involuntary Context Switches      0
3791       Block Input Operations            288
3792       Block Output Operations           264
3793
3794
3795
3796 NOTE: There were 61 observations read from the data set EMW
     S4.VARCLUS2_OUTSTATPLOT.
3797 NOTE: The data set EMWS4.VARCLUS2_OUTSTATPLOT has 61 observ
     ations and 6 variables.
3798 NOTE: PROCEDURE SORT used (Total process time):
3799       real time           0.01 seconds
3800       user cpu time       0.01 seconds
3801       system cpu time     0.00 seconds
3802       memory              160464.71k
3803       OS Memory           170880.00k
3804       Timestamp           07/01/2024 05:54:47 AM
```

```
3805        Step Count                         1  Switch Count  0
3806        Page Faults                        0
3807        Page Reclaims                      118
3808        Page Swaps                         0
3809        Voluntary Context Switches         39
3810        Involuntary Context Switches       0
3811        Block Input Operations             288
3812        Block Output Operations            264
3813
3814
3815
3816  NOTE: Deleting WORK.VARCLUS_TMP (memtype=DATA).
3817
3818  NOTE: PROCEDURE DATASETS used (Total process time):
3819        real time           0.00 seconds
3820        user cpu time       0.00 seconds
3821        system cpu time     0.00 seconds
3822        memory              160464.71k
3823        OS Memory           170880.00k
3824        Timestamp           07/01/2024 05:54:47 AM
3825        Step Count                         1  Switch Count  0
3826        Page Faults                        0
3827        Page Reclaims                      49
3828        Page Swaps                         0
3829        Voluntary Context Switches         0
3830        Involuntary Context Switches       0
3831        Block Input Operations             0
3832        Block Output Operations            8
3833
3834
3835
3836  NOTE: There were 61 observations read from the data set EMW
      S4.VARCLUS2_OUTSTAT.
3837        WHERE _type_='CORR';
3838  NOTE: The data set EMWS4.VARCLUS2_OUTCORR has 61 observatio
      ns and 62 variables.
```

```
3839 NOTE: DATA statement used (Total process time):
3840       real time           0.00 seconds
3841       user cpu time       0.00 seconds
3842       system cpu time     0.00 seconds
3843       memory              160464.71k
3844       OS Memory           170880.00k
3845       Timestamp           07/01/2024 05:54:47 AM
3846       Step Count                        1  Switch Count  0
3847       Page Faults                       0
3848       Page Reclaims                     250
3849       Page Swaps                        0
3850       Voluntary Context Switches        17
3851       Involuntary Context Switches      0
3852       Block Input Operations            0
3853       Block Output Operations           264
3854
3855
3856
3857 NOTE: There were 61 observations read from the data set EMW
     S4.VARCLUS2_OUTCORR.
3858 NOTE: The data set EMWS4.VARCLUS2_OUTCORR has 61 observatio
     ns and 62 variables.
3859 NOTE: PROCEDURE SORT used (Total process time):
3860       real time           0.01 seconds
3861       user cpu time       0.00 seconds
3862       system cpu time     0.00 seconds
3863       memory              160464.71k
3864       OS Memory           170880.00k
3865       Timestamp           07/01/2024 05:54:47 AM
3866       Step Count                        1  Switch Count  0
3867       Page Faults                       0
3868       Page Reclaims                     115
3869       Page Swaps                        0
3870       Voluntary Context Switches        39
3871       Involuntary Context Switches      0
3872       Block Input Operations            288
```

```
3873       Block Output Operations           264
3874
3875
3876
3877 NOTE: There were 61 observations read from the data set EMW
     S4.VARCLUS2_OUTCORR.
3878 NOTE: The data set EMWS4.VARCLUS2_OUTCORRPLOT has 3721 obse
     rvations and 4 variables.
3879 NOTE: PROCEDURE TRANSPOSE used (Total process time):
3880       real time           0.01 seconds
3881       user cpu time       0.00 seconds
3882       system cpu time     0.01 seconds
3883       memory              160464.71k
3884       OS Memory           170880.00k
3885       Timestamp           07/01/2024 05:54:47 AM
3886       Step Count                        1  Switch Count  0
3887       Page Faults                       0
3888       Page Reclaims                     156
3889       Page Swaps                        0
3890       Voluntary Context Switches        27
3891       Involuntary Context Switches      0
3892       Block Input Operations            288
3893       Block Output Operations           1296
3894
3895
3896
3897 NOTE: There were 3721 observations read from the data set E
     MWS4.VARCLUS2_OUTCORRPLOT.
3898 NOTE: The data set EMWS4.VARCLUS2_OUTCORRPLOT has 3721 obse
     rvations and 5 variables.
3899 NOTE: DATA statement used (Total process time):
3900       real time           0.01 seconds
3901       user cpu time       0.00 seconds
3902       system cpu time     0.00 seconds
3903       memory              160464.71k
3904       OS Memory           170880.00k
```

```
3905        Timestamp           07/01/2024 05:54:47 AM
3906        Step Count                        1  Switch Count  0
3907        Page Faults                       0
3908        Page Reclaims                     183
3909        Page Swaps                        0
3910        Voluntary Context Switches        38
3911        Involuntary Context Switches      5
3912        Block Input Operations            1056
3913        Block Output Operations           1544
3914
3915
3916
3917 NOTE: There were 3721 observations read from the data set E
     MWS4.VARCLUS2_OUTCORRPLOT.
3918        WHERE _LABEL_ not = ' ';
3919 NOTE: The data set WORK.VARCLUS_MATCH has 3721 observations
       and 2 variables.
3920 NOTE: DATA statement used (Total process time):
3921        real time           0.00 seconds
3922        user cpu time       0.00 seconds
3923        system cpu time     0.00 seconds
3924        memory              160464.71k
3925        OS Memory           170880.00k
3926        Timestamp           07/01/2024 05:54:47 AM
3927        Step Count                        1  Switch Count  0
3928        Page Faults                       0
3929        Page Reclaims                     229
3930        Page Swaps                        0
3931        Voluntary Context Switches        15
3932        Involuntary Context Switches      0
3933        Block Input Operations            1568
3934        Block Output Operations           520
3935
3936
3937
3938 NOTE: Numeric values have been converted to character value
```

```
     s at the places given by: (Line):(Column).
3939         7:246
3940 NOTE: DATA statement used (Total process time):
3941         real time           0.00 seconds
3942         user cpu time       0.01 seconds
3943         system cpu time     0.00 seconds
3944         memory              160464.71k
3945         OS Memory           170880.00k
3946         Timestamp           07/01/2024 05:54:47 AM
3947         Step Count                        1  Switch Count  0
3948         Page Faults                       0
3949         Page Reclaims                     59
3950         Page Swaps                        0
3951         Voluntary Context Switches        0
3952         Involuntary Context Switches      0
3953         Block Input Operations            0
3954         Block Output Operations           0
3955
3956
3957
3958 NOTE: There were 3721 observations read from the data set W
     ORK.VARCLUS_MATCH.
3959 NOTE: The data set WORK.VARCLUS_MATCH has 3721 observations
      and 2 variables.
3960 NOTE: PROCEDURE SORT used (Total process time):
3961         real time           0.00 seconds
3962         user cpu time       0.00 seconds
3963         system cpu time     0.00 seconds
3964         memory              160464.71k
3965         OS Memory           170880.00k
3966         Timestamp           07/01/2024 05:54:47 AM
3967         Step Count                        1  Switch Count  0
3968         Page Faults                       0
3969         Page Reclaims                     154
3970         Page Swaps                        0
3971         Voluntary Context Switches        0
```

```
3972        Involuntary Context Switches      0
3973        Block Input Operations            0
3974        Block Output Operations           520
3975
3976
3977
3978  NOTE: There were 3721 observations read from the data set E
      MWS4.VARCLUS2_OUTCORRPLOT.
3979  NOTE: The data set EMWS4.VARCLUS2_OUTCORRPLOT has 3721 obse
      rvations and 5 variables.
3980  NOTE: PROCEDURE SORT used (Total process time):
3981        real time           0.01 seconds
3982        user cpu time       0.00 seconds
3983        system cpu time     0.00 seconds
3984        memory              160464.71k
3985        OS Memory           170880.00k
3986        Timestamp           07/01/2024 05:54:47 AM
3987        Step Count                        1  Switch Count  0
3988        Page Faults                       0
3989        Page Reclaims                     451
3990        Page Swaps                        0
3991        Voluntary Context Switches        32
3992        Involuntary Context Switches      0
3993        Block Input Operations            0
3994        Block Output Operations           1552
3995
3996
3997
3998  WARNING: Multiple lengths were specified for the BY variabl
      e _NAME_ by input data sets. This might cause unexpected re
      sults.
3999  NOTE: MERGE statement has more than one data set with repea
      ts of BY values.
4000  NOTE: There were 3721 observations read from the data set W
      ORK.VARCLUS_MATCH.
4001  NOTE: There were 3721 observations read from the data set E
```

```
      MWS4.VARCLUS2_OUTCORRPLOT.
4002  NOTE: The data set EMWS4.VARCLUS2_OUTCORRPLOT has 3721 obse
      rvations and 3 variables.
4003  NOTE: DATA statement used (Total process time):
4004        real time           0.01 seconds
4005        user cpu time       0.00 seconds
4006        system cpu time     0.01 seconds
4007        memory              160464.71k
4008        OS Memory           170880.00k
4009        Timestamp           07/01/2024 05:54:47 AM
4010        Step Count                        1  Switch Count  0
4011        Page Faults                       0
4012        Page Reclaims                     629
4013        Page Swaps                        0
4014        Voluntary Context Switches        39
4015        Involuntary Context Switches      0
4016        Block Input Operations            1568
4017        Block Output Operations           1288
4018
4019
4020
4021  NOTE: There were 3721 observations read from the data set E
      MWS4.VARCLUS2_OUTCORRPLOT.
4022  NOTE: The data set EMWS4.VARCLUS2_OUTCORRPLOT has 3721 obse
      rvations and 3 variables.
4023  NOTE: PROCEDURE SORT used (Total process time):
4024        real time           0.01 seconds
4025        user cpu time       0.01 seconds
4026        system cpu time     0.00 seconds
4027        memory              160464.71k
4028        OS Memory           170880.00k
4029        Timestamp           07/01/2024 05:54:47 AM
4030        Step Count                        1  Switch Count  0
4031        Page Faults                       0
4032        Page Reclaims                     465
4033        Page Swaps                        0
```

```
4034       Voluntary Context Switches        43
4035       Involuntary Context Switches      0
4036       Block Input Operations            1312
4037       Block Output Operations           1288
4038
4039
4040
4041 NOTE: Deleting WORK.VARCLUS_MATCH (memtype=DATA).
4042
4043 NOTE: PROCEDURE DATASETS used (Total process time):
4044       real time           0.00 seconds
4045       user cpu time       0.00 seconds
4046       system cpu time     0.00 seconds
4047       memory              160464.71k
4048       OS Memory           170880.00k
4049       Timestamp           07/01/2024 05:54:47 AM
4050       Step Count                        1  Switch Count  0
4051       Page Faults                       0
4052       Page Reclaims                     49
4053       Page Swaps                        0
4054       Voluntary Context Switches        0
4055       Involuntary Context Switches      0
4056       Block Input Operations            0
4057       Block Output Operations           8
4058
4059
4060
4061 NOTE: There were 89 observations read from the data set EMW
     S4.VARCLUS2_OUTRSQUARE.
4062 NOTE: The data set EMWS4.VARCLUS2_OUTLINK has 89 observatio
     ns and 9 variables.
4063 NOTE: DATA statement used (Total process time):
4064       real time           0.00 seconds
4065       user cpu time       0.00 seconds
4066       system cpu time     0.00 seconds
4067       memory              160464.71k
```

```
4068        OS Memory           170880.00k
4069        Timestamp           07/01/2024 05:54:47 AM
4070        Step Count                      1  Switch Count  0
4071        Page Faults                     0
4072        Page Reclaims                   245
4073        Page Swaps                      0
4074        Voluntary Context Switches      27
4075        Involuntary Context Switches    0
4076        Block Input Operations          288
4077        Block Output Operations         264
4078
4079
4080
4081 NOTE: There were 89 observations read from the data set EMW
     S4.VARCLUS2_OUTRSQUARE.
4082 NOTE: The data set EMWS4.VARCLUS2_OUTNODE has 89 observatio
     ns and 3 variables.
4083 NOTE: DATA statement used (Total process time):
4084        real time           0.00 seconds
4085        user cpu time       0.00 seconds
4086        system cpu time     0.00 seconds
4087        memory              160464.71k
4088        OS Memory           170880.00k
4089        Timestamp           07/01/2024 05:54:47 AM
4090        Step Count                      1  Switch Count  0
4091        Page Faults                     0
4092        Page Reclaims                   358
4093        Page Swaps                      0
4094        Voluntary Context Switches      19
4095        Involuntary Context Switches    0
4096        Block Input Operations          0
4097        Block Output Operations         264
4098
4099
4100
4101 NOTE: There were 1422 observations read from the data set E
```

```
      MWS4.VARCLUS2_OUTSTAT.
4102  NOTE: The data set EMWS4.VARCLUS2_OUTSTATSCORE has 30 obser
      vations and 62 variables.
4103  NOTE: DATA statement used (Total process time):
4104        real time           0.00 seconds
4105        user cpu time       0.00 seconds
4106        system cpu time     0.01 seconds
4107        memory              160464.71k
4108        OS Memory           170880.00k
4109        Timestamp           07/01/2024 05:54:47 AM
4110        Step Count                        1  Switch Count  0
4111        Page Faults                       0
4112        Page Reclaims                     250
4113        Page Swaps                        0
4114        Voluntary Context Switches        19
4115        Involuntary Context Switches      0
4116        Block Input Operations            0
4117        Block Output Operations           264
4118
4119
4120
4121  NOTE: The file _FILE_ is:
4122        Filename=/home/u63452984/case-study-s2192852/Workspac
      es/EMWS4/VarClus2/EMFLOWSCORE.sas,
4123        Owner Name=u63452984,Group Name=oda,
4124        Access Permission=-rw-r--r--,
4125        Last Modified=07 January 2024 05:54:44,
4126        File Size (bytes)=19057
4127
4128  NOTE: 94 records were written to the file _FILE_.
4129        The minimum record length was 1.
4130        The maximum record length was 103.
4131  NOTE: DATA statement used (Total process time):
4132        real time           0.25 seconds
4133        user cpu time       0.25 seconds
4134        system cpu time     0.00 seconds
```

```
4135       memory              160464.71k
4136       OS Memory           170880.00k
4137       Timestamp           07/01/2024 05:54:47 AM
4138       Step Count                        1  Switch Count  0
4139       Page Faults                       0
4140       Page Reclaims                     3393
4141       Page Swaps                        0
4142       Voluntary Context Switches        18
4143       Involuntary Context Switches      0
4144       Block Input Operations            288
4145       Block Output Operations           2584
4146
4147
4148
4149 NOTE: The file _OUT is:
4150       Filename=/home/u63452984/case-study-s2192852/Workspac
     es/EMWS4/VarClus2/EMPUBLISHSCORE.sas,
4151       Owner Name=u63452984,Group Name=oda,
4152       Access Permission=-rw-r--r--,
4153       Last Modified=07 January 2024 05:54:47
4154
4155 NOTE: 731 records were written to the file _OUT.
4156       The minimum record length was 1.
4157       The maximum record length was 103.
4158 NOTE: DATA statement used (Total process time):
4159       real time           0.01 seconds
4160       user cpu time       0.01 seconds
4161       system cpu time     0.00 seconds
4162       memory              160464.71k
4163       OS Memory           170880.00k
4164       Timestamp           07/01/2024 05:54:47 AM
4165       Step Count                        1  Switch Count  0
4166       Page Faults                       0
4167       Page Reclaims                     35
4168       Page Swaps                        0
4169       Voluntary Context Switches        10
```

```
4170        Involuntary Context Switches      0
4171        Block Input Operations            0
4172        Block Output Operations           56
4173
4174
4175  NOTE: Fileref _IN has been deassigned.
4176  NOTE: Fileref _OUT has been deassigned.
4177  18172  *-------------------------------------------------
      ---------*;
4178  18173  * End TRAIN: VarClus2;
4179  18174  *-------------------------------------------------
      ---------*;
4180  18175
4181
4182  18176  *-------------------------------------------------
      ---------*;
4183  18177  * Close any missing semi colons;
4184  18178  *-------------------------------------------------
      ---------*;
4185  18179  ;
4186  18180  ;
4187  18181  ;
4188  18182  ;
4189  18183  quit;
4190  18184  *-------------------------------------------------
      ---------*;
4191  18185  * Close any unbalanced quotes;
4192  18186  *-------------------------------------------------
      ---------*;
4193  18187  /*; *"; *'; */
4194  18188  ;
4195  18189  run;
4196  18190  quit;
4197  18191  /* Reset EM Options */
4198  18192  options formchar="|----|+|---+=|-/\<>*";
4199  18193  options nocenter ls=256 ps=10000;
```

```
4200 18194  goptions reset=all device=GIF NODISPLAY;
4201
4202 *-------------------------------------------------------------
     --*
4203 * Score Log
4204 Date:                07 January 2024
4205 Time:                05:54:48
4206 *-------------------------------------------------------------
     --*
4207 18296  %let EMEXCEPTIONSTRING=;
4208 18297  *-------------------------------------------------------
     ---------*;
4209 18298  * SCORE: VarClus2;
4210 18299  *-------------------------------------------------------
     ---------*;
4211 18300  %let EM_ACTION = SCORE;
4212 18301  %let syscc = 0;
4213 18302
4214 18303  %macro main;
4215 18304
4216 18305     filename temp catalog 'sashelp.emexpl.variableclu
     stering_macros.source';
4217 18306     %include temp;
4218 18307     filename temp catalog 'sashelp.emexpl.variableclu
     stering_macros2.source';
4219 18308     %include temp;
4220 18309     filename temp;
4221 18310
4222 18311    %SetProperties;
4223 18312
4224 18313     %if %upcase(&EM_ACTION) = CREATE %then %do;
4225 18314         filename temp catalog 'sashelp.emexpl.variabl
     eclustering_create.source';
4226 18315         %include temp;
4227 18316         filename temp;
4228 18317         %create;
```

```
4229 18318     %end;
4230 18319     %else
4231 18320     %if %upcase(&EM_ACTION) = TRAIN %then %do;
4232 18321          filename temp catalog 'sashelp.emexpl.variab
     leclustering_train.source';
4233 18322             %include temp;
4234 18323             filename temp;
4235 18324             %train;
4236 18325     %end;
4237 18326     %else
4238 18327     %if %upcase(&EM_ACTION) = SCORE %then %do;
4239 18328             filename temp catalog 'sashelp.emexpl.var
     iableclustering_score.source';
4240 18329             %include temp;
4241 18330             filename temp;
4242 18331             %score;
4243 18332     %end;
4244 18333     %else
4245 18334     %if %upcase(&EM_ACTION) = REPORT %then %do;
4246 18335             filename temp catalog 'sashelp.emexpl.var
     iableclustering_report.source';
4247 18336             %include temp;
4248 18337             filename temp;
4249 18338             %report;
4250 18339     %end;
4251 18340     /*
4252 18341     %if %upcase(&EM_ACTION) = OPENTESTTABLE %then %do
     ;
4253 18342         %put 'OPENING TABLE';
4254 18343     %end;
4255 18344     %if %upcase(&EM_ACTION) = CLOSETESTTABLE %then %d
     o;
4256 18345         %put 'CLOSE TABLE';
4257 18346     %end;
4258 18347     */
4259 18348  %mend main;
```

```
4260 18349  %main;
4261 NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMEXPL.V
     ARIABLECLUSTERING_MACROS.SOURCE.
4262 18350 +
4263 18351 +/* Initialize property macro variables */
4264 18352 +%macro SetProperties;
4265 18353 +   %em_checkmacro(name=EM_PROPERTY_MAXCLUS,       gl
     obal=Y, value=DEFAULT);
4266 18354 +   %em_checkmacro(name=EM_PROPERTY_HIDEVARIABLE,  gl
     obal=Y, value=Y);
4267 18355 +   %em_checkmacro(name=EM_PROPERTY_PRINTOPTION,   gl
     obal=Y, value=SHORT);
4268 18356 +   %em_checkmacro(name=EM_PROPERTY_CLUSSOURCE,    gl
     obal=Y, value=CORR);
4269 18357 +   %em_checkmacro(name=EM_PROPERTY_CLUSCOMP,      gl
     obal=Y, value=PRINCIPAL);
4270 18358 +   %em_checkmacro(name=EM_PROPERTY_CLUSHIERACHY,
       global=Y, value=Y);
4271 18359 +   %em_checkmacro(name=EM_PROPERTY_INCLUDECLASSVAR,
         global=Y, value=N);
4272 18360 +   %em_checkmacro(name=EM_PROPERTY_EXPORTEDCOMP,
       global=Y, value=CLUSTERCOMP);
4273 18361 +   %em_checkmacro(name=EM_PROPERTY_MAXEIGEN,
      global=Y, value=DEFAULT);
4274 18362 +   %em_checkmacro(name=EM_PROPERTY_PROPORTION,
     global=Y, value=DEFAULT);
4275 18363 +   %em_checkmacro(name=EM_PROPERTY_PRINTOPTION,
      global=Y, value=SHORT);
4276 18364 +   %em_checkmacro(name=EM_PROPERTY_TWOSTAGECLUS,
       global=Y, value=AUTO);
4277 18365 +   %em_checkmacro(name=EM_PROPERTY_SUPPRESSSAMPWARN,
           global=Y, value=N);
4278 18366 +
4279 18367 +%mend SetProperties;
4280 18368 +
4281 18369 +%Macro MakeDummyVariables(indata=,
```

```
4282  18370 +                          outvar=,
4283  18371 +                          outdata=,
4284  18372 +                          fileref=,
4285  18373 +                          recreatecmeta=N, /* option
      al */
4286  18374 +                          incmeta=,     /* optional
      */
4287  18375 +                          outcmeta=,    /* optional
      */
4288  18376 +                          ndummyvars=_ndummyvars
4289  18377 +                          );
4290  18378 +    %global &ndummyvars;
4291  18379 +
4292  18380 +    proc dmdb batch data=&indata out=_dmdbdat dmdbca
      t=_dmdbcat classout=_classout;;
4293  18381 +       class
4294  18382 +       %EM_BINARY_INPUT %EM_NOMINAL_INPUT %EM_ORDINA
      L_INPUT
4295  18383 +       %EM_BINARY_REJECTED %EM_NOMINAL_REJECTED %EM_
      ORDINAL_REJECTED
4296  18384 +    ;
4297  18385 +    run;
4298  18386 +    %let &ndummyvars = 0;
4299  18387 +    data _null_;
4300  18388 +    %let dsid = %sysfunc(open(work._classout));
4301  18389 +      %let &ndummyvars = %sysfunc(attrn(&dsid, NOBS)
      );
4302  18390 +    %let dsid = %sysfunc(close(&dsid));
4303  18391 +    run;
4304  18392 +
4305  18393 +    proc dmzip data=_dmdbdat dmdbcat=_dmdbcat;
4306  18394 +       input
4307  18395 +       %EM_BINARY_INPUT %EM_NOMINAL_INPUT %EM_ORDINA
      L_INPUT
4308  18396 +       %EM_BINARY_REJECTED %EM_NOMINAL_REJECTED %EM_
      ORDINAL_REJECTED
```

```
4309  18397 +        / level=nominal stdize=no;
4310  18398 +       make outvar = &outvar;
4311  18399 +       score data = &indata out =&outdata;
4312  18400 +       code  file=  "&fileref";
4313  18401 +    run;
4314  18402 +    %if &recreatecmeta eq Y %then %do;
4315  18403 +    proc contents data =&outvar out=_tmpds(keep=NAME
      LABEL);
4316  18404 +    data _tmpds;
4317  18405 +        set _tmpds;
4318  18406 +           ROLE = 'INPUT';
4319  18407 +           LEVEL = 'INTERVAL';
4320  18408 +           CREATOR='DMZIP';
4321  18409 +           if NAME = '_TYPE_' then delete;
4322  18410 +    run;
4323  18411 +    data &outcmeta;
4324  18412 +         set &incmeta _tmpds;
4325  18413 +    run;
4326  18414 +    %end;
4327  18415 +    proc datasets lib=work nolist;
4328  18416 +      delete  _dmdbdat _dmdbcat _classout
4329  18417 +    %if &recreatecmeta eq Y %then %do;
4330  18418 +    _tmpds
4331  18419 +    %end;
4332  18420 +    ;
4333  18421 +    quit;
4334  18422 +%Mend MakeDummyVariables;
4335  18423 +
4336  18424 +/*--- Determine Optimal Number of Cluster ----
4337  18425 +%macro FindClusNum(statds=, groupds=, minvariation=)
     ;
4338  18426 +   %global optnclus;
4339  18427 +   data varclus_tmp(drop=_NAME_);
4340  18428 +      set &statDs;
4341  18429 +      where _type_ ='PROPOR';
4342  18430 +   run;
```

```
4343 18431 +   proc sort data=varclus_tmp;
4344 18432 +      by _NCL_;
4345 18433 +   run;
4346 18434 +   proc transpose data=varclus_tmp out=varclus_tmp;
4347 18435 +      by _NCL_;
4348 18436 +      var %EM_INTERVAL_INPUT
4349 18437 +      %if &EM_PROPERTY_INCLUDECLASSVAR eq Y %then %d
     o;
4350 18438 +      %let dsid = %sysfunc(open(&EM_USER_OUTDUMMY));
4351 18439 +      %let nvar = %sysfunc(attrn(&dsid, NVAR));
4352 18440 +          %do i = 2 %to &nvar;
4353 18441 +          %let varname = %sysfunc(varname(&dsid, &i)
     );
4354 18442 +          &varname
4355 18443 +          %end;
4356 18444 +      %end;
4357 18445 +   ;
4358 18446 +   run;
4359 18447 +
4360 18448 +   %if &minVariation eq %then %do;
4361 18449 +       %let minVariation = &EM_PROPERTY_MINVARIATION
     ;
4362 18450 +   %end;
4363 18451 +   %if ^(0<&minVariation<100) %then %do;
4364 18452 +       %let minVariation = 90;
4365 18453 +   %end;
4366 18454 +
4367 18455 +   data _null_;
4368 18456 +      set varclus_tmp end=eof;
4369 18457 +      by _NCL_;
4370 18458 +      retain flag 0;
4371 18459 +      if first._ncl_ then flag=0;
4372 18460 +      if .<col1 < &minVariation then flag=1;
4373 18461 +      if last._ncl_ and ^flag then do;
4374 18462 +         call symput('OPTNCL', _ncl_);
4375 18463 +         stop;
```

```
4376 18464 +      end;
4377 18465 +     if eof then call symput('OPTNCL', _ncl_);
4378 18466 +   run;
4379 18467 +
4380 18468 +   %let optnclus = &OPTNCL;
4381 18469 +
4382 18470 +   data varclus_tmp(drop=_NCL_ _NAME_);
4383 18471 +      set &statDs;
4384 18472 +      where _type_ in('RSQUARED' 'GROUP') and _NCL_=
     &OPTNCL;
4385 18473 +   run;
4386 18474 +   proc sort data=varclus_tmp;
4387 18475 +      by _TYPE_;
4388 18476 +   run;
4389 18477 +   proc transpose data=varclus_tmp out=varclus_tmp;
4390 18478 +      by _TYPE_;
4391 18479 +   run;
4392 18480 +   proc sort data=varclus_tmp;
4393 18481 +      by _name_ _type_;
4394 18482 +   run;
4395 18483 +
4396 18484 +  proc transpose data=varclus_tmp out=&groupds;
4397 18485 +      by _NAME_;
4398 18486 +   run;
4399 18487 +   proc sort data=&groupDs(rename=(col1=Cluster col2
     =Rsquare _NAME_=VARIABLE));
4400 18488 +      by Cluster descending Rsquare;
4401 18489 +      where Cluster ne 0;
4402 18490 +   run;
4403 18491 +   proc datasets lib=work nolist mt=(DATA VIEW);
4404 18492 +      delete varclus_tmp;
4405 18493 +   run;
4406 18494 +   quit;
4407 18495 +%mend findClusNum;
4408 18496 +*/
4409 18497 +
```

```
4410 18498 +%macro getNclusfromTrain(inoutstat=, nc=);
4411 18499 +%global &nc;
4412 18500 +data _null_;
4413 18501 +    set &inoutstat end=eof;
4414 18502 +    if eof then do;
4415 18503 +    call symput("&nc", _ncl_);
4416 18504 +    end;
4417 18505 +run;
4418 18506 +%mend  getNclusfromTrain;
4419 18507 +
4420 18508 +%macro MakeDeltaCode(groupds=, outstatscore=, deltac
     odefile=);
4421 18509 +
4422 18510 +     *--- Build Code to Modify Metadata ---*;
4423 18511 +     filename X "&deltacodefile";
4424 18512 +     data _null_;
4425 18513 +        FILE X;
4426 18514 +        set &groupds end=eof;
4427 18515 +        /*by Cluster;*/
4428 18516 +         if _N_=1 then do;
4429 18517 +           %if &EM_PROPERTY_INCLUDECLASSVAR eq Y %th
     en %do;
4430 18518 +            put "if upcase(strip(ROLE)) ='INPUT' and
      upcase(strip(LEVEL)) ^='INTERVAL' then ROLE ='REJECTED' ;"
     ;
4431 18519 +           %end;
4432 18520 +           put "if upcase(strip(ROLE))='INPUT' and u
     pcase(strip(LEVEL))='INTERVAL' then do;";
4433 18521 +           put "if upcase(strip(NAME)) in (";
4434 18522 +        end;
4435 18523 +        if Strip(upcase(Selected)) eq 'YES' then do;
4436 18524 +           string = '"'!!trim(left(VARIABLE))!!'"';
4437 18525 +           put string;
4438 18526 +        end;
4439 18527 +        if eof then do;
4440 18528 +           put ') then ROLE="INPUT";';
```

```
4441  18529 +           put 'else ROLE="REJECTED";';
4442  18530 +           put 'end;';
4443  18531 +
4444  18532 +           %if %upcase(&EM_PROPERTY_HIDEVARIABLE) eq
       Y %then %do;
4445  18533 +             put 'if upcase(strip(ROLE)) = "REJECTED
     " then delete ;';
4446  18534 +           %end;
4447  18535 +        end;
4448  18536 +     run;
4449  18537 +     quit;
4450  18538 +
4451  18539 +     filename X;
4452  18540 +
4453  18541 +     quit;
4454  18542 +%mend MakeDeltaCode;
4455  18543 +
4456  18544 +%macro MakeVarClusCorrData(statds=, corrds=, corrplo
     tds= );
4457  18545 +    %if ^%sysfunc(exist(&statds)) %then %do;
4458  18546 +         %goto doendc;
4459  18547 +    %end;
4460  18548 +
4461  18549 +    data &corrds(drop=_TYPE_ _NCL_) ;
4462  18550 +       set &statds;
4463  18551 +       where _type_ eq 'CORR' ;
4464  18552 +    run ;
4465  18553 +    proc sort data=&corrds;
4466  18554 +       by _NAME_ ;
4467  18555 +    run ;
4468  18556 +    proc transpose data=&corrds out=&corrplotds name
     =_TMP_;
4469  18557 +      BY _NAME_ ;
4470  18558 +    run ;
4471  18559 +    data &corrplotds;
4472  18560 +       length _Y_ $100;
```

```
4473  18561 +       set &corrplotDs;
4474  18562 +       if _LABEL_ ne '' then _Y_=_LABEL_ ; else _Y_=
      _TMP_ ;
4475  18563 +    run ;
4476  18564 +    data varclus_match(rename=(_TMP_= _NAME_ _LABEL_
      =_X_));
4477  18565 +       set &corrplotds;
4478  18566 +       where _LABEL_ ne '' ;
4479  18567 +       keep _TMP_ _LABEL_ ;
4480  18568 +    run ;
4481  18569 +    data _null_;
4482  18570 +       nobs=0;
4483  18571 +       dsid = open('varclus_match');
4484  18572 +       if dsid then do;
4485  18573 +          nobs = attrn(dsid, 'NOBS');
4486  18574 +          dsid = close(dsid);
4487  18575 +       end;
4488  18576 +       call symput ('CORR_NOBS', nobs);
4489  18577 +    run;
4490  18578 +    %if &corr_nobs %then %do;
4491  18579 +        proc sort data=varclus_match;
4492  18580 +           by _name_;
4493  18581 +        run ;
4494  18582 +        proc sort data=&corrplotds;
4495  18583 +           by _name_;
4496  18584 +        run ;
4497  18585 +        data &corrplotds(keep= _X_ _Y_ col1 rename=(
      col1=Correlation)) ;
4498  18586 +           merge varclus_match &corrplotds;
4499  18587 +           by _NAME_ ;
4500  18588 +           if _X_ eq '' then _X_=_NAME_ ;
4501  18589 +           label _X_ = "%sysfunc(sasmsg(sashelp.dmin
      e, rpt_varclus_label_variable, noquote))";
4502  18590 +           label _Y_ = "%sysfunc(sasmsg(sashelp.dmin
      e, rpt_varclus_label_variable, noquote))";
4503  18591 +           label col1 = "%sysfunc(sasmsg(sashelp.dmi
```

```
     ne, rpt_correlation_vlabel, noquote))";
4504 18592 +
4505 18593 +       run ;
4506 18594 +    %end;
4507 18595 +    %else %do;
4508 18596 +        proc sort data=&corrplotds;
4509 18597 +           by _name_;
4510 18598 +        run ;
4511 18599 +        data &corrplotds(keep= _NAME_ _Y_ col1 renam
     e=(_NAME_=_X_ col1=Correlation)) ;
4512 18600 +           set &corrplotds;
4513 18601 +           label _NAME_ = "%sysfunc(sasmsg(sashelp.d
     mine, rpt_varclus_label_variable, noquote))" ;
4514 18602 +           label _Y_ = "%sysfunc(sasmsg(sashelp.dmin
     e, rpt_varclus_label_variable, noquote))" ;
4515 18603 +           label col1 = "%sysfunc(sasmsg(sashelp.dmi
     ne, rpt_correlation_vlabel, noquote))";
4516 18604 +
4517 18605 +        run ;
4518 18606 +    %end;
4519 18607 +    proc sort data=&corrplotds;
4520 18608 +       by _X_ _Y_;
4521 18609 +    run ;
4522 18610 +    proc datasets lib=work nolist mt=(DATA VIEW);
4523 18611 +       delete varclus_match;
4524 18612 +    run;
4525 18613 +    quit;
4526 18614 +
4527 18615 +%doendc:
4528 18616 +
4529 18617 +%mend MakeVarClusCorrData;
4530 18618 +
4531 18619 +%macro MakeStatPlotData(statds= , outstatplotds=);
4532 18620 +   %if %sysfunc(exist(&statds)) %then %do;
4533 18621 +
4534 18622 +       data varclus_tmp(drop=_NAME_ _NCL_) ;
```

```
4535 18623 +          set &statDs;
4536 18624 +          where _type_ in('MEAN', 'STD', 'N');
4537 18625 +       run ;
4538 18626 +       proc transpose data=varclus_tmp out=&outstatp
     lotds;
4539 18627 +          id _TYPE_ ;
4540 18628 +       run ;
4541 18629 +       data &outstatplotds;
4542 18630 +          set &outstatplotds(obs=1000);
4543 18631 +          label _name_= "%sysfunc(sasmsg(sashelp.dmi
     ne, rpt_varclus_label_name, noquote))";
4544 18632 +          label _label_="%sysfunc(sasmsg(sashelp.dmi
     ne, rpt_varclus_label_label, noquote))" ;
4545 18633 +          if MEAN ne 0 then SCALEDSTD= STD / MEAN ;
4546 18634 +          else SCALEDSTD= STD ;
4547 18635 +          label SCALEDSTD = "%sysfunc(sasmsg(sashelp
     .dmine, rpt_varclus_label_scaledstd, noquote))";
4548 18636 +       run ;
4549 18637 +       proc sort data=&outstatplotds;
4550 18638 +          by descending SCALEDSTD ;
4551 18639 +       run ;
4552 18640 +       proc datasets lib=work nolist mt=(DATA VIEW);
4553 18641 +          delete varclus_tmp;
4554 18642 +       run;
4555 18643 +       quit;
4556 18644 +    %end;
4557 18645 +
4558 18646 +%mend MakeStatPlotData;
4559 18647 +
4560 18648 +
4561 18649 +%macro CreateScoreCode(indata=, ncluscomp=, fileref=
     );
4562 18650 +     %EM_GETNAME(KEY=OUTSTATSCORE, type=DATA);
4563 18651 +    data &EM_USER_OUTSTATSCORE;
4564 18652 +          set &indata;
4565 18653 +          if (_TYPE_ in ('SCORE' 'MEAN' 'STD') and _
```

```
     NCL_ = &ncluscomp ) or (_TYPE_ in ('MEAN' 'STD'));
4566 18654 +          if _TYPE_ = 'MEAN' then _NAME_='MEAN';
4567 18655 +          if _TYPE_ = 'STD' then _NAME_='STD';
4568 18656 +          DROP _TYPE_ _NCL_;
4569 18657 +     run;
4570 18658 +
4571 18659 +     filename _file_  "&fileRef";
4572 18660 +
4573 18661 +     data _null_;
4574 18662 +        FILE _file_ MOD;
4575 18663 +        put ' ';
4576 18664 +        put '/*-----------------------------------
     ------------*/';
4577 18665 +        put '/* ' "%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_score_title_begin , noquote))" '*/';
4578 18666 +        put '/*-----------------------------------
     ------------*/';
4579 18667 +        put ' ';
4580 18668 +        %let dsid = %sysfunc(open(&EM_USER_OUTSTATSC
     ORE));
4581 18669 +
4582 18670 +        %let nvar = %sysfunc(attrn(&dsid, NVAR));
4583 18671 +        %let vn_name =%sysfunc(varnum(&dsid, _NAME_)
     );
4584 18672 +
4585 18673 +        %let k = 1;
4586 18674 +        %do %while(^%sysfunc(fetch(&dsid)));
4587 18675 +               %let _name = %sysfunc(getvarc(&dsid,
      &vn_name));
4588 18676 +               %if &k > 2 %then %do;
4589 18677 +                %let cn = %eval(&k-2);
4590 18678 +                put "&_name = 0 ; /*---" "%sysfunc(
     sasmsg(sashelp.dmine, rpt_varclus_score_cluscompnum, noquot
     e, &cn))"  "------ */";
4591 18679 +               %end;
4592 18680 +               %let k = %eval(&k+1);
```

```
4593 18681 +        %end;
4594 18682 +
4595 18683 +        %let rc = %sysfunc(rewind(&dsid));
4596 18684 +
4597 18685 +        %do i= 2 %to &nvar;
4598 18686 +            %let _varname =  %sysfunc(varname(&dsid,
      &i));
4599 18687 +            %do %while(^%sysfunc(fetch(&dsid)));
4600 18688 +                %let _name = %sysfunc(getvarc(&dsid,
      &vn_name));
4601 18689 +                %if &_name = MEAN %then
4602 18690 +                %let _mean = %sysfunc(getvarn(&dsid,
      &i));
4603 18691 +                %else %if &_name = STD %then
4604 18692 +                %let _std = %sysfunc(getvarn(&dsid,
     &i));
4605 18693 +                %else %do;
4606 18694 +                      %let coeff =  %sysfunc(getvarn
     (&dsid, &i));
4607 18695 +                      %let abscoeff = %sysfunc(abs(&
     coeff));
4608 18696 +                         %if &abscoeff >  0 %then %
     do;
4609 18697 +                       put "&_name = &_name+&coeff *
      (&_varname - &_mean)/&_std;";
4610 18698 +                          %end;
4611 18699 +                %end;
4612 18700 +             %end;
4613 18701 +             %let rc = %sysfunc(rewind(&dsid));
4614 18702 +         %end;
4615 18703 +
4616 18704 +        %let dsid= %sysfunc(close(&dsid));
4617 18705 +       run;
4618 18706 +%mend CreateScoreCode;
4619 18707 +
4620 18708 +
```

```
4621 18709 +
4622 18710 +/*----------------------------------------------
     --------
4623 18711 +    Instead of using %MakeRSquareData,
4624 18712 +    %MakeVarClusResultTable at macro2.source is used
4625 18713 + +---------------------------------------------
     --------*/
4626 18714 +
4627 18715 +
4628 18716 +%macro MakeRSquareData(indata=, inClusRSquare=, outd
     ata=, ncluster=);
4629 18717 +
4630 18718 +/* modifying  from ods rsquare = data */
4631 18719 +
4632 18720 +data &outdata(drop= ControlVar  NumberOfClusters Cur
     rentCluster);
4633 18721 +    Length Cluster $16;
4634 18722 +    length Variable $32;
4635 18723 +    Length VariableLabel $64;
4636 18724 +    set &indata; retain CurrentCluster;
4637 18725 +    if NumberOfClusters ^= &ncluster then delete;
4638 18726 +    if strip(Cluster) eq '' then Cluster = CurrentCl
     uster;
4639 18727 +    CurrentCluster = Cluster;
4640 18728 +   run;
4641 18729 +proc sort data =&outdata ;
4642 18730 +     by Cluster RsquareRatio;
4643 18731 +run;
4644 18732 +data _tmprsq(drop=index);
4645 18733 +     set &outdata; by Cluster;
4646 18734 +     if first.Cluster then do;
4647 18735 +     index = strip(scan(Cluster,2 ));
4648 18736 +     Variable = "Clus"||index;
4649 18737 +     VariableLabel = "Cluster Component "||index;
4650 18738 +     OwnCluster = 1;
4651 18739 +     NextClosest = .;
```

```
4652 18740 +     RsquareRatio = 0;
4653 18741 +     output;
4654 18742 +     end;
4655 18743 +run;
4656 18744 +
4657 18745 +proc sort data = _tmprsq ;
4658 18746 +     by Cluster RsquareRatio;
4659 18747 +run;
4660 18748 +data &outdata;
4661 18749 +     set &outdata _tmprsq;
4662 18750 +by Cluster;
4663 18751 +run;
4664 18752 +
4665 18753 +
4666 18754 +/* Just create the Selected variable with all YES */
4667 18755 +
4668 18756 +data &outdata;
4669 18757 +    set &outdata; by cluster;
4670 18758 +    length Selected $8;
4671 18759 +    Selected = 'YES';
4672 18760 +    label  OwnCluster = 'R-Sqaure with Cluster Compo
     nent';
4673 18761 +    label  NextClosest = 'R-Sqaure with Next Cluster
      Component';
4674 18762 +    rename OwnCluster = RSqWithClusterComp;
4675 18763 +    rename NextClosest = RSqWithNextClusComp;
4676 18764 +run;
4677 18765 +
4678 18766 +
4679 18767 +/* Selected = Y/N will be done %score section -----
4680 18768 +
4681 18769 +%if &EM_PROPERTY_EXPORTEDCOMP ne CLUSTERCOMP %then %
     do;
4682 18770 +data &outdata;
4683 18771 +    set &outdata; by cluster;
4684 18772 +    length Selected $8;
```

```
4685 18773 +    if first.Cluster then Selected = 'YES';
4686 18774 +    else Selected = 'NO';
4687 18775 +    label  OwnCluster = 'R-Sqaure with Cluster Compo
     nent';
4688 18776 +    label  NextClosest = 'R-Sqaure with Next Cluster
      Component';
4689 18777 +    rename OwnCluster = RSqWithClusterComp;
4690 18778 +    rename NextClosest = RSqWithNextClusComp;
4691 18779 +run;
4692 18780 +%end;
4693 18781 +%else %do;
4694 18782 +data &outdata;
4695 18783 +    set &outdata; by cluster;
4696 18784 +    if last.Cluster then Selected = 'YES';
4697 18785 +    else Selected = 'NO';
4698 18786 +    label  OwnCluster = 'R-Sqaure with Cluster Compo
     nent';
4699 18787 +    label  NextClosest = 'R-Sqaure with Next Cluster
      Component';
4700 18788 +    rename OwnCluster = RSqWithClusterComp;
4701 18789 +    rename NextClosest = RSqWithNextClusComp;
4702 18790 +run;
4703 18791 +%end;
4704 18792 +-----------------------------------------------------
     -----------*/
4705 18793 +
4706 18794 +%if %sysfunc(exist(&inClusRSquare)) %then %do;
4707 18795 +/* to calculate NextClosestClusRsq */
4708 18796 +proc transpose data = &inClusRSquare  out=_clusRsq;
4709 18797 +      by cluster;
4710 18798 +      run;
4711 18799 +data _clusRsq;
4712 18800 +     set _clusRsq;
4713 18801 +     if strip(upcase(Cluster)) eq strip(upcase(_NAME
     _)) then delete;
4714 18802 +run;
```

```
4715 18803 +
4716 18804 +proc sort data=_clusRsq;
4717 18805 +    by cluster col1;
4718 18806 +    run;
4719 18807 +data _clusRsq(drop=_NAME_ _LABEL_);
4720 18808 +     set _clusRsq; by cluster;
4721 18809 +     if last.Cluster then output;
4722 18810 +     label  COL1 = 'R-Sqaure with Next Cluster Compo
     nent';
4723 18811 +     rename COL1 = RSqWithNextClusComp;
4724 18812 +     rename Cluster = Variable;
4725 18813 +     label  Cluster = "Variable";
4726 18814 +run;
4727 18815 +
4728 18816 +proc sort data =&outdata;
4729 18817 +     by Variable;
4730 18818 +run;
4731 18819 +data &outdata;
4732 18820 +    merge &outdata _clusRsq;
4733 18821 +    by Variable;
4734 18822 +run;
4735 18823 +proc sort data =&outdata;
4736 18824 +by Cluster  RsquareRatio;
4737 18825 +run;
4738 18826 +quit;
4739 18827 +%end;
4740 18828 +
4741 18829 +proc datasets lib = work nolist;
4742 18830 +     delete _tmprsq _clusRsq;
4743 18831 +     run;
4744 18832 +quit;
4745 18833 +
4746 18834 +%mend MakeRSquareData;
4747 18835 +
4748 18836 +
4749 18837 +/*-------------------------------------------------
```

```
     ------------------------*/
4750 18838 +
4751 18839 +
4752 18840 +
4753 18841 +%macro ModifyCorr(indata=,
4754 18842 +                  outdata=,
4755 18843 +                  rsquare = Y
4756 18844 +                  );
4757 18845 +    data corr_tmp;
4758 18846 +        set &indata;
4759 18847 +    run;
4760 18848 +     proc sql;
4761 18849 +           update &indata
4762 18850 +           set
4763 18851 +     %let dsid = %sysfunc(open(work.corr_tmp));
4764 18852 +     %let nvar = %sysfunc(attrn(&dsid, NVAR));
4765 18853 +          %do i = 4 %to &nvar;
4766 18854 +          %let _name = %sysfunc(varname(&dsid, &i));
4767 18855 +             %if &rsquare eq Y %then %let _name_md =
      &_name.**2;
4768 18856 +            %else  %let _name_md = &_name;
4769 18857 +            %if &i < &nvar %then %do;
4770 18858 +               &_name = 1- &_name_md ,
4771 18859 +            %end;
4772 18860 +            %else %do;
4773 18861 +               &_name = &_name_md where _TYPE_ conta
     ins 'CORR' ;
4774 18862 +            %end;
4775 18863 +          %end;
4776 18864 +      %let dsid= %sysfunc(close(&dsid));
4777 18865 +
4778 18866 +      select * from &indata;
4779 18867 +      run;
4780 18868 +      proc datasets lib = work nolist;
4781 18869 +           delete corr_tmp;
4782 18870 +      run;
```

```
4783 18871 +      quit;
4784 18872 +
4785 18873 + %mend ModifyCorr;
4786 18874 +
4787 18875 + %macro MakeClusStructCorrData(indata=,outdata=, ncl
     uster=, Rsquare=N);
4788 18876 +    data &outdata(drop= _NCL_  _TYPE_);
4789 18877 +       set &indata;
4790 18878 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
     E_) eq 'STRUCTUR') then delete;
4791 18879 +       rename _NAME_ = Cluster;
4792 18880 +       label _NAME_ = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_clustername,  noquote))";
4793 18881 +    run;
4794 18882 +    %if &RSquare eq Y %then %do;
4795 18883 +     data corr_tmp;
4796 18884 +        set &outdata;
4797 18885 +     run;
4798 18886 +
4799 18887 +     data &outdata(drop=i);
4800 18888 +          set &outdata;
4801 18889 +          %let dsid = %sysfunc(open(work.corr_tmp));
4802 18890 +          %let nvar = %sysfunc(attrn(&dsid, NVAR));
4803 18891 +          %do i = 2 %to &nvar;
4804 18892 +            %let _name = %sysfunc(varname(&dsid, &i)
     );
4805 18893 +            %let _name_md = &_name.**2;
4806 18894 +                &_name = &_name_md;
4807 18895 +          %end;
4808 18896 +      %let dsid= %sysfunc(close(&dsid));
4809 18897 +      run;
4810 18898 +      proc datasets lib = work nolist;
4811 18899 +           delete corr_tmp;
4812 18900 +      run;
4813 18901 +
4814 18902 +    %end;
```

```
4815 18903 +     quit;
4816 18904 +%mend MakeClusStructCorrData;
4817 18905 +
4818 18906 +%macro MakeInterClusCorrData(indata=, outdata=, nclu
     ster=, RSquare=N, makeplotds=N, plotds=);
4819 18907 +    data &outdata(drop= _NCL_  _TYPE_);
4820 18908 +       set &indata;
4821 18909 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
     E_) eq 'CCORR') then delete;
4822 18910 +       rename _NAME_ = Cluster;
4823 18911 +       label _NAME_ = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_clustername,  noquote))";
4824 18912 +    run;
4825 18913 +    data corr_tmp;
4826 18914 +        set &outdata;
4827 18915 +    run;
4828 18916 +
4829 18917 +    %let dsid = %sysfunc(open(work.corr_tmp));
4830 18918 +    %let nclus2= %eval(&ncluster+1);
4831 18919 +    data &outdata;
4832 18920 +         set &outdata;
4833 18921 +          %do i = 2 %to &nclus2;
4834 18922 +          %let i_1 = %eval(&i-1);
4835 18923 +            %let _name = %sysfunc(varname(&dsid, &i)
     );
4836 18924 +            %let _newName = Clus&i_1;
4837 18925 +                rename &_name = &_newName; ;
4838 18926 +                *label &_name ="Cluster &i_1";
4839 18927 +                label &_name = "%sysfunc(sasmsg(sash
     elp.dmine, rpt_varclus_label_clusternum, noquote,  &i_1))";
4840 18928 +          %end;
4841 18929 +          keep Cluster
4842 18930 +          %do i = 2 %to &nclus2;
4843 18931 +                %let _name = %sysfunc(varname(&dsid,
      &i));
4844 18932 +                &_name
```

```
4845  18933 +          %end;
4846  18934 +          ;
4847  18935 +    %let dsid= %sysfunc(close(&dsid));
4848  18936 +     run;
4849  18937 +     quit;
4850  18938 +
4851  18939 +    %if &RSquare eq Y %then %do;
4852  18940 +
4853  18941 +       data corr_tmp;
4854  18942 +        set &outdata;
4855  18943 +     run;
4856  18944 +
4857  18945 +     data &outdata(drop=i);
4858  18946 +          set &outdata;
4859  18947 +          %let dsid = %sysfunc(open(work.corr_tmp));
4860  18948 +          %let nvar = %sysfunc(attrn(&dsid, NVAR));
4861  18949 +          %do i = 2 %to &nvar;
4862  18950 +            %let _name = %sysfunc(varname(&dsid, &i)
     );
4863  18951 +            %let _name_md = &_name.**2;
4864  18952 +                &_name = &_name_md;
4865  18953 +          %end;
4866  18954 +      %let dsid= %sysfunc(close(&dsid));
4867  18955 +      run;
4868  18956 +    %end;
4869  18957 +
4870  18958 +    %if &makeplotds eq Y %then %do;
4871  18959 +     proc transpose data = &outdata
4872  18960 +          out=&plotds(drop=_LABEL_ rename=(_NAME_ =
     Y Cluster=X Col1= Correlation));
4873  18961 +          by cluster;
4874  18962 +     run;
4875  18963 +     data &plotds;
4876  18964 +          set  &plotds;
4877  18965 +          label x="%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_label_cluster, noquote))";
```

```
4878 18966 +          label Y="%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_label_cluster, noquote))";
4879 18967 +     run;
4880 18968 +     %end;
4881 18969 +     proc datasets lib = work nolist;
4882 18970 +           delete corr_tmp;
4883 18971 +     run;
4884 18972 +     quit;
4885 18973 +%mend MakeInterClusCorrData;
4886 18974 +
4887 18975 +
4888 18976 +%macro MakeClusConstellData(indata=, outlink=, outno
     de=);
4889 18977 +
4890 18978 +data &outlink(drop = Selected);
4891 18979 +     set &indata;
4892 18980 +     LINKID = _N_;
4893 18981 +     label LINKID = "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_linkid,  noquote))";
4894 18982 +     if strip(upcase(Cluster)) eq strip(upcase(Varia
     ble)) then Variable = ClosestCluster;
4895 18983 +run;
4896 18984 +data &outnode(keep=NODEID TYPE LABEL);
4897 18985 +    set &indata;
4898 18986 +    length TYPE $16;
4899 18987 +    rename Variable = NODEID;
4900 18988 +    label  Variable= "%sysfunc(sasmsg(sashelp.dmine,
      rpt_varclus_label_nodeidvar, noquote))";
4901 18989 +    if strip(upcase(Cluster)) eq strip(upcase(Variab
     le))
4902 18990 +    then  TYPE = "CLUSTER";
4903 18991 +    else  TYPE="VARIABLE";
4904 18992 +    label TYPE = "%sysfunc(sasmsg(sashelp.dmine, rpt
     _varclus_label_nodetype, noquote))";
4905 18993 +run;
4906 18994 +quit;
```

```
4907 18995 +%mend MakeClusConstellData;
4908 18996 +
4909 18997 +
4910 18998 +
4911 18999 +%macro MakeClusConstellData(indata=, outlink=, outno
     de=);
4912 19000 +
4913 19001 +data &outlink(drop = Selected);
4914 19002 +     set &indata;
4915 19003 +     LINKID = _N_;
4916 19004 +     label LINKID = "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_linkid,  noquote))";
4917 19005 +     if strip(upcase(Cluster)) eq strip(upcase(Varia
     ble)) then Variable = ClosestCluster;
4918 19006 +run;
4919 19007 +data &outnode(keep=NODEID TYPE LABEL);
4920 19008 +    set &indata;
4921 19009 +    length TYPE $16;
4922 19010 +    rename Variable = NODEID;
4923 19011 +    label  Variable= "%sysfunc(sasmsg(sashelp.dmine,
      rpt_varclus_label_nodeidvar, noquote))";
4924 19012 +    if strip(upcase(Cluster)) eq strip(upcase(Variab
     le))
4925 19013 +    then  TYPE = "CLUSTER";
4926 19014 +    else  TYPE="VARIABLE";
4927 19015 +    label TYPE = "%sysfunc(sasmsg(sashelp.dmine, rpt
     _varclus_label_nodetype, noquote))";
4928 19016 +run;
4929 19017 +quit;
4930 19018 +%mend MakeClusConstellData;
4931 19019 +
4932 19020 +
4933 19021 +/*--- This will work only when inds is not a view da
     ta -------
4934 19022 +
4935 19023 +%macro getNVarNObs(inds=, nvar=, nobs=);
```

```
4936 19024 +
4937 19025 +    %global &nvar;
4938 19026 +    %global &nobs;
4939 19027 +    data _null_;
4940 19028 +         dsid = open("&inds");
4941 19029 +         nv = attrn(dsid, 'NVAR');
4942 19030 +         no = attrn(dsid, 'NOBS');
4943 19031 +         dsid = close(dsid);
4944 19032 +         call symput("&nvar", nv);
4945 19033 +         call symput("&nobs", no);
4946 19034 +    run;
4947 19035 +    quit;
4948 19036 +%mend  getNVarNObs;
4949 19037 +
4950 19038 ++-------------------------------------------------
     ------------*/
4951 19039 +
4952 19040 +
4953 19041 +%macro getNVar(inds=, nvar=);
4954 19042 +    %global &nvar;
4955 19043 +    data _null_;
4956 19044 +         dsid = open("&inds");
4957 19045 +         nv = attrn(dsid, 'NVAR');
4958 19046 +         dsid = close(dsid);
4959 19047 +         call symput("&nvar", nv);
4960 19048 +    run;
4961 19049 +    quit;
4962 19050 +%mend  getNVar;
4963 19051 +
4964 19052 +
4965 19053 +
4966 19054 +%macro getNObs(inds=, nobs=);
4967 19055 +    %global &nobs;
4968 19056 +    data _null_;
4969 19057 +        set &inds end=eof;
4970 19058 +        if eof then call symput("&nobs", _N_);
```

```
4971  19059 +    run;
4972  19060 +    quit;
4973  19061 +%mend  getNObs;
4974  19062 +
4975  19063 +%Macro CreateVarclusMeta(trainnum=);
4976  19064 +   %EM_GETNAME(KEY=VARCLUSMETA, TYPE=DATA);
4977  19065 +    data &EM_USER_VARCLUSMETA;
4978  19066 +         length TrainNum 8.;
4979  19067 +         length NewTrain $8;
4980  19068 +         length NGCluster 8.;
4981  19069 +         length ExportedComp $16;
4982  19070 +         length HideVariable $8;
4983  19071 +         TrainNum = &trainnum;
4984  19072 +         NewTrain = "Y";
4985  19073 +         ExportedComp = "&EM_PROPERTY_EXPORTEDCOMP";
4986  19074 +         HideVariable = "&EM_PROPERTY_HIDEVARIABLE";
4987  19075 +         NGCluster = 0; /* zero means no twostage */
4988  19076 +   run;
4989  19077 +   quit;
4990  19078 +%mend CreateVarclusMeta;
4991  NOTE: %INCLUDE (level 1) ending.
4992  NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMEXPL.V
      ARIABLECLUSTERING_MACROS2.SOURCE.
4993  19079 +
4994  19080 +%macro MakeInterClusCorrData(indata=, outdata=, nclu
      ster=, globalclusid=, RSquare=N, makeplotds=N, plotds=);
4995  19081 +    data &outdata(drop= _NCL_  _TYPE_);
4996  19082 +       set &indata;
4997  19083 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
      E_) eq 'CCORR') then delete;
4998  19084 +       %if &globalclusid ne %then %do;
4999  19085 +       _NAME_ = "GC&globalclusid._"||upcase(_NAME_);
5000  19086 +       rename _NAME_ = Cluster;
5001  19087 +       %end;
5002  19088 +       %else %do;
5003  19089 +        _NAME_ = upcase(_NAME_);
```

```
5004 19090 +       rename _NAME_ = Cluster;
5005 19091 +       %end;
5006 19092 +       label _NAME_ = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_clustername,  noquote))";
5007 19093 +    run;
5008 19094 +    data corr_tmp;
5009 19095 +        set &outdata;
5010 19096 +    run;
5011 19097 +
5012 19098 +    %let dsid = %sysfunc(open(work.corr_tmp));
5013 19099 +    %let nclus2= %eval(&ncluster+1);
5014 19100 +    data &outdata;
5015 19101 +         set &outdata;
5016 19102 +          %do i = 2 %to &nclus2;
5017 19103 +          %let i_1 = %eval(&i-1);
5018 19104 +            %let _name = %sysfunc(varname(&dsid, &i)
     );
5019 19105 +            %if &globalclusid ne %then
5020 19106 +                %do; %let _newName = GC&globalclusid
     ._CLUS&i_1;
5021 19107 +                     rename &_name = &_newName;
5022 19108 +                     *label &_name ="GC &globalclusi
     d : Cluster &i_1";
5023 19109 +                     label &_name = "%sysfunc(sasmsg
     (sashelp.dmine, rpt_varclus_label_gc_clusternum, noquote,
     &globalclusid, &i_1))";
5024 19110 +                %end;
5025 19111 +            %else
5026 19112 +                %do; %let _newName = CLUS&i_1;
5027 19113 +                     rename &_name = &_newName;
5028 19114 +                     *label &_name ="Cluster &i_1";
5029 19115 +                     label &_name = "%sysfunc(sasmsg
     (sashelp.dmine, rpt_varclus_label_clusternum, noquote,  &i_
     1))";
5030 19116 +                %end;
5031 19117 +          %end;
```

```
5032  19118 +          keep Cluster
5033  19119 +          %do i = 2 %to &nclus2;
5034  19120 +                %let _name = %sysfunc(varname(&dsid,
      &i));
5035  19121 +                &_name
5036  19122 +          %end;
5037  19123 +          ;
5038  19124 +     %let dsid= %sysfunc(close(&dsid));
5039  19125 +      run;
5040  19126 +      quit;
5041  19127 +
5042  19128 +     %if &RSquare eq Y %then %do;
5043  19129 +
5044  19130 +       data corr_tmp;
5045  19131 +        set &outdata;
5046  19132 +      run;
5047  19133 +
5048  19134 +      data &outdata;
5049  19135 +          set &outdata;
5050  19136 +          %let dsid = %sysfunc(open(work.corr_tmp));
5051  19137 +          %let nvar = %sysfunc(attrn(&dsid, NVAR));
5052  19138 +          %do i = 2 %to &nvar;
5053  19139 +            %let _name = %sysfunc(varname(&dsid, &i)
     );
5054  19140 +            %let _name_md = &_name.**2;
5055  19141 +                &_name = &_name_md;
5056  19142 +          %end;
5057  19143 +       %let dsid= %sysfunc(close(&dsid));
5058  19144 +       run;
5059  19145 +     %end;
5060  19146 +
5061  19147 +     %if &makeplotds eq Y %then %do;
5062  19148 +      proc transpose data = &outdata
5063  19149 +          out=&plotds(drop=_LABEL_ rename=(_NAME_ =
     Y Cluster=X Col1= Correlation));
5064  19150 +          by cluster;
```

```
5065 19151 +     run;
5066 19152 +     data &plotds;
5067 19153 +          set  &plotds;
5068 19154 +          label x="%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_label_cluster, noquote))";
5069 19155 +          label Y="%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_label_cluster, noquote))";
5070 19156 +          label  Correlation="%sysfunc(sasmsg(sashel
     p.dmine, rpt_correlation_vlabel, noquote))";
5071 19157 +     run;
5072 19158 +     %end;
5073 19159 +     proc datasets lib = work nolist;
5074 19160 +           delete corr_tmp;
5075 19161 +     run;
5076 19162 +     quit;
5077 19163 +%mend MakeInterClusCorrData;
5078 19164 +
5079 19165 +%macro MakeOwnRSquare(indata=, outdata=, ncluster=,
     globalclusid=);
5080 19166 +    data _tmpds(drop= _NCL_);
5081 19167 +       set &indata;
5082 19168 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
     E_) in ('GROUP','RSQUARED')) then delete;
5083 19169 +       %if &globalclusid ne %then %do;
5084 19170 +           _NAME_ = "GC&globalclusid.";
5085 19171 +       rename _NAME_ = Cluster;
5086 19172 +       %end;
5087 19173 +       %else %do;
5088 19174 +        _NAME_ = "CLUS";
5089 19175 +        rename _NAME_ = Cluster;
5090 19176 +       %end;
5091 19177 +       label _NAME_ = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_clustername,  noquote))";
5092 19178 +    run;
5093 19179 +    proc transpose data =_tmpds out =&outdata;
5094 19180 +    run;
```

```
5095  19181 +
5096  19182 +    data &outdata(drop=COL1);
5097  19183 +        %if &globalclusid ne %then %do;
5098  19184 +        length GCluster $16;
5099  19185 +        %end;
5100  19186 +        length Cluster $32;
5101  19187 +        length _NAME_ $32;
5102  19188 +         set &outdata;
5103  19189 +         _NAME_ = upcase(_NAME_);
5104  19190 +         rename _NAME_=Variable;
5105  19191 +         *label _NAME_="Variable";
5106  19192 +         label _NAME_ = "%sysfunc(sasmsg(sashelp.dmi
      ne, rpt_varclus_label_variable, noquote))" ;
5107  19193 +         label Cluster = "%sysfunc(sasmsg(sashelp.dm
      ine, rpt_varclus_label_cluster, noquote))" ;
5108  19194 +         label GCluster = "%sysfunc(sasmsg(sashelp.d
      mine, rpt_varclus_label_gcluster, noquote))" ;
5109  19195 +
5110  19196 +         %if &globalclusid ne %then %do;
5111  19197 +          GCluster = "GC&globalclusid";
5112  19198 +          Cluster = "GC&globalclusid._CLUS"||strip(C
      OL1);
5113  19199 +         %end;
5114  19200 +         %else %do;
5115  19201 +         Cluster = "CLUS"||strip(COL1);
5116  19202 +        %end;
5117  19203 +         rename COL2 = RSqWithOwnClusComp;
5118  19204 +         *label COL2 = "R-Square With Own Cluster Co
      mponent";
5119  19205 +         label COL2 = "%sysfunc(sasmsg(sashelp.dmine
      , rpt_varclus_label_ownrsq, noquote))" ;
5120  19206 +
5121  19207 +   run;
5122  19208 +   proc sort data =&outdata;
5123  19209 +        by Cluster RSqWithOwnClusComp;
5124  19210 +   run;
```

```
5125 19211 +   proc datasets lib = work nolist;
5126 19212 +           delete _tmpds;
5127 19213 +   run;
5128 19214 +   quit;
5129 19215 +%mend MakeOwnRSquare;
5130 19216 +
5131 19217 +%macro MakeClusStructCorrData(indata=, outdata=, glo
     balclusid=, ncluster=, Rsquare=N);
5132 19218 +    data &outdata(drop= _NCL_  _TYPE_);
5133 19219 +       %if &globalclusid ne %then %do;
5134 19220 +        length GCluster $16;
5135 19221 +        %end;
5136 19222 +       set &indata;
5137 19223 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
     E_) eq 'STRUCTUR') then delete;
5138 19224 +       %if &globalclusid ne %then %do;
5139 19225 +         GCluster = "GC&globalclusid";
5140 19226 +         _NAME_ = "GC&globalclusid._"||upcase(_NAME_
     );
5141 19227 +         rename _NAME_ = Cluster;
5142 19228 +         label _NAME_ = "%sysfunc(sasmsg(sashelp.dmi
     ne, rpt_varclus_label_cluster, noquote))" ;
5143 19229 +         label GCluster = "%sysfunc(sasmsg(sashelp.d
     mine, rpt_varclus_label_gcluster, noquote))" ;
5144 19230 +
5145 19231 +       %end;
5146 19232 +       %else %do;
5147 19233 +         _NAME_ = upcase(_NAME_);
5148 19234 +         rename _NAME_ = Cluster;
5149 19235 +         label _NAME_ = "%sysfunc(sasmsg(sashelp.dmi
     ne, rpt_varclus_label_cluster, noquote))" ;
5150 19236 +         label GCluster = "%sysfunc(sasmsg(sashelp.d
     mine, rpt_varclus_label_gcluster, noquote))" ;
5151 19237 +
5152 19238 +       %end;
5153 19239 +     run;
```

```
5154 19240 +    %if &RSquare eq Y %then %do;
5155 19241 +     data corr_tmp;
5156 19242 +          set &outdata;
5157 19243 +     run;
5158 19244 +     %let istart = 2;
5159 19245 +     %if &globalclusid ne %then %let istart = 3;
5160 19246 +     data &outdata;
5161 19247 +          set &outdata;
5162 19248 +          %let dsid = %sysfunc(open(work.corr_tmp));
5163 19249 +          %let nvar = %sysfunc(attrn(&dsid, NVAR));
5164 19250 +          %do i =&istart %to &nvar;
5165 19251 +            %let _name = %sysfunc(varname(&dsid, &i)
     );
5166 19252 +            %let _name_md = &_name.**2;
5167 19253 +                &_name = &_name_md;
5168 19254 +          %end;
5169 19255 +      %let dsid= %sysfunc(close(&dsid));
5170 19256 +      run;
5171 19257 +      proc datasets lib = work nolist;
5172 19258 +           delete corr_tmp;
5173 19259 +      run;
5174 19260 +    %end;
5175 19261 +     quit;
5176 19262 +%mend MakeClusStructCorrData;
5177 19263 +
5178 19264 +/*
5179 19265 +%MakeClusStructCorrData(indata=playpen._outstat, out
     data=_structrsq , ncluster=7, Rsquare=Y);
5180 19266 +*/
5181 19267 +
5182 19268 +%macro FindNextClosestClusByVar(indata=, outdata=, g
     lobalclusid=, ncluster=);
5183 19269 +
5184 19270 +     /* The indata should be the outdata
5185 19271 +        from %MakeClusStructCorrData(indata=, outdat
     a=, ); */
```

```
5186  19272 +
5187  19273 +     proc sort data =&indata out=_tmpclusRsq;
5188  19274 +     by cluster;
5189  19275 +     run;
5190  19276 +
5191  19277 +     proc transpose data =_tmpclusRsq out=_tmpclusRs
      q;
5192  19278 +      by cluster;
5193  19279 +     run;
5194  19280 +
5195  19281 +     proc sort data=_tmpclusRsq;
5196  19282 +        by _NAME_ COL1;
5197  19283 +     run;
5198  19284 +
5199  19285 +     data _tmpclusRsq;
5200  19286 +         length _NAME_ $32;
5201  19287 +         set _tmpclusRsq; by _NAME_;
5202  19288 +          _NAME_ = upcase(_NAME_);
5203  19289 +          %if &ncluster ne 1 %then %do;
5204  19290 +             if last._NAME_ then delete;
5205  19291 +          %end;
5206  19292 +          %else %do;
5207  19293 +             COL1 = 0;
5208  19294 +          %end;
5209  19295 +     run;
5210  19296 +     /* need to sort again */
5211  19297 +     proc sort data=_tmpclusRsq;
5212  19298 +        by _NAME_ COL1;
5213  19299 +     run;
5214  19300 +
5215  19301 +     data &outdata;
5216  19302 +         set _tmpclusRsq; by _NAME_;
5217  19303 +         Cluster = upcase(Cluster);
5218  19304 +         if last._NAME_ then output;
5219  19305 +         *label  COL1 = 'R-Sqaure with Next Cluster
      Component';
```

```
5220 19306 +         label COL1 = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_nextrsq, noquote))" ;
5221 19307 +         rename COL1 = RSqWithNextClusComp;
5222 19308 +         Cluster = upcase(Cluster);
5223 19309 +         rename Cluster = ClosestCluster;
5224 19310 +         *label  Cluster = "Next Closest Cluster";
5225 19311 +         label Cluster = "%sysfunc(sasmsg(sashelp.dm
     ine, rpt_varclus_label_nextclus, noquote))" ;
5226 19312 +         rename _NAME_ = Variable;
5227 19313 +         label  _NAME_ = "%sysfunc(sasmsg(sashelp.dm
     ine, rpt_varclus_label_variable, noquote))" ;
5228 19314 +    run;
5229 19315 +
5230 19316 +
5231 19317 +    %if &globalclusid ne %then %do;
5232 19318 +    data &outdata;
5233 19319 +       length GCluster $16;
5234 19320 +       set &outdata;
5235 19321 +       GCluster = "GC&globalclusid";
5236 19322 +       run;
5237 19323 +    %end;
5238 19324 +    proc datasets lib = work nolist;
5239 19325 +           delete _tmpclusRsq;
5240 19326 +    run;
5241 19327 +    quit;
5242 19328 +%mend FindNextClosestClusByVar;
5243 19329 +
5244 19330 +
5245 19331 +%macro FindNextClosestClusByCluster(indata=, outdata
     =, globalclusid=, ncluster=);
5246 19332 +     /* The indata should be the outdata from %MakeI
     nterClusCorrData(indata=, outdata=, ); */
5247 19333 +     proc sort data =&indata out=_tmpclusRsq;
5248 19334 +     by cluster;
5249 19335 +     run;
5250 19336 +     proc transpose data =_tmpclusRsq out=_tmpclusRs
```

```
     q;
5251 19337 +     by cluster;
5252 19338 +     run;
5253 19339 +     proc sort data=_tmpclusRsq;
5254 19340 +        by _NAME_ col1;
5255 19341 +     run;
5256 19342 +     data _tmpclusRsq;
5257 19343 +         length _NAME_ $32;
5258 19344 +         set _tmpclusRsq; by _NAME_;
5259 19345 +          _NAME_ = upcase(_NAME_);
5260 19346 +          %if &ncluster ne 1 %then %do;
5261 19347 +             if last._NAME_ then delete;
5262 19348 +          %end;
5263 19349 +          %else %do;
5264 19350 +             COL1 = 0;
5265 19351 +          %end;
5266 19352 +     run;
5267 19353 +     data &outdata;
5268 19354 +         set _tmpclusRsq; by _NAME_;
5269 19355 +         Cluster = upcase(Cluster);
5270 19356 +         if last._NAME_ then output;
5271 19357 +         *label  COL1 = 'R-Sqaure with Next Cluster
     Component';
5272 19358 +         label COL1 = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_nextrsq, noquote))" ;
5273 19359 +         rename COL1 = RSqWithNextClusComp;
5274 19360 +         Cluster = upcase(Cluster);
5275 19361 +         rename Cluster = ClosestCluster;
5276 19362 +         *label  Cluster = "Next Closest Cluster";
5277 19363 +         label Cluster = "%sysfunc(sasmsg(sashelp.dm
     ine, rpt_varclus_label_nextclus, noquote))" ;
5278 19364 +         rename _NAME_ = Variable;
5279 19365 +         *label  _NAME_ = "Variable";
5280 19366 +         label  _NAME_ = "%sysfunc(sasmsg(sashelp.dm
     ine, rpt_varclus_label_variable, noquote))" ;
5281 19367 +
```

```
5282 19368 +    run;
5283 19369 +    %if &globalclusid ne %then %do;
5284 19370 +    data &outdata;
5285 19371 +       length GCluster $16;
5286 19372 +       set &outdata;
5287 19373 +       GCluster = "GC&globalclusid";
5288 19374 +       run;
5289 19375 +    %end;
5290 19376 +
5291 19377 +    proc datasets lib = work nolist;
5292 19378 +           delete _tmpclusRsq;
5293 19379 +    run;
5294 19380 +
5295 19381 +    quit;
5296 19382 +%mend FindNextClosestClusByCluster;
5297 19383 +
5298 19384 +%macro MakeVarClusResultTable(indata1=, indata2=, in
     data3=, outdata=, globalclusid=, ncluster=, selectedcomp=cl
     ustercomp);
5299 19385 +/*----
5300 19386 +  indata1=_ownRsq, indata2=_nextVarRsq, indata3=_nex
     tClusRSq,
5301 19387 +-----------*/
5302 19388 +
5303 19389 +proc sort data =&indata1;
5304 19390 +    by Variable;
5305 19391 +run;
5306 19392 +proc sort data =&indata2;
5307 19393 +    by Variable;
5308 19394 +run;
5309 19395 +data &outdata;
5310 19396 +    merge &indata1 &indata2;
5311 19397 +    by Variable;
5312 19398 +    length Type $16;
5313 19399 +    Type = 'Variable';
5314 19400 +    *label Type ='Type';
```

```
5315 19401 +    label Type = "%sysfunc(sasmsg(sashelp.dmine, rpt
     _varclus_label_type, noquote))" ;
5316 19402 +run;
5317 19403 +
5318 19404 +
5319 19405 +data &indata3;
5320 19406 +    set  &indata3 ;
5321 19407 +    length RSqWithOwnClusComp 8.;
5322 19408 +    Cluster = Variable;
5323 19409 +    RSqWithOwnClusComp = 1;
5324 19410 +    *label RSqWithOwnClusComp = "R-Square With Own C
     luster Component";
5325 19411 +    label RSqWithOwnClusComp = "%sysfunc(sasmsg(sash
     elp.dmine, rpt_varclus_label_ownrsq, noquote))" ;
5326 19412 +    length Type $16;
5327 19413 +    Type = 'ClusterComp';
5328 19414 +    label Type = "%sysfunc(sasmsg(sashelp.dmine, rpt
     _varclus_label_type, noquote))" ;
5329 19415 +
5330 19416 +;
5331 19417 +run;
5332 19418 +
5333 19419 +proc sort data=&outdata;
5334 19420 +    by Cluster;
5335 19421 +run;
5336 19422 +proc sort data =&indata3;
5337 19423 +    by Cluster;
5338 19424 +run;
5339 19425 +
5340 19426 +data &outdata;
5341 19427 +     set &outdata &indata3;
5342 19428 +     by Cluster;
5343 19429 +run;
5344 19430 +
5345 19431 +
5346 19432 +/* Create the Selected variable with all YES */
```

```
5347 19433 +
5348 19434 +data &outdata;
5349 19435 +     set &outdata;
5350 19436 +     length RsqRatio 8.;
5351 19437 +     length Selected $8;
5352 19438 +     *label RSqRatio = "1-R**2 Ratio";
5353 19439 +     label RSqRatio =  "%sysfunc(sasmsg(sashelp.dmin
     e, rpt_varclus_label_oneminusrsq, noquote))" ;
5354 19440 +     *label Selected = "Variable Selected";
5355 19441 +     label Selected = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_varselected, noquote))" ;
5356 19442 +     RsqRatio = (1-RSqWithOwnClusComp)/(1-RSqWithNex
     tClusComp);
5357 19443 +     Selected ='YES';
5358 19444 +     rename _LABEL_ = Label;
5359 19445 +     label _LABEL_ =  "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_label, noquote))" ;
5360 19446 +run;
5361 19447 +
5362 19448 +
5363 19449 +/*----  Selected = Y/N will be assigened at the %sco
     re                   -------+
5364 19450 +        Just create the Selected variable with all Y
     ES at the step above
5365 19451 + +-----------------------------------------------
     ----------------------------+
5366 19452 +
5367 19453 +proc sort data=&outdata;
5368 19454 +    by Cluster RsqRatio;
5369 19455 +run;
5370 19456 +
5371 19457 +%if &selectedcomp eq CLUSTERCOMP %then %do;
5372 19458 +data &outdata;
5373 19459 +     set &outdata; by Cluster;
5374 19460 +     length Selected $8;
5375 19461 +     label Selected = "Variable Selected";
```

```
5376 19462 +     if  first.Cluster then Selected ='Yes';
5377 19463 +     else Selected = 'No';
5378 19464 +    run;
5379 19465 +%end;
5380 19466 +%else %do;
5381 19467 +data &outdata(drop = _var _varchange);
5382 19468 +     set &outdata; retain _var 0; by Cluster;
5383 19469 +     length Selected $8;
5384 19470 +     label Selected = "Variable Selected";
5385 19471 +     if first.Cluster then  _varchange = 0;
5386 19472 +     else _varchange =1;
5387 19473 +     if _var ne _varchange then  Selected  = 'Yes';
5388 19474 +     else Selected = 'No';
5389 19475 +     if last.cluster then  _var = 0;
5390 19476 +     else _var = _varchange;
5391 19477 +run;
5392 19478 +%end;
5393 19479 +
5394 19480 +-------------------------------------------------
     ----------*/
5395 19481 +
5396 19482 +quit;
5397 19483 +%mend MakeVarClusResultTable;
5398 19484 +
5399 19485 +%Macro MakePlotDataFromCorrTable(indata=, outdata=,
     globalclusid=);
5400 19486 +     proc sort data =&indata;
5401 19487 +        by cluster;
5402 19488 +     run;
5403 19489 +     proc transpose data =&indata
5404 19490 +          out=&outdata(drop=_LABEL_ rename=(_NAME_ =
      Y Cluster=X Col1= Correlation));
5405 19491 +          by cluster;
5406 19492 +     run;
5407 19493 +     data &outdata;
5408 19494 +          set  &outdata;
```

```
5409 19495 +          label x= "%sysfunc(sasmsg(sashelp.dmine, r
     pt_varclus_label_cluster, noquote))" ;
5410 19496 +          label Y= "%sysfunc(sasmsg(sashelp.dmine, r
     pt_varclus_label_cluster, noquote))" ;
5411 19497 +          label Correlation = "%sysfunc(sasmsg(sashe
     lp.dmine, rpt_correlation_vlabel, noquote))" ;
5412 19498 +     run;
5413 19499 +     %if &globalclusid ne %then %do;
5414 19500 +     data &outdata;
5415 19501 +          Length GCluster $16;
5416 19502 +          label GCluster = "%sysfunc(sasmsg(sashelp.
     dmine, rpt_varclus_label_gcluster, noquote))" ;
5417 19503 +          set  &outdata;
5418 19504 +          GCluster ="GC&globalclusid.";
5419 19505 +          run;
5420 19506 +     %end;
5421 19507 +
5422 19508 +%Mend MakePlotDataFromCorrTable;
5423 19509 +
5424 19510 +
5425 19511 +%macro MakeCorrelation(indata=,
5426 19512 +                       outstat=_tmpoutstat,
5427 19513 +                       corrmatrix=N,
5428 19514 +                       outcorr=_tmpoutcorr,
5429 19515 +                       includeclassvar=N,
5430 19516 +                       target=,
5431 19517 +                       freq=,
5432 19518 +                       weight=);
5433 19519 +   %if &target eq  %then %do;
5434 19520 +
5435 19521 +       proc varclus data=&indata outstat=&outstat hi
      maxclusters=1 noprint;
5436 19522 +            var %EM_INTERVAL_INPUT %EM_INTERVAL_REJE
     CTED
5437 19523 +       %if &includeclassvar eq Y %then %do;
5438 19524 +          %let dsid = %sysfunc(open(&EM_USER_OUTDUMM
```

```
     Y));
5439 19525 +          %let nvar = %sysfunc(attrn(&dsid, NVAR));
5440 19526 +          %do i = 2 %to &nvar;
5441 19527 +          %let varname = %sysfunc(varname(&dsid, &i)
     );
5442 19528 +          &varname
5443 19529 +          %end;
5444 19530 +       %end;
5445 19531 +       ;
5446 19532 +      %if &freq ne  %then %do;
5447 19533 +          freq &freq;
5448 19534 +      %end;
5449 19535 +      %if &weight ne  %then %do;
5450 19536 +          weight &weight;
5451 19537 +      %end;
5452 19538 +
5453 19539 +      run;
5454 19540 +      %if &corrmatrix eq Y %then %do;
5455 19541 +      data  &outcorr (drop = _NCL_ _TYPE_);
5456 19542 +           set  &outstat ;
5457 19543 +           if _TYPE_ ='CORR' then output;
5458 19544 +      run;
5459 19545 +      %end;
5460 19546 +    %end;
5461 19547 +    %else %do;
5462 19548 +       proc corr data=&indata outp=&outstat noprint;
5463 19549 +            var
5464 19550 +       %let dsid = %sysfunc(open(&indata));
5465 19551 +       %let nvar = %sysfunc(attrn(&dsid, NVAR));
5466 19552 +          %do i = 1 %to &nvar;
5467 19553 +              %let _name = %sysfunc(varname(&dsid, &
     i));
5468 19554 +              %if &_name ne &target %then;
5469 19555 +              &_name
5470 19556 +           %end;
5471 19557 +        %let dsid= %sysfunc(close(&dsid));
```

```
5472  19558 +        ;
5473  19559 +        with &target;
5474  19560 +        run;
5475  19561 +    %end;
5476  19562 +    quit;
5477  19563 +%mend MakeCorrelation;
5478  19564 +
5479  19565 +
5480  19566 +%macro MakeCorrelationDistance(indata=,
5481  19567 +                                outdata=,
5482  19568 +                                rsquare = N
5483  19569 +                                );
5484  19570 +    data corr_tmp;
5485  19571 +        set &indata;
5486  19572 +        if _N_ = 1 then do;
5487  19573 +           output;
5488  19574 +           stop;
5489  19575 +         end;
5490  19576 +    run;
5491  19577 +    %if &outdata ne  %then %let  _outdata = &outdata
     ;
5492  19578 +    %else %let _outdata = &indata;
5493  19579 +
5494  19580 +    data &_outdata;
5495  19581 +         set &indata;
5496  19582 +
5497  19583 +         %let dsid = %sysfunc(open(work.corr_tmp));
5498  19584 +         %let nvar = %sysfunc(attrn(&dsid, NVAR));
5499  19585 +          %do i = 2 %to &nvar;
5500  19586 +              %let _name = %sysfunc(varname(&dsid, &
     i));
5501  19587 +              %if &rsquare eq Y %then %let _name_md
     = &_name.**2;
5502  19588 +              %else  %let _name_md = &_name;
5503  19589 +              &_name = 1- &_name_md;
5504  19590 +          %end;
```

```
5505  19591 +      %let dsid= %sysfunc(close(&dsid));
5506  19592 +      run;
5507  19593 +      proc datasets lib = work nolist;
5508  19594 +           delete corr_tmp;
5509  19595 +      run;
5510  19596 +      quit;
5511  19597 +  %mend MakeCorrelationDistance;
5512  19598 +
5513  19599 +
5514  19600 +%macro UpdateOutStatCorrToDistance(indata=, /* indat
      a should be a outstat from proc varclus */
5515  19601 +                                    rsquare = N
5516  19602 +                                    );
5517  19603 +    data corr_tmp;
5518  19604 +        set &indata;
5519  19605 +    run;
5520  19606 +    proc sql noprint;
5521  19607 +           update &indata
5522  19608 +           set
5523  19609 +     %let dsid = %sysfunc(open(work.corr_tmp));
5524  19610 +     %let nvar = %sysfunc(attrn(&dsid, NVAR));
5525  19611 +          %do i = 4 %to &nvar;
5526  19612 +          %let _name = %sysfunc(varname(&dsid, &i));
5527  19613 +             %if &rsquare eq Y %then %let _name_md =
       &_name.**2;
5528  19614 +           %else  %let _name_md = &_name;
5529  19615 +           %if &i < &nvar %then %do;
5530  19616 +              &_name = 1- &_name_md ,
5531  19617 +           %end;
5532  19618 +           %else %do;
5533  19619 +              &_name = &_name_md where _TYPE_ eq 'C
      ORR' ;
5534  19620 +           %end;
5535  19621 +          %end;
5536  19622 +      %let dsid= %sysfunc(close(&dsid));
5537  19623 +
```

```
5538  19624 +      select * from &indata;
5539  19625 +      run;
5540  19626 +     data &indata( drop = _NCL_);
5541  19627 +          set &indata;
5542  19628 +          if _TYPE_ not in ('CORR', 'STD', 'N', 'ME
      AN') then delete;
5543  19629 +          if _TYPE_ ='CORR' then _TYPE_ ='DISTANCE'
      ;
5544  19630 +     run;
5545  19631 +     data &indata(DROP = _NCL_);
5546  19632 +          set &indata;
5547  19633 +          if _TYPE_ = 'CORR' then _TYPE_ ='DISTANCE
      ';
5548  19634 +          if _TYPE_ not in ('DISTANCE', 'N', 'STD',
       'MEAN') then delete;
5549  19635 +          rename _NAME_ = _VAR_;
5550  19636 +     run;
5551  19637 +     proc datasets lib = work nolist;
5552  19638 +          delete corr_tmp;
5553  19639 +     run;
5554  19640 +     quit;
5555  19641 + %mend UpdateOutStatCorrToDistance;
5556  19642 +
5557  19643 +
5558  19644 +%macro HierClusWithCorr(indata= ,
5559  19645 +                         ncluster=,
5560  19646 +                         method = Ward,
5561  19647 +                         outtree = _outtree,
5562  19648 +                         idvar =_VAR_,
5563  19649 +                         outdata=,
5564  19650 +                         rescore = N,
5565  19651 +                         newncluster=
5566  19652 +                         );
5567  19653 +      %global &newncluster;
5568  19654 +      %if &rescore ne Y %then %do;
5569  19655 +      proc cluster data=&indata(type=Distance where=
```

```
      (upcase(strip(_TYPE_)) = "DISTANCE"))
5570  19656 +                   method=&method outtree=&outtree n
      oprint;
5571  19657 +            id &idvar;
5572  19658 +       run;
5573  19659 +       %end;
5574  19660 +       proc tree data=&outtree nclusters = &ncluster
      out=&outdata noprint;
5575  19661 +       run;
5576  19662 +       /* ----- Check some variables like CL1, CL5...
      , remove them ----*/
5577  19663 +       proc contents data =&indata out=_outcontent(ke
      ep=NAME) noprint;
5578  19664 +       run;
5579  19665 +       data _outcontent;
5580  19666 +           set _outcontent;
5581  19667 +           if NAME in ('_TYPE_' , '_VAR_') then delet
      e;
5582  19668 +           index = 1;
5583  19669 +           rename NAME = _NAME_;
5584  19670 +       run;
5585  19671 +       proc sort data=_outcontent;
5586  19672 +           by _NAME_;
5587  19673 +       run;
5588  19674 +       proc sort data =&outdata;
5589  19675 +            by _NAME_;
5590  19676 +       run;
5591  19677 +       data &outdata(drop=index);
5592  19678 +           merge &outdata _outcontent;
5593  19679 +           by _NAME_;
5594  19680 +           if index = . then delete;
5595  19681 +       run;
5596  19682 +       /*-----------------------------------------
      ---------------*/
5597  19683 +       data &outdata;
5598  19684 +            length CLUSNAME $16;
```

```
5599  19685 +           set &outdata;
5600  19686 +           if CLUSTER > &ncluster then delete;
5601  19687 +           CLUSNAME='GC'||strip(CLUSTER);
5602  19688 +           *label CLUSNAME = "Cluster Name";
5603  19689 +           label CLUSNAME  = "%sysfunc(sasmsg(sashel
      p.dmine, rpt_varclus_label_clustername, noquote))" ;
5604  19690 +           rename _NAME_ = VARIABLE ;
5605  19691 +           *label _NAME_ = "Variable";
5606  19692 +           *label CLUSTER = "Cluster";
5607  19693 +           label _NAME_ ="%sysfunc(sasmsg(sashelp.dm
      ine, rpt_varclus_label_variable, noquote))" ;
5608  19694 +           label CLUSTER ="%sysfunc(sasmsg(sashelp.d
      mine, rpt_varclus_label_cluster, noquote))" ;
5609  19695 +      run;
5610  19696 +      proc sort data=&outdata out=&outdata;
5611  19697 +           by CLUSTER;
5612  19698 +      run;
5613  19699 +      proc means data =&outdata noprint;
5614  19700 +           output out=_meanout;
5615  19701 +      run;
5616  19702 +      data _null_;
5617  19703 +            set _meanout;
5618  19704 +            if strip(_STAT_) eq 'MAX' then do;
5619  19705 +            call symput("&newncluster", CLUSTER);
5620  19706 +            stop;
5621  19707 +            end;
5622  19708 +      run;
5623  19709 +
5624  19710 +      proc datasets lib = work nolist;
5625  19711 +           delete _outcontent _meanout;
5626  19712 +      run;
5627  19713 +      quit;
5628  19714 +%mend  HierClusWithCorr;
5629  19715 +
5630  19716 +%macro CreateScoreCode2(indata=, ncluscomp=, globalc
      lusid=, fileref=);
```

```
5631 19717 +
5632 19718 +     data _tmpindata;
5633 19719 +          set &indata;
5634 19720 +          if (_TYPE_ in ('SCORE' 'MEAN' 'STD') and _
     NCL_ = &ncluscomp ) or (_TYPE_ in ('MEAN' 'STD'));
5635 19721 +          if _TYPE_ = 'MEAN' then _NAME_='MEAN';
5636 19722 +          if _TYPE_ = 'STD' then _NAME_='STD';
5637 19723 +          if _TYPE_ = 'SCORE' then _NAME_=upcase("GC
     &globalclusid._"||_NAME_);
5638 19724 +          DROP _TYPE_ _NCL_;
5639 19725 +     run;
5640 19726 +
5641 19727 +     filename _file_  "&fileRef";
5642 19728 +
5643 19729 +     data _null_;
5644 19730 +        FILE _file_ MOD;
5645 19731 +        put ' ';
5646 19732 +        put "/*-----------------------------------
     ------------*/";
5647 19733 +        put '/* ' "%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_score_title_gclus, noquote, &globalclusid))" '*/'
     ;
5648 19734 +        put "/*-----------------------------------
     ------------*/";
5649 19735 +        put ' ';
5650 19736 +        %let dsid = %sysfunc(open(work._tmpindata));
5651 19737 +
5652 19738 +        %let nvar = %sysfunc(attrn(&dsid, NVAR));
5653 19739 +        %let vn_name =%sysfunc(varnum(&dsid, _NAME_)
     );
5654 19740 +        %let k = 1;
5655 19741 +        %do %while(^%sysfunc(fetch(&dsid)));
5656 19742 +              %let _name = %sysfunc(getvarc(&dsid,
     &vn_name));
5657 19743 +              %if &k > 2 %then %do;
5658 19744 +               %let cn = %eval(&k-2);
```

```
5659 19745 +                 put "&_name = 0 ; /*---" "%sysfunc(
     sasmsg(sashelp.dmine, rpt_varclus_score_gcluscompnum, noquo
     te, &globalclusid, &cn))" "------ */";
5660 19746 +               %end;
5661 19747 +               %let k = %eval(&k+1);
5662 19748 +
5663 19749 +        %end;
5664 19750 +        %let rc = %sysfunc(rewind(&dsid));
5665 19751 +
5666 19752 +        %do j= 2 %to &nvar;
5667 19753 +            %let _varname = %sysfunc(varname(&dsid,
     &j));
5668 19754 +            %do %while(^%sysfunc(fetch(&dsid)));
5669 19755 +               %let _name = %sysfunc(getvarc(&dsid,
      &vn_name));
5670 19756 +               %if &_name = MEAN %then
5671 19757 +               %let _mean = %sysfunc(getvarn(&dsid,
      &j));
5672 19758 +               %else %if &_name = STD %then
5673 19759 +               %let _std = %sysfunc(getvarn(&dsid,
     &j));
5674 19760 +               %else %do;
5675 19761 +                     %let coeff =  %sysfunc(getvarn
     (&dsid, &j));
5676 19762 +                     %let abscoeff = %sysfunc(abs(&
     coeff));
5677 19763 +                        %if &abscoeff >  0 %then %
     do;
5678 19764 +                         put "&_name = &_name+&coeff *
      (&_varname - &_mean)/&_std;";
5679 19765 +                          %end;
5680 19766 +                %end;
5681 19767 +             %end;
5682 19768 +             %let rc = %sysfunc(rewind(&dsid));
5683 19769 +         %end;
5684 19770 +
```

```
5685  19771 +        %let dsid= %sysfunc(close(&dsid));
5686  19772 +
5687  19773 +     run;
5688  19774 +
5689  19775 +     filename _file_;
5690  19776 +     proc datasets lib = work nolist;
5691  19777 +           delete _tmpindata;
5692  19778 +     run;
5693  19779 +     quit;
5694  19780 +%mend CreateScoreCode2;
5695  19781 +
5696  19782 +
5697  19783 +%macro MakeDeltaCode2(groupds=,deltacodefile=);
5698  19784 +
5699  19785 +     /*--- Build Code to Modify Metadata ---*/
5700  19786 +     filename X "&deltacodefile";
5701  19787 +     data _null_;
5702  19788 +        FILE X;
5703  19789 +        set &groupds end=eof;
5704  19790 +        if _N_=1 then do;
5705  19791 +           %if &EM_PROPERTY_INCLUDECLASSVAR eq Y %th
     en %do;
5706  19792 +            put "if upcase(strip(ROLE)) ='INPUT' and
      upcase(strip(LEVEL)) ^='INTERVAL' then ROLE ='REJECTED' ;"
     ;
5707  19793 +           %end;
5708  19794 +           put "if upcase(strip(ROLE))='INPUT' and u
     pcase(strip(LEVEL))='INTERVAL' then do;";
5709  19795 +           put "if upcase(strip(NAME)) in (";
5710  19796 +        end;
5711  19797 +        if Strip(upcase(Selected)) eq 'YES' then do;
5712  19798 +           string = '"'!!trim(left(VARIABLE))!!'"';
5713  19799 +           put string;
5714  19800 +        end;
5715  19801 +        if eof then do;
5716  19802 +           put ') then ROLE="INPUT";';
```

```
5717 19803 +           put 'else ROLE="REJECTED";';
5718 19804 +           put 'end;';
5719 19805 +
5720 19806 +           %if %upcase(&EM_PROPERTY_HIDEVARIABLE) eq
      Y %then %do;
5721 19807 +             put 'if upcase(strip(ROLE)) = "REJECTED
     " then delete ;';
5722 19808 +           %end;
5723 19809 +        end;
5724 19810 +     run;
5725 19811 +     quit;
5726 19812 +
5727 19813 +     filename X;
5728 19814 +     quit;
5729 19815 +%mend MakeDeltaCode2;
5730 19816 +
5731 19817 +%macro getInitialGClusterNumber(indata=, ninput=, nd
     ummy=0, div=100, ngc=);
5732 19818 +  %global &ngc;
5733 19819 +  data _null_;
5734 19820 +  %if &indata ne %then %do;
5735 19821 +      %let dsid = %sysfunc(open(&indata));
5736 19822 +           %let nvar = %sysfunc(attrn(&dsid, NVAR));
5737 19823 +      %let dsid = %sysfunc(close(&dsid));
5738 19824 +  %end;
5739 19825 +  %else %do;
5740 19826 +       %let nvar = %eval(&ninput+&ndummy); ;
5741 19827 +  %end;
5742 19828 +  %let numgc = %eval(&nvar/&div+2);
5743 19829 +  %let &ngc = &numgc;
5744 19830 +   run;
5745 19831 +  quit;
5746 19832 +%mend getInitialGClusterNumber;
5747 19833 +
5748 19834 +
5749 19835 +%macro MakeGobalConstellData(indata=, outlink=, outn
```

```
     ode=);
5750 19836 +data &outlink(drop = Selected);
5751 19837 +     set &indata;
5752 19838 +     LINKID = _N_;
5753 19839 +     label LINKID = "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_linkid, noquote))" ;
5754 19840 +run;
5755 19841 +data &outnode(keep=NODEID TYPE LABEL);
5756 19842 +    set &indata;
5757 19843 +    length TYPE $16;
5758 19844 +    rename VARIABLE = NODEID;
5759 19845 +    *label  CLUSNAME="Node ID";
5760 19846 +    label CLUSNAME= "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_nodeidvar, noquote))" ;
5761 19847 +    TYPE = "VARIABLE";
5762 19848 +    *label TYPE = "Node Type";
5763 19849 +    label TYPE =  "%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_label_nodetype, noquote))" ;
5764 19850 +    run;
5765 19851 +data _tmp(keep=NODEID TYPE LABEL);
5766 19852 +    set &indata;
5767 19853 +    length TYPE $16;
5768 19854 +    rename CLUSNAME = NODEID;
5769 19855 +    label  CLUSNAME= "%sysfunc(sasmsg(sashelp.dmine,
      rpt_varclus_label_nodeidvar, noquote))" ;
5770 19856 +    TYPE = "GCLUSTER";
5771 19857 +    label TYPE = "%sysfunc(sasmsg(sashelp.dmine, rpt
     _varclus_label_nodetype, noquote))" ;
5772 19858 +    run;
5773 19859 +proc sort data=_tmp;
5774 19860 +    by NODEID;
5775 19861 +run;
5776 19862 +data _tmp;
5777 19863 +    set _tmp; by NODEID;
5778 19864 +    if first.NODEID then output;
5779 19865 +run;
```

```
5780 19866 +proc sort data=&outnode;
5781 19867 +    by NODEID;
5782 19868 +run;
5783 19869 +data  &outnode;
5784 19870 +    set _tmp &outnode;
5785 19871 +run;
5786 19872 +proc datasets lib = work nolist;
5787 19873 +     delete _tmp;
5788 19874 +run;
5789 19875 +quit;
5790 19876 +%mend MakeGobalConstellData;
5791 19877 +
5792 19878 +/* Make contellation plot data among GCLUSTERS */
5793 19879 +
5794 19880 +%Macro MakeGClusterConstData(indata=, inoutrsq=, out
     node=, outlink=);
5795 19881 +
5796 19882 +data &outlink(keep = _NAME_ _PARENT_ _LABEL_ LINKID)
     ;
5797 19883 +     set &indata;
5798 19884 +     LINKID = _N_;
5799 19885 +     if upcase(substr(strip(_NAME_),1, 2))="CL" then
      do;
5800 19886 +        _NAME_ = "ROOT"||upcase(substr(strip(_NAME_)
     ,5));
5801 19887 +     end;
5802 19888 +     if _PARENT_ ne " " and upcase(substr(strip(_PAR
     ENT_),1, 2))="CL" then do;
5803 19889 +        _PARENT_ = "ROOT"||upcase(substr(strip(_PARE
     NT_),5));
5804 19890 +     end;
5805 19891 +     if upcase(substr(strip(_LABEL_),1, 2))="CL" the
     n do;
5806 19892 +        _LABEL_ = "ROOT"||upcase(substr(strip(_LABEL
     _),5));
5807 19893 +    end;
```

```
5808 19894 +run;
5809 19895 +
5810 19896 +data _tmp_outrsquare;
5811 19897 +        set &inoutrsq;
5812 19898 +        if upcase(strip(TYPE)) = 'CLUSTERCOMP' then
     delete;
5813 19899 +run;
5814 19900 +
5815 19901 +proc freq data =_tmp_outrsquare noprint;
5816 19902 +         tables GCluster/out=_tmp_GCLUSFREQ(rename=(
     GCLUSTER=_NAME_));
5817 19903 +run;
5818 19904 +
5819 19905 +data &outnode(keep=_NAME_ TYPE LABEL);
5820 19906 +    set &outlink;
5821 19907 +    length TYPE $16;
5822 19908 +    length LABEL $100;
5823 19909 +    /*label CLUSNAME= "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_nodeidvar, noquote))" ;*/
5824 19910 +
5825 19911 +    if upcase(substr(strip(_NAME_),1, 2))='GC' then
     do;
5826 19912 +    TYPE = "GCLUSTER";
5827 19913 +    LABEL = "%sysfunc(sasmsg(sashelp.dmine, rpt_varc
     lus_label_gcluster, noquote)):"||_NAME_;
5828 19914 +    end;
5829 19915 +    else do;
5830 19916 +    TYPE= "ROOT";
5831 19917 +    LABEL= _NAME_;
5832 19918 +    end;
5833 19919 +    label TYPE  =  "%sysfunc(sasmsg(sashelp.dmine, r
     pt_varclus_label_nodetype, noquote))" ;
5834 19920 +    label LABEL =  "%sysfunc(sasmsg(sashelp.dmine, r
     pt_varclus_label_label, noquote))" ;
5835 19921 +    label _NAME_ =  "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_nodeidvar, noquote))" ;
```

```
5836 19922 +    run;
5837 19923 +
5838 19924 +proc sort data=&outnode;
5839 19925 +     by _NAME_;
5840 19926 +proc sort data=_tmp_GCLUSFREQ;
5841 19927 +     by _NAME_;
5842 19928 +run;
5843 19929 +
5844 19930 +data  &outnode;
5845 19931 +     merge &outnode _tmp_GCLUSFREQ; by _NAME_;
5846 19932 +     if COUNT=. then COUNT=1;
5847 19933 +run;
5848 19934 +
5849 19935 +proc datasets lib = work nolist;
5850 19936 +     delete _tmp_outrsquare _tmp_GCLUSFREQ;
5851 19937 +run;
5852 19938 +
5853 19939 +quit;
5854 19940 +%Mend MakeGClusterConstData;
5855 19941 +
5856 19942 +
5857 19943 +%macro CreateGClusterScoreCode(indata=,  globalclusi
     d=, fileref=);
5858 19944 +
5859 19945 +    data _gscoretmpds;
5860 19946 +          set &indata;
5861 19947 +          if (_TYPE_ in ('SCORE' 'MEAN' 'STD') and _
     NCL_ = 1 ) or (_TYPE_ in ('MEAN' 'STD'));
5862 19948 +          if _TYPE_ = 'MEAN' then _NAME_='MEAN';
5863 19949 +          if _TYPE_ = 'STD' then _NAME_='STD';
5864 19950 +          if _TYPE_ = 'SCORE' then _NAME_ = "GC"||st
     rip(&globalclusid);
5865 19951 +          DROP _TYPE_ _NCL_;
5866 19952 +     run;
5867 19953 +
5868 19954 +    /*  %let gscorefile =  %bquote(&EM_NODEDIR)&EM_D
```

```
     SEP.gclusterscore.sas;
5869 19955 +        GCluster Component &globalclusid ------ */
5870 19956 +
5871 19957 +    filename _file_  "&fileref";
5872 19958 +
5873 19959 +    data _null_;
5874 19960 +        %if &globalclusid eq 1 %then %do;
5875 19961 +          FILE _file_;
5876 19962 +        %end;
5877 19963 +        %else %do;
5878 19964 +          FILE _file_ MOD;
5879 19965 +        %end;
5880 19966 +
5881 19967 +        %let dsid = %sysfunc(open(work._gscoretmpds)
     );
5882 19968 +        %let nvar = %sysfunc(attrn(&dsid, NVAR));
5883 19969 +        %let vn_name =%sysfunc(varnum(&dsid, _NAME_)
     );
5884 19970 +
5885 19971 +        %let k = 1;
5886 19972 +        %do %while(^%sysfunc(fetch(&dsid)));
5887 19973 +                %let _name = %sysfunc(getvarc(&dsid,
      &vn_name));
5888 19974 +                %if &k > 2 %then %do;
5889 19975 +                put "&_name = 0 ; ";
5890 19976 +                %end;
5891 19977 +                %let k = %eval(&k+1);
5892 19978 +        %end;
5893 19979 +
5894 19980 +        %let rc = %sysfunc(rewind(&dsid));
5895 19981 +        %do i= 2 %to &nvar;
5896 19982 +            %let _varname =  %sysfunc(varname(&dsid,
      &i));
5897 19983 +            %do %while(^%sysfunc(fetch(&dsid)));
5898 19984 +                %let _name = %sysfunc(getvarc(&dsid,
      &vn_name));
```

```
5899  19985 +                %if &_name = MEAN %then
5900  19986 +                %let _mean = %sysfunc(getvarn(&dsid,
       &i));
5901  19987 +                %else %if &_name = STD %then
5902  19988 +                %let _std = %sysfunc(getvarn(&dsid,
      &i));
5903  19989 +                %else %do;
5904  19990 +                      %let coeff =  %sysfunc(getvarn
      (&dsid, &i));
5905  19991 +                      %let abscoeff = %sysfunc(abs(&
      coeff));
5906  19992 +                          %if &abscoeff >  0 %then %
      do;
5907  19993 +                       put "&_name = &_name+&coeff *
       (&_varname - &_mean)/&_std;";
5908  19994 +                           %end;
5909  19995 +                 %end;
5910  19996 +             %end;
5911  19997 +             %let rc = %sysfunc(rewind(&dsid));
5912  19998 +
5913  19999 +         %end;
5914  20000 +
5915  20001 +        %let dsid= %sysfunc(close(&dsid));
5916  20002 +       run;
5917  20003 +
5918  20004 +
5919  20005 +       proc datasets lib=work nolist;
5920  20006 +            delete _gscoretmpds;
5921  20007 +       run;
5922  20008 +      quit;
5923  20009 +
5924  20010 +%mend CreateGClusterScoreCode;
5925  20011 +
5926  20012 +
5927  20013 +%macro MakeGClusterCorrelation(Indata=, ngcluster=,
      gscorecode=, outrsquare=);
```

```
5928  20014 +
5929  20015 +   %EM_REGISTER(KEY=GSCORE, TYPE=DATA);
5930  20016 +   %EM_GETNAME(KEY=GSCORE, TYPE=DATA);
5931  20017 +   %EM_REGISTER(KEY=GSCORESTAT, TYPE=DATA);
5932  20018 +   %EM_GETNAME(KEY=GSCORESTAT, TYPE=DATA);
5933  20019 +   %EM_REGISTER(KEY=GSCORETREE, TYPE=DATA);
5934  20020 +   %EM_GETNAME(KEY=GSCORETREE, TYPE=DATA);
5935  20021 +   %EM_REGISTER(KEY=GSCORECORR, TYPE=DATA);
5936  20022 +   %EM_GETNAME(KEY=GSCORECORR, TYPE=DATA);
5937  20023 +   %EM_REGISTER(KEY=GSCORECORRPLOT, TYPE=DATA);
5938  20024 +   %EM_GETNAME(KEY=GSCORECORRPLOT, TYPE=DATA);
5939  20025 +   %EM_REGISTER(KEY=GCLUSLINK, TYPE=DATA);
5940  20026 +   %EM_GETNAME(KEY=GCLUSLINK, TYPE=DATA);
5941  20027 +   %EM_REGISTER(KEY=GCLUSNODE, TYPE=DATA);
5942  20028 +   %EM_GETNAME(KEY=GCLUSNODE, TYPE=DATA);
5943  20029 +
5944  20030 +   filename gsfile "&gscorecode";
5945  20031 +
5946  20032 +   data &EM_USER_GSCORE;
5947  20033 +             set &indata;
5948  20034 +             %include  gsfile;
5949  20035 +          keep
5950  20036 +          %do i=1 %to &ngcluster;
5951  20037 +           %let gcvarname = GC&i;
5952  20038 +           &gcvarname
5953  20039 +          %end;
5954  20040 +     ;
5955  20041 +   run;
5956  20042 +
5957  20043 +   proc varclus data=&EM_USER_GSCORE outstat=&EM_USE
      R_GSCORESTAT outtree=&EM_USER_GSCORETREE
5958  20044 +    %if %upcase(&EM_PROPERTY_CLUSCOMP) eq CENTROID %
      then %do; centroid %end;
5959  20045 +    %if %upcase(&EM_PROPERTY_CLUSSOURCE) eq COV %the
      n %do; cov %end;
5960  20046 +    %if %upcase(&EM_PROPERTY_CLUSHIERACHY) eq Y %the
```

```
     n %do; hi %end;
5961 20047 +    noprint  ;
5962 20048 +     var
5963 20049 +       %do i=1 %to &ngcluster;
5964 20050 +         %let gcvarname = GC&i;
5965 20051 +         &gcvarname
5966 20052 +        %end;
5967 20053 +   ;
5968 20054 +   run;
5969 20055 +
5970 20056 +
5971 20057 +   %MakeVarClusCorrData(statds=&EM_USER_GSCORESTAT,
     corrds=&EM_USER_GSCORECORR, corrplotds=&EM_USER_GSCORECORRP
     LOT );
5972 20058 +   data &EM_USER_GSCORECORRPLOT ;
5973 20059 +        set &EM_USER_GSCORECORRPLOT;
5974 20060 +        rename _X_ = X;
5975 20061 +        rename _Y_ = Y;
5976 20062 +        label _X_ = "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_gcluster, noquote))" ;
5977 20063 +        label _Y_ = "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_gcluster, noquote))" ;
5978 20064 +   run;
5979 20065 +
5980 20066 +   %MakeGClusterConstData(indata=&EM_USER_GSCORETREE
     , inoutrsq=&outrsquare, outnode=&EM_USER_GCLUSNODE, outlink
     =&EM_USER_GCLUSLINK);
5981 20067 +
5982 20068 +   data &EM_USER_GSCORETREE;
5983 20069 +        length _NAME_ $32;
5984 20070 +        length _LABEL_ $100;
5985 20071 +        set &EM_USER_GSCORETREE(DROP=_LABEL_);
5986 20072 +        if upcase(substr(strip(_NAME_),1, 2))='GC' t
     hen do;
5987 20073 +         _LABEL_ = "%sysfunc(sasmsg(sashelp.dmine, r
     pt_varclus_label_gcluster, noquote)):"||_NAME_;
```

```
5988  20074 +        end;else do;
5989  20075 +         _LABEL_ = _NAME_;
5990  20076 +        end;
5991  20077 +       label _LABEL_ = "%sysfunc(sasmsg(sashelp.dmin
      e, rpt_varclus_label_variable, noquote))";
5992  20078 +
5993  20079 +   run;
5994  20080 +
5995  20081 +   quit;
5996  20082 +
5997  20083 +%mend MakeGClusterCorrelation;
5998  20084 +
5999  20085 +
6000  NOTE: %INCLUDE (level 1) ending.
6001  NOTE: Fileref TEMP has been deassigned.
6002  NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMEXPL.V
      ARIABLECLUSTERING_SCORE.SOURCE.
6003  20086 +%macro score;
6004  20087 +
6005  20088 +filename temp catalog 'sashelp.emexpl.variablecluste
      ring_score_macros.source';
6006  20089 +%include temp;
6007  20090 +filename temp;
6008  20091 +
6009  20092 +%EM_GETNAME(key=VARCLUSMETA, type=DATA) ;
6010  20093 +
6011  20094 +data _null_;
6012  20095 +         set &EM_USER_VARCLUSMETA;
6013  20096 +         if _N_=1 then
6014  20097 +         call symput('_trainnum', TrainNum);
6015  20098 +         call symput('_exportedcomp', ExportedComp);
6016  20099 +         call symput('_hidevariable', HideVariable);
6017  20100 +         call symput('_newTrain', NewTrain);
6018  20101 +run;
6019  20102 +
6020  20103 +%if &_newTrain = N %then %do;
```

```
6021 20104 +        %if &EM_PROPERTY_EXPORTEDCOMP eq &_exportedc
     omp %then %let _exportedCompChanged = N;
6022 20105 +        %else %let _exportedCompChanged = Y;
6023 20106 +        %if &EM_PROPERTY_HIDEVARIABLE eq &_hidevaria
     ble %then %let _hideVariableChanged = N;
6024 20107 +        %else  %let _hideVariableChanged = Y;
6025 20108 +%end;
6026 20109 +%else %do;
6027 20110 +        %let _exportedCompChanged = Y;
6028 20111 +        %let _hideVariableChanged = Y;
6029 20112 +%end;
6030 20113 +
6031 20114 +%if (&_trainnum = 1 ) or %upcase(&EM_PROPERTY_TWOSTA
     GECLUS) = NO %then %do;
6032 20115 +     filename temp catalog 'sashelp.emexpl.variablec
     lustering_score1.source';
6033 20116 +     %include temp;
6034 20117 +     filename temp;
6035 20118 +     %score1(ExportedCompChanged=&_exportedCompChang
     ed, HideVariableChanged=&_hideVariableChanged);
6036 20119 +%end;
6037 20120 +%if (&_trainnum = 2 ) or %upcase(&EM_PROPERTY_TWOSTA
     GECLUS) = YES %then %do;
6038 20121 +     filename temp catalog 'sashelp.emexpl.variablec
     lustering_score2.source';
6039 20122 +     %include temp;
6040 20123 +     filename temp;
6041 20124 +     %score2(ExportedCompChanged=&_exportedCompChang
     ed, HideVariableChanged=&_hideVariableChanged);
6042 20125 +%end;
6043 20126 +
6044 20127 +  /* store  current property values */
6045 20128 + data &EM_USER_VARCLUSMETA;
6046 20129 +       set &EM_USER_VARCLUSMETA;
6047 20130 +        ExportedComp = "&EM_PROPERTY_EXPORTEDCOMP";
6048 20131 +        HideVariable = "&EM_PROPERTY_HIDEVARIABLE";
```

```
6049  20132 + run;
6050  20133 +
6051  20134 +%mend score;
6052  20135 +
6053  20136 +
6054  NOTE: %INCLUDE (level 1) ending.
6055  NOTE: Fileref TEMP has been deassigned.
6056  NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMEXPL.V
      ARIABLECLUSTERING_SCORE_MACROS.SOURCE.
6057  20137 +
6058  20138 +%macro ChangeVariableSelectionFlag(Indata=, gcluster
      =);
6059  20139 +     proc sort data =&indata;
6060  20140 +               by Cluster RsqRatio Type;
6061  20141 +     run;
6062  20142 +     /*-----------------------------------------------
      -------------------------
6063  20143 +      The Type variabe is used becuse the variable n
      eed to be selected
6064  20144 +        instead of Cluster component when only one v
      ariable is in the cluster
6065  20145 +     +-----------------------------------------------
      -------------------------*/
6066  20146 +
6067  20147 +     %if &EM_PROPERTY_EXPORTEDCOMP eq CLUSTERCOMP %t
      hen %do;
6068  20148 +         data &indata;
6069  20149 +              set &indata; by cluster;
6070  20150 +              if first.Cluster then Selected = 'YES'
      ;
6071  20151 +              else Selected = 'NO';
6072  20152 +         run;
6073  20153 +         quit;
6074  20154 +    %end;
6075  20155 +    %else %do;
6076  20156 +         data &indata(drop = _var _varchange);
```

```
6077  20157 +              set &indata; retain _var 0; by Cluster
      ;
6078  20158 +             if first.Cluster then  _varchange = 0;
6079  20159 +                 else _varchange =1;
6080  20160 +             if _var ne _varchange then  Selected  =
       'YES';
6081  20161 +             else Selected = 'NO';
6082  20162 +             if last.cluster then  _var = 0;
6083  20163 +             else _var = _varchange;
6084  20164 +         run;
6085  20165 +         quit;
6086  20166 +    %end;
6087  20167 +%mend ChangeVariableSelectionFlag;
6088  20168 +
6089  NOTE: %INCLUDE (level 1) ending.
6090  NOTE: Fileref TEMP has been deassigned.
6091
6092  NOTE: Numeric values have been converted to character value
      s at the places given by: (Line):(Column).
6093        3:15
6094  NOTE: There were 1 observations read from the data set EMWS
      4.VARCLUS2_VARCLUSMETA.
6095  NOTE: DATA statement used (Total process time):
6096        real time           0.00 seconds
6097        user cpu time       0.00 seconds
6098        system cpu time     0.00 seconds
6099        memory              160464.71k
6100        OS Memory           170880.00k
6101        Timestamp           07/01/2024 05:54:48 AM
6102        Step Count                        1  Switch Count  0
6103        Page Faults                       0
6104        Page Reclaims                     62
6105        Page Swaps                        0
6106        Voluntary Context Switches        9
6107        Involuntary Context Switches      0
6108        Block Input Operations            288
```

```
6109        Block Output Operations           0
6110
6111
6112  NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMEXPL.V
      ARIABLECLUSTERING_SCORE1.SOURCE.
6113  20169 +
6114  20170 +%macro score1(ExportedCompChanged=,
6115  20171 +              HideVariableChanged=);
6116  20172 +
6117  20173 +   %EM_GETNAME(KEY=OUTRSQUARE, TYPE=DATA);
6118  20174 +   %if &ExportedCompChanged = Y %then %do;
6119  20175 +   %ChangeVariableSelectionFlag(Indata=&EM_USER_OUTR
      SQUARE);
6120  20176 +   %end;
6121  20177 +   %MakeDeltaCode(groupds=&EM_USER_OUTRSQUARE, Delta
      CodeFile=&EM_FILE_CDELTA_TRAIN);
6122  20178 +
6123  20179 +%mend score1;
6124  20180 +
6125  NOTE: %INCLUDE (level 1) ending.
6126  NOTE: Fileref TEMP has been deassigned.
6127
6128  NOTE: There were 89 observations read from the data set EMW
      S4.VARCLUS2_OUTRSQUARE.
6129  NOTE: The data set EMWS4.VARCLUS2_OUTRSQUARE has 89 observa
      tions and 9 variables.
6130  NOTE: PROCEDURE SORT used (Total process time):
6131        real time           0.01 seconds
6132        user cpu time       0.00 seconds
6133        system cpu time     0.00 seconds
6134        memory              160464.71k
6135        OS Memory           170880.00k
6136        Timestamp           07/01/2024 05:54:48 AM
6137        Step Count                        1  Switch Count  0
6138        Page Faults                       0
6139        Page Reclaims                     117
```

```
6140        Page Swaps                          0
6141        Voluntary Context Switches          33
6142        Involuntary Context Switches        0
6143        Block Input Operations              0
6144        Block Output Operations             264
6145
6146
6147
6148  NOTE: There were 89 observations read from the data set EMW
      S4.VARCLUS2_OUTRSQUARE.
6149  NOTE: The data set EMWS4.VARCLUS2_OUTRSQUARE has 89 observa
      tions and 9 variables.
6150  NOTE: DATA statement used (Total process time):
6151        real time           0.01 seconds
6152        user cpu time       0.01 seconds
6153        system cpu time     0.00 seconds
6154        memory              160464.71k
6155        OS Memory           170880.00k
6156        Timestamp           07/01/2024 05:54:48 AM
6157        Step Count                          1  Switch Count  0
6158        Page Faults                         0
6159        Page Reclaims                       126
6160        Page Swaps                          0
6161        Voluntary Context Switches          37
6162        Involuntary Context Switches        0
6163        Block Input Operations              288
6164        Block Output Operations             264
6165
6166
6167
6168  NOTE: The file X is:
6169        Filename=/home/u63452984/case-study-s2192852/Workspac
      es/EMWS4/VarClus2/CDELTA_TRAIN.sas,
6170        Owner Name=u63452984,Group Name=oda,
6171        Access Permission=-rw-r--r--,
6172        Last Modified=07 January 2024 05:54:48
```

```
6173
6174 NOTE: 35 records were written to the file X.
6175       The minimum record length was 4.
6176       The maximum record length was 93.
6177 NOTE: There were 89 observations read from the data set EMW
     S4.VARCLUS2_OUTRSQUARE.
6178 NOTE: DATA statement used (Total process time):
6179       real time           0.00 seconds
6180       user cpu time       0.00 seconds
6181       system cpu time     0.00 seconds
6182       memory              160464.71k
6183       OS Memory           170880.00k
6184       Timestamp           07/01/2024 05:54:48 AM
6185       Step Count                        1  Switch Count  0
6186       Page Faults                       0
6187       Page Reclaims                     67
6188       Page Swaps                        0
6189       Voluntary Context Switches        14
6190       Involuntary Context Switches      0
6191       Block Input Operations            288
6192       Block Output Operations           8
6193
6194
6195 NOTE: Fileref X has been deassigned.
6196
6197 NOTE: There were 1 observations read from the data set EMWS
     4.VARCLUS2_VARCLUSMETA.
6198 NOTE: The data set EMWS4.VARCLUS2_VARCLUSMETA has 1 observa
     tions and 5 variables.
6199 NOTE: DATA statement used (Total process time):
6200       real time           0.01 seconds
6201       user cpu time       0.00 seconds
6202       system cpu time     0.00 seconds
6203       memory              160464.71k
6204       OS Memory           170880.00k
6205       Timestamp           07/01/2024 05:54:48 AM
```

```
6206        Step Count                         1  Switch Count  0
6207        Page Faults                        0
6208        Page Reclaims                      128
6209        Page Swaps                         0
6210        Voluntary Context Switches         33
6211        Involuntary Context Switches       0
6212        Block Input Operations             0
6213        Block Output Operations            264
6214
6215
6216 20181  *-------------------------------------------------
     ---------*;
6217 20182  * End SCORE: VarClus2;
6218 20183  *-------------------------------------------------
     ---------*;
6219 20184
6220
6221 20185  filename emflow "/home/u63452984/case-study-s2192852
     /Workspaces/EMWS4/VarClus2/EMFLOWSCORE.sas";
6222 20186  *-------------------------------------------------
     ---------*;
6223 20187  * VarClus2: Scoring DATA data;
6224 20188  *-------------------------------------------------
     ---------*;
6225 20189  data EMWS4.VarClus2_TRAIN
6226 20190  / view=EMWS4.VarClus2_TRAIN
6227 20191  ;
6228 20192  set EMWS4.Impt_TRAIN
6229 20193  ;
6230 20194  %inc emflow;
6231 NOTE: %INCLUDE (level 1) file EMFLOW is file /home/u6345298
     4/case-study-s2192852/Workspaces/EMWS4/VarClus2/EMFLOWSCORE
     .sas.
6232 20195 +*** Begin Class Look-up, Standardization, Replacemen
     t ;
6233 20196 +
```

```
6234  20197 +*** Generate dummy variables for Gender ;
6235  20198 +label GenderFemale = 'Gender=Female' ;
6236  20199 +label GenderMale = 'Gender=Male' ;
6237  20200 +if missing( Gender ) then do;
6238  20201 +   GenderFemale = .;
6239  20202 +   GenderMale = .;
6240  20203 +end;
6241  20204 +else do;
6242  20205 +   length _dm6 $ 6; drop _dm6 ;
6243  20206 +   _dm6 = put( Gender , $6. );
6244  20207 +   %DMNORMIP( _dm6 )
6245  20208 +   if _dm6 = 'MALE'  then do;
6246  20209 +      GenderFemale = 0;
6247  20210 +      GenderMale = 1;
6248  20211 +   end;
6249  20212 +   else if _dm6 = 'FEMALE'  then do;
6250  20213 +      GenderFemale = 1;
6251  20214 +      GenderMale = 0;
6252  20215 +   end;
6253  20216 +   else do;
6254  20217 +      delete;
6255  20218 +   end;
6256  20219 +end;
6257  20220 +
6258  20221 +*** Generate dummy variables for IMP_Churn ;
6259  20222 +label IMP_Churn0 = 'IMP_Churn=0' ;
6260  20223 +label IMP_Churn1 = 'IMP_Churn=1' ;
6261  20224 +if missing( IMP_Churn ) then do;
6262  20225 +   IMP_Churn0 = .;
6263  20226 +   IMP_Churn1 = .;
6264  20227 +end;
6265  20228 +else do;
6266  20229 +   length _dm12 $ 12; drop _dm12 ;
6267  20230 +   _dm12 = put( IMP_Churn , BEST12. );
6268  20231 +   %DMNORMIP( _dm12 )
6269  20232 +   if _dm12 = '0'  then do;
```

```
6270 20233 +      IMP_Churn0 = 1;
6271 20234 +      IMP_Churn1 = 0;
6272 20235 +   end;
6273 20236 +   else if _dm12 = '1'  then do;
6274 20237 +      IMP_Churn0 = 0;
6275 20238 +      IMP_Churn1 = 1;
6276 20239 +   end;
6277 20240 +   else do;
6278 20241 +      delete;
6279 20242 +   end;
6280 20243 +end;
6281 20244 +
6282 20245 +*** Generate dummy variables for FavouriteCategory ;
6283 20246 +label FavouriteCategoryBooks = 'FavouriteCategory=Bo
     oks' ;
6284 20247 +label FavouriteCategoryClothing = 'FavouriteCategory
     =Clothing' ;
6285 20248 +label FavouriteCategoryElectronics = 'FavouriteCateg
     ory=Electronics' ;
6286 20249 +label FavouriteCategoryHome = 'FavouriteCategory=Hom
     e' ;
6287 20250 +*** encoding is sparse, initialize to zero;
6288 20251 +FavouriteCategoryBooks = 0;
6289 20252 +FavouriteCategoryClothing = 0;
6290 20253 +FavouriteCategoryElectronics = 0;
6291 20254 +FavouriteCategoryHome = 0;
6292 20255 +if missing( FavouriteCategory ) then do;
6293 20256 +   FavouriteCategoryBooks = .;
6294 20257 +   FavouriteCategoryClothing = .;
6295 20258 +   FavouriteCategoryElectronics = .;
6296 20259 +   FavouriteCategoryHome = .;
6297 20260 +end;
6298 20261 +else do;
6299 20262 +   length _dm11 $ 11; drop _dm11 ;
6300 20263 +   _dm11 = put( FavouriteCategory , $11. );
6301 20264 +   %DMNORMIP( _dm11 )
```

```
6302  20265 +   if _dm11 = 'BOOKS'  then do;
6303  20266 +      FavouriteCategoryBooks = 1;
6304  20267 +   end;
6305  20268 +   else if _dm11 = 'CLOTHING'  then do;
6306  20269 +      FavouriteCategoryClothing = 1;
6307  20270 +   end;
6308  20271 +   else if _dm11 = 'ELECTRONICS'  then do;
6309  20272 +      FavouriteCategoryElectronics = 1;
6310  20273 +   end;
6311  20274 +   else if _dm11 = 'HOME'  then do;
6312  20275 +      FavouriteCategoryHome = 1;
6313  20276 +   end;
6314  20277 +   else do;
6315  20278 +      delete;
6316  20279 +   end;
6317  20280 +end;
6318  20281 +
6319  20282 +*** Generate dummy variables for Location ;
6320  20283 +label LocationJohor = 'Location=Johor' ;
6321  20284 +label LocationKedah = 'Location=Kedah' ;
6322  20285 +label LocationKelantan = 'Location=Kelantan' ;
6323  20286 +label LocationKuala_Lumpur = 'Location=Kuala Lumpur'
       ;
6324  20287 +label LocationMalacca = 'Location=Malacca' ;
6325  20288 +label LocationNegeri_Sembilan = 'Location=Negeri Sem
      bilan' ;
6326  20289 +label LocationPahang = 'Location=Pahang' ;
6327  20290 +label LocationPenang = 'Location=Penang' ;
6328  20291 +label LocationPerak = 'Location=Perak' ;
6329  20292 +label LocationPerlis = 'Location=Perlis' ;
6330  20293 +label LocationPutrajaya = 'Location=Putrajaya' ;
6331  20294 +label LocationSabah = 'Location=Sabah' ;
6332  20295 +label LocationSarawak = 'Location=Sarawak' ;
6333  20296 +label LocationSelangor = 'Location=Selangor' ;
6334  20297 +label LocationTerengganu = 'Location=Terengganu' ;
6335  20298 +*** encoding is sparse, initialize to zero;
```

```
6336  20299 +LocationJohor = 0;
6337  20300 +LocationKedah = 0;
6338  20301 +LocationKelantan = 0;
6339  20302 +LocationKuala_Lumpur = 0;
6340  20303 +LocationMalacca = 0;
6341  20304 +LocationNegeri_Sembilan = 0;
6342  20305 +LocationPahang = 0;
6343  20306 +LocationPenang = 0;
6344  20307 +LocationPerak = 0;
6345  20308 +LocationPerlis = 0;
6346  20309 +LocationPutrajaya = 0;
6347  20310 +LocationSabah = 0;
6348  20311 +LocationSarawak = 0;
6349  20312 +LocationSelangor = 0;
6350  20313 +LocationTerengganu = 0;
6351  20314 +if missing( Location ) then do;
6352  20315 +   LocationJohor = .;
6353  20316 +   LocationKedah = .;
6354  20317 +   LocationKelantan = .;
6355  20318 +   LocationKuala_Lumpur = .;
6356  20319 +   LocationMalacca = .;
6357  20320 +   LocationNegeri_Sembilan = .;
6358  20321 +   LocationPahang = .;
6359  20322 +   LocationPenang = .;
6360  20323 +   LocationPerak = .;
6361  20324 +   LocationPerlis = .;
6362  20325 +   LocationPutrajaya = .;
6363  20326 +   LocationSabah = .;
6364  20327 +   LocationSarawak = .;
6365  20328 +   LocationSelangor = .;
6366  20329 +   LocationTerengganu = .;
6367  20330 +end;
6368  20331 +else do;
6369  20332 +   length _dm15 $ 15; drop _dm15 ;
6370  20333 +   _dm15 = put( Location , $15. );
6371  20334 +   %DMNORMIP( _dm15 )
```

```
6372 20335 +   _dm_find = 0; drop _dm_find;
6373 20336 +   if _dm15 <= 'PENANG'  then do;
6374 20337 +      if _dm15 <= 'KUALA LUMPUR'  then do;
6375 20338 +         if _dm15 <= 'KEDAH'  then do;
6376 20339 +            if _dm15 = 'JOHOR'  then do;
6377 20340 +               LocationJohor = 1;
6378 20341 +               _dm_find = 1;
6379 20342 +            end;
6380 20343 +            else do;
6381 20344 +               if _dm15 = 'KEDAH'  then do;
6382 20345 +                  LocationKedah = 1;
6383 20346 +                  _dm_find = 1;
6384 20347 +               end;
6385 20348 +            end;
6386 20349 +         end;
6387 20350 +         else do;
6388 20351 +            if _dm15 = 'KELANTAN'  then do;
6389 20352 +               LocationKelantan = 1;
6390 20353 +               _dm_find = 1;
6391 20354 +            end;
6392 20355 +            else do;
6393 20356 +               if _dm15 = 'KUALA LUMPUR'  then do;
6394 20357 +                  LocationKuala_Lumpur = 1;
6395 20358 +                  _dm_find = 1;
6396 20359 +               end;
6397 20360 +            end;
6398 20361 +         end;
6399 20362 +      end;
6400 20363 +      else do;
6401 20364 +         if _dm15 <= 'NEGERI SEMBILAN'  then do;
6402 20365 +            if _dm15 = 'MALACCA'  then do;
6403 20366 +               LocationMalacca = 1;
6404 20367 +               _dm_find = 1;
6405 20368 +            end;
6406 20369 +            else do;
6407 20370 +               if _dm15 = 'NEGERI SEMBILAN'  then do
```

```
     ;
6408 20371 +                  LocationNegeri_Sembilan = 1;
6409 20372 +                  _dm_find = 1;
6410 20373 +               end;
6411 20374 +            end;
6412 20375 +         end;
6413 20376 +         else do;
6414 20377 +            if _dm15 = 'PAHANG'  then do;
6415 20378 +               LocationPahang = 1;
6416 20379 +               _dm_find = 1;
6417 20380 +            end;
6418 20381 +            else do;
6419 20382 +               if _dm15 = 'PENANG'  then do;
6420 20383 +                  LocationPenang = 1;
6421 20384 +                  _dm_find = 1;
6422 20385 +               end;
6423 20386 +            end;
6424 20387 +         end;
6425 20388 +      end;
6426 20389 +   end;
6427 20390 +   else do;
6428 20391 +      if _dm15 <= 'SABAH'  then do;
6429 20392 +         if _dm15 <= 'PERLIS'  then do;
6430 20393 +            if _dm15 = 'PERAK'  then do;
6431 20394 +               LocationPerak = 1;
6432 20395 +               _dm_find = 1;
6433 20396 +            end;
6434 20397 +            else do;
6435 20398 +               if _dm15 = 'PERLIS'  then do;
6436 20399 +                  LocationPerlis = 1;
6437 20400 +                  _dm_find = 1;
6438 20401 +               end;
6439 20402 +            end;
6440 20403 +         end;
6441 20404 +         else do;
6442 20405 +            if _dm15 = 'PUTRAJAYA'  then do;
```

```
6443 20406 +               LocationPutrajaya = 1;
6444 20407 +               _dm_find = 1;
6445 20408 +            end;
6446 20409 +            else do;
6447 20410 +               if _dm15 = 'SABAH'  then do;
6448 20411 +                  LocationSabah = 1;
6449 20412 +                  _dm_find = 1;
6450 20413 +               end;
6451 20414 +            end;
6452 20415 +         end;
6453 20416 +      end;
6454 20417 +      else do;
6455 20418 +         if _dm15 <= 'SELANGOR'  then do;
6456 20419 +            if _dm15 = 'SARAWAK'  then do;
6457 20420 +               LocationSarawak = 1;
6458 20421 +               _dm_find = 1;
6459 20422 +            end;
6460 20423 +            else do;
6461 20424 +               if _dm15 = 'SELANGOR'  then do;
6462 20425 +                  LocationSelangor = 1;
6463 20426 +                  _dm_find = 1;
6464 20427 +               end;
6465 20428 +            end;
6466 20429 +         end;
6467 20430 +         else do;
6468 20431 +            if _dm15 = 'TERENGGANU'  then do;
6469 20432 +               LocationTerengganu = 1;
6470 20433 +               _dm_find = 1;
6471 20434 +            end;
6472 20435 +         end;
6473 20436 +      end;
6474 20437 +   end;
6475 20438 +   if not _dm_find then do;
6476 20439 +      delete;
6477 20440 +   end;
6478 20441 +end;
```

```
6479  20442 +
6480  20443 +*** Generate dummy variables for M_Variable ;
6481  20444 +label M_Variable0 = 'M_Variable=0' ;
6482  20445 +label M_Variable1 = 'M_Variable=1' ;
6483  20446 +if missing( M_Variable ) then do;
6484  20447 +   M_Variable0 = .;
6485  20448 +   M_Variable1 = .;
6486  20449 +end;
6487  20450 +else do;
6488  20451 +   length _dm12 $ 12; drop _dm12 ;
6489  20452 +   _dm12 = put( M_Variable , BEST12. );
6490  20453 +   %DMNORMIP( _dm12 )
6491  20454 +   if _dm12 = '0'  then do;
6492  20455 +      M_Variable0 = 1;
6493  20456 +      M_Variable1 = 0;
6494  20457 +   end;
6495  20458 +   else if _dm12 = '1'  then do;
6496  20459 +      M_Variable0 = 0;
6497  20460 +      M_Variable1 = 1;
6498  20461 +   end;
6499  20462 +   else do;
6500  20463 +      delete;
6501  20464 +   end;
6502  20465 +end;
6503  20466 +
6504  20467 +*** Generate dummy variables for MembershipLevel ;
6505  20468 +label MembershipLevelBronze = 'MembershipLevel=Bronz
      e' ;
6506  20469 +label MembershipLevelGold = 'MembershipLevel=Gold' ;
6507  20470 +label MembershipLevelPlatinum = 'MembershipLevel=Pla
      tinum' ;
6508  20471 +label MembershipLevelSilver = 'MembershipLevel=Silve
      r' ;
6509  20472 +*** encoding is sparse, initialize to zero;
6510  20473 +MembershipLevelBronze = 0;
6511  20474 +MembershipLevelGold = 0;
```

```
6512  20475 +MembershipLevelPlatinum = 0;
6513  20476 +MembershipLevelSilver = 0;
6514  20477 +if missing( MembershipLevel ) then do;
6515  20478 +   MembershipLevelBronze = .;
6516  20479 +   MembershipLevelGold = .;
6517  20480 +   MembershipLevelPlatinum = .;
6518  20481 +   MembershipLevelSilver = .;
6519  20482 +end;
6520  20483 +else do;
6521  20484 +   length _dm8 $ 8; drop _dm8 ;
6522  20485 +   _dm8 = put( MembershipLevel , $8. );
6523  20486 +   %DMNORMIP( _dm8 )
6524  20487 +   if _dm8 = 'GOLD'  then do;
6525  20488 +      MembershipLevelGold = 1;
6526  20489 +   end;
6527  20490 +   else if _dm8 = 'PLATINUM'  then do;
6528  20491 +      MembershipLevelPlatinum = 1;
6529  20492 +   end;
6530  20493 +   else if _dm8 = 'SILVER'  then do;
6531  20494 +      MembershipLevelSilver = 1;
6532  20495 +   end;
6533  20496 +   else if _dm8 = 'BRONZE'  then do;
6534  20497 +      MembershipLevelBronze = 1;
6535  20498 +   end;
6536  20499 +   else do;
6537  20500 +      delete;
6538  20501 +   end;
6539  20502 +end;
6540  20503 +
6541  20504 +*** Generate dummy variables for Occupation ;
6542  20505 +label OccupationArchitecture = 'Occupation=Architect
      ure' ;
6543  20506 +label OccupationArts = 'Occupation=Arts' ;
6544  20507 +label OccupationBusiness_Analysis = 'Occupation=Busi
      ness Analysis' ;
6545  20508 +label OccupationConstruction = 'Occupation=Construct
```

```
ion' ;
20509 +label OccupationCustomer_Service = 'Occupation=Custo
mer Service' ;
20510 +label OccupationDentistry = 'Occupation=Dentistry' ;
20511 +label OccupationDesign = 'Occupation=Design' ;
20512 +label OccupationEducation = 'Occupation=Education' ;
20513 +label OccupationEngineering = 'Occupation=Engineerin
g' ;
20514 +label OccupationEntrepreneurship = 'Occupation=Entre
preneurship' ;
20515 +label OccupationEvent_Planning = 'Occupation=Event P
lanning' ;
20516 +label OccupationFinance = 'Occupation=Finance' ;
20517 +label OccupationFitness = 'Occupation=Fitness' ;
20518 +label OccupationHealthcare = 'Occupation=Healthcare'
 ;
20519 +label OccupationHuman_Resources = 'Occupation=Human
Resources' ;
20520 +label OccupationInformation_Techno = 'Occupation=Inf
ormation Technology' ;
20521 +label OccupationJournalism = 'Occupation=Journalism'
 ;
20522 +label OccupationLaw_Enforcement = 'Occupation=Law En
forcement' ;
20523 +label OccupationMarketing = 'Occupation=Marketing' ;
20524 +label OccupationMedicine = 'Occupation=Medicine' ;
20525 +label OccupationPharmacy = 'Occupation=Pharmacy' ;
20526 +label OccupationPhotography = 'Occupation=Photograph
y' ;
20527 +label OccupationPiloting = 'Occupation=Piloting' ;
20528 +label OccupationPlumbing = 'Occupation=Plumbing' ;
20529 +label OccupationReal_Estate = 'Occupation=Real Estat
e' ;
20530 +label OccupationResearch = 'Occupation=Research' ;
20531 +label OccupationSocial_Work = 'Occupation=Social Wor
k' ;
```

```
6569  20532 +label OccupationStudent = 'Occupation=Student' ;
6570  20533 +label OccupationWriting = 'Occupation=Writing' ;
6571  20534 +*** encoding is sparse, initialize to zero;
6572  20535 +OccupationArchitecture = 0;
6573  20536 +OccupationArts = 0;
6574  20537 +OccupationBusiness_Analysis = 0;
6575  20538 +OccupationConstruction = 0;
6576  20539 +OccupationCustomer_Service = 0;
6577  20540 +OccupationDentistry = 0;
6578  20541 +OccupationDesign = 0;
6579  20542 +OccupationEducation = 0;
6580  20543 +OccupationEngineering = 0;
6581  20544 +OccupationEntrepreneurship = 0;
6582  20545 +OccupationEvent_Planning = 0;
6583  20546 +OccupationFinance = 0;
6584  20547 +OccupationFitness = 0;
6585  20548 +OccupationHealthcare = 0;
6586  20549 +OccupationHuman_Resources = 0;
6587  20550 +OccupationInformation_Techno = 0;
6588  20551 +OccupationJournalism = 0;
6589  20552 +OccupationLaw_Enforcement = 0;
6590  20553 +OccupationMarketing = 0;
6591  20554 +OccupationMedicine = 0;
6592  20555 +OccupationPharmacy = 0;
6593  20556 +OccupationPhotography = 0;
6594  20557 +OccupationPiloting = 0;
6595  20558 +OccupationPlumbing = 0;
6596  20559 +OccupationReal_Estate = 0;
6597  20560 +OccupationResearch = 0;
6598  20561 +OccupationSocial_Work = 0;
6599  20562 +OccupationStudent = 0;
6600  20563 +OccupationWriting = 0;
6601  20564 +if missing( Occupation ) then do;
6602  20565 +   OccupationArchitecture = .;
6603  20566 +   OccupationArts = .;
6604  20567 +   OccupationBusiness_Analysis = .;
```

```
6605 20568 +   OccupationConstruction = .;
6606 20569 +   OccupationCustomer_Service = .;
6607 20570 +   OccupationDentistry = .;
6608 20571 +   OccupationDesign = .;
6609 20572 +   OccupationEducation = .;
6610 20573 +   OccupationEngineering = .;
6611 20574 +   OccupationEntrepreneurship = .;
6612 20575 +   OccupationEvent_Planning = .;
6613 20576 +   OccupationFinance = .;
6614 20577 +   OccupationFitness = .;
6615 20578 +   OccupationHealthcare = .;
6616 20579 +   OccupationHuman_Resources = .;
6617 20580 +   OccupationInformation_Techno = .;
6618 20581 +   OccupationJournalism = .;
6619 20582 +   OccupationLaw_Enforcement = .;
6620 20583 +   OccupationMarketing = .;
6621 20584 +   OccupationMedicine = .;
6622 20585 +   OccupationPharmacy = .;
6623 20586 +   OccupationPhotography = .;
6624 20587 +   OccupationPiloting = .;
6625 20588 +   OccupationPlumbing = .;
6626 20589 +   OccupationReal_Estate = .;
6627 20590 +   OccupationResearch = .;
6628 20591 +   OccupationSocial_Work = .;
6629 20592 +   OccupationStudent = .;
6630 20593 +   OccupationWriting = .;
6631 20594 +end;
6632 20595 +else do;
6633 20596 +   length _dm22 $ 22; drop _dm22 ;
6634 20597 +   _dm22 = put( Occupation , $22. );
6635 20598 +   %DMNORMIP( _dm22 )
6636 20599 +   _dm_find = 0; drop _dm_find;
6637 20600 +   if _dm22 <= 'HUMAN RESOURCES'  then do;
6638 20601 +      if _dm22 <= 'EDUCATION'  then do;
6639 20602 +         if _dm22 <= 'CONSTRUCTION'  then do;
6640 20603 +            if _dm22 <= 'ARTS'  then do;
```

```
6641  20604 +              if _dm22 = 'ARCHITECTURE'  then do;
6642  20605 +                 OccupationArchitecture = 1;
6643  20606 +                 _dm_find = 1;
6644  20607 +              end;
6645  20608 +              else do;
6646  20609 +                 if _dm22 = 'ARTS'  then do;
6647  20610 +                    OccupationArts = 1;
6648  20611 +                    _dm_find = 1;
6649  20612 +                 end;
6650  20613 +              end;
6651  20614 +           end;
6652  20615 +           else do;
6653  20616 +              if _dm22 = 'BUSINESS ANALYSIS'  then
      do;
6654  20617 +                 OccupationBusiness_Analysis = 1;
6655  20618 +                 _dm_find = 1;
6656  20619 +              end;
6657  20620 +              else do;
6658  20621 +                 if _dm22 = 'CONSTRUCTION'  then do
      ;
6659  20622 +                    OccupationConstruction = 1;
6660  20623 +                    _dm_find = 1;
6661  20624 +                 end;
6662  20625 +              end;
6663  20626 +           end;
6664  20627 +        end;
6665  20628 +        else do;
6666  20629 +           if _dm22 <= 'DENTISTRY'  then do;
6667  20630 +              if _dm22 = 'CUSTOMER SERVICE'  then d
      o;
6668  20631 +                 OccupationCustomer_Service = 1;
6669  20632 +                 _dm_find = 1;
6670  20633 +              end;
6671  20634 +              else do;
6672  20635 +                 if _dm22 = 'DENTISTRY'  then do;
6673  20636 +                    OccupationDentistry = 1;
```

```
6674  20637 +                  _dm_find = 1;
6675  20638 +               end;
6676  20639 +            end;
6677  20640 +         end;
6678  20641 +         else do;
6679  20642 +            if _dm22 = 'DESIGN'  then do;
6680  20643 +               OccupationDesign = 1;
6681  20644 +               _dm_find = 1;
6682  20645 +            end;
6683  20646 +            else do;
6684  20647 +               if _dm22 = 'EDUCATION'  then do;
6685  20648 +                  OccupationEducation = 1;
6686  20649 +                  _dm_find = 1;
6687  20650 +               end;
6688  20651 +            end;
6689  20652 +         end;
6690  20653 +      end;
6691  20654 +   end;
6692  20655 +   else do;
6693  20656 +      if _dm22 <= 'FINANCE'  then do;
6694  20657 +         if _dm22 <= 'ENTREPRENEURSHIP'  then do;
6695  20658 +            if _dm22 = 'ENGINEERING'  then do;
6696  20659 +               OccupationEngineering = 1;
6697  20660 +               _dm_find = 1;
6698  20661 +            end;
6699  20662 +            else do;
6700  20663 +               if _dm22 = 'ENTREPRENEURSHIP'  the
      n do;
6701  20664 +                  OccupationEntrepreneurship = 1;
6702  20665 +                  _dm_find = 1;
6703  20666 +               end;
6704  20667 +            end;
6705  20668 +         end;
6706  20669 +         else do;
6707  20670 +            if _dm22 = 'EVENT PLANNING'  then do;
6708  20671 +               OccupationEvent_Planning = 1;
```

```
6709 20672 +                  _dm_find = 1;
6710 20673 +               end;
6711 20674 +               else do;
6712 20675 +                  if _dm22 = 'FINANCE'  then do;
6713 20676 +                     OccupationFinance = 1;
6714 20677 +                     _dm_find = 1;
6715 20678 +                  end;
6716 20679 +               end;
6717 20680 +            end;
6718 20681 +         end;
6719 20682 +         else do;
6720 20683 +            if _dm22 <= 'HEALTHCARE'  then do;
6721 20684 +               if _dm22 = 'FITNESS'  then do;
6722 20685 +                  OccupationFitness = 1;
6723 20686 +                  _dm_find = 1;
6724 20687 +               end;
6725 20688 +               else do;
6726 20689 +                  if _dm22 = 'HEALTHCARE'  then do;
6727 20690 +                     OccupationHealthcare = 1;
6728 20691 +                     _dm_find = 1;
6729 20692 +                  end;
6730 20693 +               end;
6731 20694 +            end;
6732 20695 +            else do;
6733 20696 +               if _dm22 = 'HUMAN RESOURCES'  then do
     ;
6734 20697 +                  OccupationHuman_Resources = 1;
6735 20698 +                  _dm_find = 1;
6736 20699 +               end;
6737 20700 +            end;
6738 20701 +         end;
6739 20702 +      end;
6740 20703 +   end;
6741 20704 +   else do;
6742 20705 +      if _dm22 <= 'PHOTOGRAPHY'  then do;
6743 20706 +         if _dm22 <= 'MARKETING'  then do;
```

```
6744  20707 +           if _dm22 <= 'JOURNALISM'  then do;
6745  20708 +              if _dm22 = 'INFORMATION TECHNOLOGY'
      then do;
6746  20709 +                 OccupationInformation_Techno = 1;
6747  20710 +                 _dm_find = 1;
6748  20711 +              end;
6749  20712 +              else do;
6750  20713 +                 if _dm22 = 'JOURNALISM'  then do;
6751  20714 +                    OccupationJournalism = 1;
6752  20715 +                    _dm_find = 1;
6753  20716 +                 end;
6754  20717 +              end;
6755  20718 +           end;
6756  20719 +           else do;
6757  20720 +              if _dm22 = 'LAW ENFORCEMENT'  then do
      ;
6758  20721 +                 OccupationLaw_Enforcement = 1;
6759  20722 +                 _dm_find = 1;
6760  20723 +              end;
6761  20724 +              else do;
6762  20725 +                 if _dm22 = 'MARKETING'  then do;
6763  20726 +                    OccupationMarketing = 1;
6764  20727 +                    _dm_find = 1;
6765  20728 +                 end;
6766  20729 +              end;
6767  20730 +           end;
6768  20731 +        end;
6769  20732 +        else do;
6770  20733 +           if _dm22 <= 'PHARMACY'  then do;
6771  20734 +              if _dm22 = 'MEDICINE'  then do;
6772  20735 +                 OccupationMedicine = 1;
6773  20736 +                 _dm_find = 1;
6774  20737 +              end;
6775  20738 +              else do;
6776  20739 +                 if _dm22 = 'PHARMACY'  then do;
6777  20740 +                    OccupationPharmacy = 1;
```

```
6778 20741 +                     _dm_find = 1;
6779 20742 +                  end;
6780 20743 +               end;
6781 20744 +            end;
6782 20745 +            else do;
6783 20746 +               if _dm22 = 'PHOTOGRAPHY'  then do;
6784 20747 +                  OccupationPhotography = 1;
6785 20748 +                  _dm_find = 1;
6786 20749 +               end;
6787 20750 +            end;
6788 20751 +         end;
6789 20752 +      end;
6790 20753 +      else do;
6791 20754 +         if _dm22 <= 'RESEARCH'  then do;
6792 20755 +            if _dm22 <= 'PLUMBING'  then do;
6793 20756 +               if _dm22 = 'PILOTING'  then do;
6794 20757 +                  OccupationPiloting = 1;
6795 20758 +                  _dm_find = 1;
6796 20759 +               end;
6797 20760 +               else do;
6798 20761 +                  if _dm22 = 'PLUMBING'  then do;
6799 20762 +                     OccupationPlumbing = 1;
6800 20763 +                     _dm_find = 1;
6801 20764 +                  end;
6802 20765 +               end;
6803 20766 +            end;
6804 20767 +            else do;
6805 20768 +               if _dm22 = 'REAL ESTATE'  then do;
6806 20769 +                  OccupationReal_Estate = 1;
6807 20770 +                  _dm_find = 1;
6808 20771 +               end;
6809 20772 +               else do;
6810 20773 +                  if _dm22 = 'RESEARCH'  then do;
6811 20774 +                     OccupationResearch = 1;
6812 20775 +                     _dm_find = 1;
6813 20776 +                  end;
```

```
6814 20777 +              end;
6815 20778 +           end;
6816 20779 +        end;
6817 20780 +        else do;
6818 20781 +           if _dm22 <= 'STUDENT'  then do;
6819 20782 +              if _dm22 = 'SOCIAL WORK'  then do;
6820 20783 +                 OccupationSocial_Work = 1;
6821 20784 +                 _dm_find = 1;
6822 20785 +              end;
6823 20786 +              else do;
6824 20787 +                 if _dm22 = 'STUDENT'  then do;
6825 20788 +                    OccupationStudent = 1;
6826 20789 +                    _dm_find = 1;
6827 20790 +                 end;
6828 20791 +              end;
6829 20792 +           end;
6830 20793 +           else do;
6831 20794 +              if _dm22 = 'WRITING'  then do;
6832 20795 +                 OccupationWriting = 1;
6833 20796 +                 _dm_find = 1;
6834 20797 +              end;
6835 20798 +           end;
6836 20799 +        end;
6837 20800 +     end;
6838 20801 +  end;
6839 20802 +  if not _dm_find then do;
6840 20803 +     delete;
6841 20804 +  end;
6842 20805 +end;
6843 20806 +
6844 20807 +*** Generate dummy variables for PaymentMethod ;
6845 20808 +label PaymentMethodCash = 'PaymentMethod=Cash' ;
6846 20809 +label PaymentMethodCredit_Card = 'PaymentMethod=Cred
     it Card' ;
6847 20810 +label PaymentMethodTnG = 'PaymentMethod=TnG' ;
6848 20811 +*** encoding is sparse, initialize to zero;
```

```
6849  20812 +PaymentMethodCash = 0;
6850  20813 +PaymentMethodCredit_Card = 0;
6851  20814 +PaymentMethodTnG = 0;
6852  20815 +if missing( PaymentMethod ) then do;
6853  20816 +   PaymentMethodCash = .;
6854  20817 +   PaymentMethodCredit_Card = .;
6855  20818 +   PaymentMethodTnG = .;
6856  20819 +end;
6857  20820 +else do;
6858  20821 +   length _dm11 $ 11; drop _dm11 ;
6859  20822 +   _dm11 = put( PaymentMethod , $11. );
6860  20823 +   %DMNORMIP( _dm11 )
6861  20824 +   if _dm11 = 'CREDIT CARD'  then do;
6862  20825 +      PaymentMethodCredit_Card = 1;
6863  20826 +   end;
6864  20827 +   else if _dm11 = 'CASH'  then do;
6865  20828 +      PaymentMethodCash = 1;
6866  20829 +   end;
6867  20830 +   else if _dm11 = 'TNG'  then do;
6868  20831 +      PaymentMethodTnG = 1;
6869  20832 +   end;
6870  20833 +   else do;
6871  20834 +      delete;
6872  20835 +   end;
6873  20836 +end;
6874  20837 +
6875  20838 +*** End Class Look-up, Standardization, Replacement
     ;
6876  20839 +
6877  20840 +
6878  20841 +/*-------------------------------------------------*
     /
6879  20842 +/* Varclus Score Code Begins*/
6880  20843 +/*-------------------------------------------------*
     /
6881  20844 +
```

```
6882  20845 +Clus1 = 0 ; /*---Cluster Component 1------ */
6883  20846 +Clus2 = 0 ; /*---Cluster Component 2------ */
6884  20847 +Clus3 = 0 ; /*---Cluster Component 3------ */
6885  20848 +Clus4 = 0 ; /*---Cluster Component 4------ */
6886  20849 +Clus5 = 0 ; /*---Cluster Component 5------ */
6887  20850 +Clus6 = 0 ; /*---Cluster Component 6------ */
6888  20851 +Clus7 = 0 ; /*---Cluster Component 7------ */
6889  20852 +Clus8 = 0 ; /*---Cluster Component 8------ */
6890  20853 +Clus9 = 0 ; /*---Cluster Component 9------ */
6891  20854 +Clus10 = 0 ; /*---Cluster Component 10------ */
6892  20855 +Clus11 = 0 ; /*---Cluster Component 11------ */
6893  20856 +Clus12 = 0 ; /*---Cluster Component 12------ */
6894  20857 +Clus13 = 0 ; /*---Cluster Component 13------ */
6895  20858 +Clus14 = 0 ; /*---Cluster Component 14------ */
6896  20859 +Clus15 = 0 ; /*---Cluster Component 15------ */
6897  20860 +Clus16 = 0 ; /*---Cluster Component 16------ */
6898  20861 +Clus17 = 0 ; /*---Cluster Component 17------ */
6899  20862 +Clus18 = 0 ; /*---Cluster Component 18------ */
6900  20863 +Clus19 = 0 ; /*---Cluster Component 19------ */
6901  20864 +Clus20 = 0 ; /*---Cluster Component 20------ */
6902  20865 +Clus21 = 0 ; /*---Cluster Component 21------ */
6903  20866 +Clus22 = 0 ; /*---Cluster Component 22------ */
6904  20867 +Clus23 = 0 ; /*---Cluster Component 23------ */
6905  20868 +Clus24 = 0 ; /*---Cluster Component 24------ */
6906  20869 +Clus25 = 0 ; /*---Cluster Component 25------ */
6907  20870 +Clus26 = 0 ; /*---Cluster Component 26------ */
6908  20871 +Clus27 = 0 ; /*---Cluster Component 27------ */
6909  20872 +Clus28 = 0 ; /*---Cluster Component 28------ */
6910  20873 +Clus2 = Clus2+0.49982403901665 * (GenderFemale - 0.4
      9957998319932)/0.50000982428237;
6911  20874 +Clus2 = Clus2+-0.49982403901665 * (GenderMale - 0.50
      042001680067)/0.50000982428237;
6912  20875 +Clus1 = Clus1+0.49986308914933 * (IMP_Churn0 - 0.776
      6310652426)/0.41651193657331;
6913  20876 +Clus1 = Clus1+-0.49986308914933 * (IMP_Churn1 - 0.22
      336893475739)/0.41651193657331;
```

```
6914 20877 +Clus7 = Clus7+0.61221205633923 * (FavouriteCategoryB
     ooks - 0.26281051242049)/0.44016916889722;
6915 20878 +Clus10 = Clus10+0.22354917739188 * (FavouriteCategor
     yClothing - 0.25733029321172)/0.43717165790503;
6916 20879 +Clus28 = Clus28+0.70329858384501 * (FavouriteCategor
     yElectronics - 0.2446897875915)/0.42991172198395;
6917 20880 +Clus7 = Clus7+-0.61219389264702 * (FavouriteCategory
     Home - 0.23516940677627)/0.42411313591853;
6918 20881 +Clus9 = Clus9+-0.67959793107948 * (LocationJohor - 0
     .06776271050842)/0.25134329633423;
6919 20882 +Clus6 = Clus6+0.68320266687987 * (LocationKedah - 0.
     06428257130285)/0.24526053195517;
6920 20883 +Clus14 = Clus14+0.68321243429362 * (LocationKelantan
      - 0.06604264170566)/0.24836158845531;
6921 20884 +Clus12 = Clus12+0.70166225268599 * (LocationKuala_Lu
     mpur - 0.06616264650586)/0.24857116147485;
6922 20885 +Clus13 = Clus13+0.24174030673078 * (LocationMalacca
     - 0.06364254570182)/0.24411996222351;
6923 20886 +Clus7 = Clus7+0.03668597107268 * (LocationNegeri_Sem
     bilan - 0.0670026801072)/0.25003164139475;
6924 20887 +Clus11 = Clus11+-0.67430023620727 * (LocationPahang
     - 0.06796271850874)/0.25168695110365;
6925 20888 +Clus10 = Clus10+0.66037946034645 * (LocationPenang -
      0.06948277931117)/0.2542784086276;
6926 20889 +Clus2 = Clus2+0.01531339136335 * (LocationPerak - 0.
     06640265610624)/0.24898960480391;
6927 20890 +Clus15 = Clus15+0.18171788631907 * (LocationPerlis -
      0.06508260330413)/0.246676492841;
6928 20891 +Clus14 = Clus14+-0.68321243429362 * (LocationPutraja
     ya - 0.06684267370694)/0.2497543310285;
6929 20892 +Clus11 = Clus11+0.67465452639828 * (LocationSabah -
     0.06640265610624)/0.24898960480391;
6930 20893 +Clus9 = Clus9+0.68116519717342 * (LocationSarawak -
     0.06880275211008)/0.25312348048835;
6931 20894 +Clus10 = Clus10+-0.64710694549097 * (LocationSelango
     r - 0.06540261610464)/0.24723988171709;
```

```
6932 20895 +Clus6 = Clus6+-0.68320266687987 * (LocationTerenggan
     u - 0.06872274890995)/0.25298713978054;
6933 20896 +Clus3 = Clus3+0.5 * (M_Variable0 - 0.99863994559782)
     /0.03685456534449;
6934 20897 +Clus3 = Clus3+-0.5 * (M_Variable1 - 0.00136005440217
     )/0.03685456534449;
6935 20898 +Clus10 = Clus10+0.14949580475937 * (MembershipLevelB
     ronze - 0.24464978599143)/0.42988796290185;
6936 20899 +Clus5 = Clus5+0.61054829334456 * (MembershipLevelGol
     d - 0.25705028201128)/0.43701610285863;
6937 20900 +Clus5 = Clus5+-0.61054829334456 * (MembershipLevelPl
     atinum - 0.25189007560302)/0.43410713387066;
6938 20901 +Clus28 = Clus28+-0.703298583845 * (MembershipLevelSi
     lver - 0.24640985639425)/0.43092861049348;
6939 20902 +Clus21 = Clus21+0.69461265313387 * (OccupationArchit
     ecture - 0.03608144325773)/0.18649655225869;
6940 20903 +Clus22 = Clus22+0.69486847071608 * (OccupationArts -
      0.03496139845593)/0.1836857336482;
6941 20904 +Clus15 = Clus15+-0.69272712743273 * (OccupationBusin
     ess_Analysis - 0.03536141445657)/0.18469528790081;
6942 20905 +Clus21 = Clus21+-0.69461265313387 * (OccupationConst
     ruction - 0.0340013600544)/0.18123625874716;
6943 20906 +Clus18 = Clus18+0.69443715115543 * (OccupationCustom
     er_Service - 0.03412136485459)/0.18154452815884;
6944 20907 +Clus17 = Clus17+0.69440598899639 * (OccupationDentis
     try - 0.03584143365734)/0.18589837996694;
6945 20908 +Clus19 = Clus19+0.6947945092869 * (OccupationDesign
     - 0.03440137605504)/0.18226148853699;
6946 20909 +Clus20 = Clus20+0.69457174876623 * (OccupationEducat
     ion - 0.03484139365574)/0.18338161344672;
6947 20910 +Clus12 = Clus12+0.701662252686 * (OccupationEngineer
     ing - 0.03464138565542)/0.18287344754425;
6948 20911 +Clus24 = Clus24+1 * (OccupationEntrepreneurship - 0.
     03356134245369)/0.18010073916937;
6949 20912 +Clus16 = Clus16+0.69428719319622 * (OccupationEvent_
     Planning - 0.03684147365894)/0.18837621654105;
```

```
6950 20913 +Clus27 = Clus27+0.69510696219125 * (OccupationFinanc
     e - 0.0344413776551)/0.18236364624978;
6951 20914 +Clus18 = Clus18+-0.69443715115543 * (OccupationFitne
     ss - 0.03696147845913)/0.18867101392707;
6952 20915 +Clus26 = Clus26+1 * (OccupationHealthcare - 0.032761
     31045241)/0.17801481570798;
6953 20916 +Clus11 = Clus11+0.14652628146335 * (OccupationHuman_
     Resources - 0.03028121124844)/0.17136345630889;
6954 20917 +Clus16 = Clus16+-0.69428719319622 * (OccupationInfor
     mation_Techno - 0.03504140165606)/0.18388815751518;
6955 20918 +Clus1 = Clus1+0.01350845549106 * (OccupationJournali
     sm - 0.03508140325613)/0.18398927288833;
6956 20919 +Clus22 = Clus22+-0.69486847071608 * (OccupationLaw_E
     nforcement - 0.03368134725389)/0.18041124156153;
6957 20920 +Clus25 = Clus25+1 * (OccupationMarketing - 0.0329213
     1685267)/0.17843423817025;
6958 20921 +Clus13 = Clus13+-0.65918775447905 * (OccupationMedic
     ine - 0.03464138565542)/0.18287344754425;
6959 20922 +Clus15 = Clus15+0.6714947619848 * (OccupationPharmac
     y - 0.03460138405536)/0.18277161855399;
6960 20923 +Clus23 = Clus23+0.6949784627025 * (OccupationPhotogr
     aphy - 0.03492139685587)/0.18358442494144;
6961 20924 +Clus13 = Clus13+0.68549427561095 * (OccupationPiloti
     ng - 0.03504140165606)/0.18388815751518;
6962 20925 +Clus20 = Clus20+-0.69457174876623 * (OccupationPlumb
     ing - 0.0354414176567)/0.18489643342567;
6963 20926 +Clus9 = Clus9+0.07416114195486 * (OccupationReal_Est
     ate - 0.03352134085363)/0.17999710156308;
6964 20927 +Clus23 = Clus23+-0.6949784627025 * (OccupationResear
     ch - 0.03312132485299)/0.17895693271726;
6965 20928 +Clus19 = Clus19+-0.6947945092869 * (OccupationSocial
     _Work - 0.03464138565542)/0.18287344754425;
6966 20929 +Clus17 = Clus17+-0.69440598899639 * (OccupationStude
     nt - 0.03536141445657)/0.18469528790081;
6967 20930 +Clus27 = Clus27+-0.69510696219125 * (OccupationWriti
     ng - 0.03288131525261)/0.17832948849534;
```

```
6968  20931 +Clus4 = Clus4+0.55038238175606 * (PaymentMethodCash
      - 0.13572542901716)/0.34250361964824;
6969  20932 +Clus4 = Clus4+-0.55038238175606 * (PaymentMethodCred
      it_Card - 0.72938917556702)/0.44428425809226;
6970  20933 +Clus8 = Clus8+1 * (PaymentMethodTnG - 0.134885395415
      81)/0.34160795298967;
6971  NOTE: %INCLUDE (level 1) ending.
6972  20934  run;
6973
6974  NOTE: DATA STEP view saved on file EMWS4.VARCLUS2_TRAIN.
6975  NOTE: A stored DATA STEP view cannot run under a different
      operating system.
6976  NOTE: View EMWS4.IMPT_TRAIN.VIEW used (Total process time):
6977        real time           0.09 seconds
6978        user cpu time       0.05 seconds
6979        system cpu time     0.05 seconds
6980        memory              165287.75k
6981        OS Memory           176264.00k
6982        Timestamp           07/01/2024 05:54:48 AM
6983        Step Count                        1  Switch Count  7
6984        Page Faults                       0
6985        Page Reclaims                     33171
6986        Page Swaps                        0
6987        Voluntary Context Switches        29
6988        Involuntary Context Switches      2
6989        Block Input Operations            0
6990        Block Output Operations           520
6991
6992  NOTE: DATA statement used (Total process time):
6993        real time           0.10 seconds
6994        user cpu time       0.05 seconds
6995        system cpu time     0.06 seconds
6996        memory              165287.75k
6997        OS Memory           176264.00k
6998        Timestamp           07/01/2024 05:54:48 AM
6999        Step Count                        1  Switch Count  4
```

```
7000        Page Faults                          0
7001        Page Reclaims                        33240
7002        Page Swaps                           0
7003        Voluntary Context Switches           32
7004        Involuntary Context Switches         2
7005        Block Input Operations               0
7006        Block Output Operations              520
7007
7008
7009  20935  quit;
7010  20936  filename emflow;
7011  NOTE: Fileref EMFLOW has been deassigned.
7012
7013  20938  *-------------------------------------------------
      ---------*;
7014  20939  * VarClus2: Computing metadata for TRAIN data;
7015  20940  *-------------------------------------------------
      ---------*;
7016
7017  NOTE: View EMWS4.VARCLUS2_TRAIN.VIEW used (Total process ti
      me):
7018        real time            0.12 seconds
7019        user cpu time        0.04 seconds
7020        system cpu time      0.08 seconds
7021        memory               230359.31k
7022        OS Memory            241540.00k
7023        Timestamp            07/01/2024 05:54:48 AM
7024        Step Count                        1  Switch Count  9
7025        Page Faults                       0
7026        Page Reclaims                     49523
7027        Page Swaps                        0
7028        Voluntary Context Switches        35
7029        Involuntary Context Switches      0
7030        Block Input Operations            0
7031        Block Output Operations           0
7032
```

```
7033 NOTE: View EMWS4.VARCLUS2_TRAIN.VIEW used (Total process ti
     me):
7034       real time           0.12 seconds
7035       user cpu time       0.03 seconds
7036       system cpu time     0.09 seconds
7037       memory              235458.71k
7038       OS Memory           245980.00k
7039       Timestamp           07/01/2024 05:54:49 AM
7040       Step Count                        1  Switch Count  9
7041       Page Faults                       0
7042       Page Reclaims                     49520
7043       Page Swaps                        0
7044       Voluntary Context Switches        34
7045       Involuntary Context Switches      0
7046       Block Input Operations            0
7047       Block Output Operations           0
7048
7049 *------------------------------------------------------------
     --*
7050 * Report Log
7051 Date:                07 January 2024
7052 Time:                05:54:49
7053 *------------------------------------------------------------
     --*
7054 21336  %let EMEXCEPTIONSTRING=;
7055 21337  *-------------------------------------------------
     ---------*;
7056 21338  * REPORT: VarClus2;
7057 21339  *-------------------------------------------------
     ---------*;
7058 21340  %let EM_ACTION = REPORT;
7059 21341  %let syscc = 0;
7060 21342
7061 21343  %macro main;
7062 21344
7063 21345     filename temp catalog 'sashelp.emexpl.variableclu
```

```
     stering_macros.source';
7064 21346     %include temp;
7065 21347     filename temp catalog 'sashelp.emexpl.variableclu
     stering_macros2.source';
7066 21348     %include temp;
7067 21349     filename temp;
7068 21350
7069 21351    %SetProperties;
7070 21352
7071 21353     %if %upcase(&EM_ACTION) = CREATE %then %do;
7072 21354         filename temp catalog 'sashelp.emexpl.variabl
     eclustering_create.source';
7073 21355         %include temp;
7074 21356         filename temp;
7075 21357         %create;
7076 21358     %end;
7077 21359     %else
7078 21360     %if %upcase(&EM_ACTION) = TRAIN %then %do;
7079 21361          filename temp catalog 'sashelp.emexpl.variab
     leclustering_train.source';
7080 21362             %include temp;
7081 21363             filename temp;
7082 21364             %train;
7083 21365     %end;
7084 21366     %else
7085 21367     %if %upcase(&EM_ACTION) = SCORE %then %do;
7086 21368             filename temp catalog 'sashelp.emexpl.var
     iableclustering_score.source';
7087 21369             %include temp;
7088 21370             filename temp;
7089 21371             %score;
7090 21372     %end;
7091 21373     %else
7092 21374     %if %upcase(&EM_ACTION) = REPORT %then %do;
7093 21375             filename temp catalog 'sashelp.emexpl.var
     iableclustering_report.source';
```

```
7094  21376             %include temp;
7095  21377             filename temp;
7096  21378             %report;
7097  21379     %end;
7098  21380     /*
7099  21381     %if %upcase(&EM_ACTION) = OPENTESTTABLE %then %do
      ;
7100  21382         %put 'OPENING TABLE';
7101  21383     %end;
7102  21384     %if %upcase(&EM_ACTION) = CLOSETESTTABLE %then %d
      o;
7103  21385         %put 'CLOSE TABLE';
7104  21386     %end;
7105  21387     */
7106  21388  %mend main;
7107  21389  %main;
7108  NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMEXPL.V
      ARIABLECLUSTERING_MACROS.SOURCE.
7109  21390 +
7110  21391 +/* Initialize property macro variables */
7111  21392 +%macro SetProperties;
7112  21393 +   %em_checkmacro(name=EM_PROPERTY_MAXCLUS,       gl
      obal=Y, value=DEFAULT);
7113  21394 +   %em_checkmacro(name=EM_PROPERTY_HIDEVARIABLE,  gl
      obal=Y, value=Y);
7114  21395 +   %em_checkmacro(name=EM_PROPERTY_PRINTOPTION,   gl
      obal=Y, value=SHORT);
7115  21396 +   %em_checkmacro(name=EM_PROPERTY_CLUSSOURCE,    gl
      obal=Y, value=CORR);
7116  21397 +   %em_checkmacro(name=EM_PROPERTY_CLUSCOMP,      gl
      obal=Y, value=PRINCIPAL);
7117  21398 +   %em_checkmacro(name=EM_PROPERTY_CLUSHIERACHY,
        global=Y, value=Y);
7118  21399 +   %em_checkmacro(name=EM_PROPERTY_INCLUDECLASSVAR,
           global=Y, value=N);
7119  21400 +   %em_checkmacro(name=EM_PROPERTY_EXPORTEDCOMP,
```

```
       global=Y, value=CLUSTERCOMP);
7120 21401 +   %em_checkmacro(name=EM_PROPERTY_MAXEIGEN,
      global=Y, value=DEFAULT);
7121 21402 +   %em_checkmacro(name=EM_PROPERTY_PROPORTION,
     global=Y, value=DEFAULT);
7122 21403 +   %em_checkmacro(name=EM_PROPERTY_PRINTOPTION,
      global=Y, value=SHORT);
7123 21404 +   %em_checkmacro(name=EM_PROPERTY_TWOSTAGECLUS,
       global=Y, value=AUTO);
7124 21405 +   %em_checkmacro(name=EM_PROPERTY_SUPPRESSSAMPWARN,
           global=Y, value=N);
7125 21406 +
7126 21407 +%mend SetProperties;
7127 21408 +
7128 21409 +%Macro MakeDummyVariables(indata=,
7129 21410 +                          outvar=,
7130 21411 +                          outdata=,
7131 21412 +                          fileref=,
7132 21413 +                          recreatecmeta=N, /* option
     al */
7133 21414 +                          incmeta=,     /* optional
     */
7134 21415 +                          outcmeta=,    /* optional
     */
7135 21416 +                          ndummyvars=_ndummyvars
7136 21417 +                          );
7137 21418 +    %global &ndummyvars;
7138 21419 +
7139 21420 +    proc dmdb batch data=&indata out=_dmdbdat dmdbca
     t=_dmdbcat classout=_classout;;
7140 21421 +       class
7141 21422 +       %EM_BINARY_INPUT %EM_NOMINAL_INPUT %EM_ORDINA
     L_INPUT
7142 21423 +       %EM_BINARY_REJECTED %EM_NOMINAL_REJECTED %EM_
     ORDINAL_REJECTED
7143 21424 +    ;
```

```
7144  21425 +    run;
7145  21426 +    %let &ndummyvars = 0;
7146  21427 +    data _null_;
7147  21428 +    %let dsid = %sysfunc(open(work._classout));
7148  21429 +      %let &ndummyvars = %sysfunc(attrn(&dsid, NOBS)
      );
7149  21430 +    %let dsid = %sysfunc(close(&dsid));
7150  21431 +    run;
7151  21432 +
7152  21433 +    proc dmzip data=_dmdbdat dmdbcat=_dmdbcat;
7153  21434 +       input
7154  21435 +       %EM_BINARY_INPUT %EM_NOMINAL_INPUT %EM_ORDINA
      L_INPUT
7155  21436 +       %EM_BINARY_REJECTED %EM_NOMINAL_REJECTED %EM_
      ORDINAL_REJECTED
7156  21437 +        / level=nominal stdize=no;
7157  21438 +       make outvar = &outvar;
7158  21439 +       score data = &indata out =&outdata;
7159  21440 +       code  file=  "&fileref";
7160  21441 +    run;
7161  21442 +    %if &recreatecmeta eq Y %then %do;
7162  21443 +    proc contents data =&outvar out=_tmpds(keep=NAME
       LABEL);
7163  21444 +    data _tmpds;
7164  21445 +        set _tmpds;
7165  21446 +           ROLE = 'INPUT';
7166  21447 +           LEVEL = 'INTERVAL';
7167  21448 +           CREATOR='DMZIP';
7168  21449 +           if NAME = '_TYPE_' then delete;
7169  21450 +    run;
7170  21451 +    data &outcmeta;
7171  21452 +         set &incmeta _tmpds;
7172  21453 +    run;
7173  21454 +    %end;
7174  21455 +    proc datasets lib=work nolist;
7175  21456 +      delete  _dmdbdat _dmdbcat _classout
```

```
7176 21457 +    %if &recreatecmeta eq Y %then %do;
7177 21458 +    _tmpds
7178 21459 +    %end;
7179 21460 +    ;
7180 21461 +    quit;
7181 21462 +%Mend MakeDummyVariables;
7182 21463 +
7183 21464 +/*--- Determine Optimal Number of Cluster ----
7184 21465 +%macro FindClusNum(statds=, groupds=, minvariation=)
     ;
7185 21466 +   %global optnclus;
7186 21467 +   data varclus_tmp(drop=_NAME_);
7187 21468 +      set &statDs;
7188 21469 +      where _type_ ='PROPOR';
7189 21470 +   run;
7190 21471 +   proc sort data=varclus_tmp;
7191 21472 +      by _NCL_;
7192 21473 +   run;
7193 21474 +   proc transpose data=varclus_tmp out=varclus_tmp;
7194 21475 +      by _NCL_;
7195 21476 +      var %EM_INTERVAL_INPUT
7196 21477 +      %if &EM_PROPERTY_INCLUDECLASSVAR eq Y %then %d
     o;
7197 21478 +      %let dsid = %sysfunc(open(&EM_USER_OUTDUMMY));
7198 21479 +      %let nvar = %sysfunc(attrn(&dsid, NVAR));
7199 21480 +          %do i = 2 %to &nvar;
7200 21481 +          %let varname = %sysfunc(varname(&dsid, &i)
     );
7201 21482 +          &varname
7202 21483 +          %end;
7203 21484 +      %end;
7204 21485 +   ;
7205 21486 +   run;
7206 21487 +
7207 21488 +   %if &minVariation eq %then %do;
7208 21489 +       %let minVariation = &EM_PROPERTY_MINVARIATION
```

```
     ;
7209 21490 +   %end;
7210 21491 +   %if ^(0<&minVariation<100) %then %do;
7211 21492 +       %let minVariation = 90;
7212 21493 +   %end;
7213 21494 +
7214 21495 +   data _null_;
7215 21496 +      set varclus_tmp end=eof;
7216 21497 +      by _NCL_;
7217 21498 +      retain flag 0;
7218 21499 +      if first._ncl_ then flag=0;
7219 21500 +      if .<col1 < &minVariation then flag=1;
7220 21501 +      if last._ncl_ and ^flag then do;
7221 21502 +         call symput('OPTNCL', _ncl_);
7222 21503 +         stop;
7223 21504 +      end;
7224 21505 +     if eof then call symput('OPTNCL', _ncl_);
7225 21506 +   run;
7226 21507 +
7227 21508 +   %let optnclus = &OPTNCL;
7228 21509 +
7229 21510 +   data varclus_tmp(drop=_NCL_ _NAME_);
7230 21511 +      set &statDs;
7231 21512 +      where _type_ in('RSQUARED' 'GROUP') and _NCL_=
     &OPTNCL;
7232 21513 +   run;
7233 21514 +   proc sort data=varclus_tmp;
7234 21515 +      by _TYPE_;
7235 21516 +   run;
7236 21517 +   proc transpose data=varclus_tmp out=varclus_tmp;
7237 21518 +      by _TYPE_;
7238 21519 +   run;
7239 21520 +   proc sort data=varclus_tmp;
7240 21521 +      by _name_ _type_;
7241 21522 +   run;
7242 21523 +
```

```
7243  21524 +  proc transpose data=varclus_tmp out=&groupds;
7244  21525 +      by _NAME_;
7245  21526 +   run;
7246  21527 +   proc sort data=&groupDs(rename=(col1=Cluster col2
      =Rsquare _NAME_=VARIABLE));
7247  21528 +      by Cluster descending Rsquare;
7248  21529 +      where Cluster ne 0;
7249  21530 +   run;
7250  21531 +   proc datasets lib=work nolist mt=(DATA VIEW);
7251  21532 +      delete varclus_tmp;
7252  21533 +   run;
7253  21534 +   quit;
7254  21535 +%mend findClusNum;
7255  21536 +*/
7256  21537 +
7257  21538 +%macro getNclusfromTrain(inoutstat=, nc=);
7258  21539 +%global &nc;
7259  21540 +data _null_;
7260  21541 +    set &inoutstat end=eof;
7261  21542 +    if eof then do;
7262  21543 +    call symput("&nc", _ncl_);
7263  21544 +    end;
7264  21545 +run;
7265  21546 +%mend  getNclusfromTrain;
7266  21547 +
7267  21548 +%macro MakeDeltaCode(groupds=, outstatscore=, deltac
      odefile=);
7268  21549 +
7269  21550 +     *--- Build Code to Modify Metadata ---*;
7270  21551 +     filename X "&deltacodefile";
7271  21552 +     data _null_;
7272  21553 +        FILE X;
7273  21554 +        set &groupds end=eof;
7274  21555 +        /*by Cluster;*/
7275  21556 +         if _N_=1 then do;
7276  21557 +           %if &EM_PROPERTY_INCLUDECLASSVAR eq Y %th
```

```
     en %do;
7277 21558 +            put "if upcase(strip(ROLE)) ='INPUT' and
      upcase(strip(LEVEL)) ^='INTERVAL' then ROLE ='REJECTED' ;"
     ;
7278 21559 +           %end;
7279 21560 +           put "if upcase(strip(ROLE))='INPUT' and u
     pcase(strip(LEVEL))='INTERVAL' then do;";
7280 21561 +           put "if upcase(strip(NAME)) in (";
7281 21562 +        end;
7282 21563 +        if Strip(upcase(Selected)) eq 'YES' then do;
7283 21564 +           string = '"'!!trim(left(VARIABLE))!!'"';
7284 21565 +           put string;
7285 21566 +        end;
7286 21567 +        if eof then do;
7287 21568 +           put ') then ROLE="INPUT";';
7288 21569 +           put 'else ROLE="REJECTED";';
7289 21570 +           put 'end;';
7290 21571 +
7291 21572 +           %if %upcase(&EM_PROPERTY_HIDEVARIABLE) eq
      Y %then %do;
7292 21573 +             put 'if upcase(strip(ROLE)) = "REJECTED
     " then delete ;';
7293 21574 +           %end;
7294 21575 +        end;
7295 21576 +     run;
7296 21577 +     quit;
7297 21578 +
7298 21579 +     filename X;
7299 21580 +
7300 21581 +     quit;
7301 21582 +%mend MakeDeltaCode;
7302 21583 +
7303 21584 +%macro MakeVarClusCorrData(statds=, corrds=, corrplo
     tds= );
7304 21585 +    %if ^%sysfunc(exist(&statds)) %then %do;
7305 21586 +         %goto doendc;
```

```
7306  21587 +    %end;
7307  21588 +
7308  21589 +    data &corrds(drop=_TYPE_ _NCL_) ;
7309  21590 +       set &statds;
7310  21591 +       where _type_ eq 'CORR' ;
7311  21592 +    run ;
7312  21593 +    proc sort data=&corrds;
7313  21594 +       by _NAME_ ;
7314  21595 +    run ;
7315  21596 +    proc transpose data=&corrds out=&corrplotds name
     =_TMP_;
7316  21597 +      BY _NAME_ ;
7317  21598 +    run ;
7318  21599 +    data &corrplotds;
7319  21600 +       length _Y_ $100;
7320  21601 +       set &corrplotDs;
7321  21602 +       if _LABEL_ ne '' then _Y_=_LABEL_ ; else _Y_=
     _TMP_ ;
7322  21603 +    run ;
7323  21604 +    data varclus_match(rename=(_TMP_= _NAME_ _LABEL_
     =_X_));
7324  21605 +       set &corrplotds;
7325  21606 +       where _LABEL_ ne '' ;
7326  21607 +       keep _TMP_ _LABEL_ ;
7327  21608 +    run ;
7328  21609 +    data _null_;
7329  21610 +       nobs=0;
7330  21611 +       dsid = open('varclus_match');
7331  21612 +       if dsid then do;
7332  21613 +          nobs = attrn(dsid, 'NOBS');
7333  21614 +          dsid = close(dsid);
7334  21615 +       end;
7335  21616 +       call symput ('CORR_NOBS', nobs);
7336  21617 +    run;
7337  21618 +    %if &corr_nobs %then %do;
7338  21619 +        proc sort data=varclus_match;
```

```
7339  21620 +           by _name_;
7340  21621 +        run ;
7341  21622 +        proc sort data=&corrplotds;
7342  21623 +           by _name_;
7343  21624 +        run ;
7344  21625 +        data &corrplotds(keep= _X_ _Y_ col1 rename=(
      col1=Correlation)) ;
7345  21626 +           merge varclus_match &corrplotds;
7346  21627 +           by _NAME_ ;
7347  21628 +           if _X_ eq '' then _X_=_NAME_ ;
7348  21629 +           label _X_ = "%sysfunc(sasmsg(sashelp.dmin
      e, rpt_varclus_label_variable, noquote))";
7349  21630 +           label _Y_ = "%sysfunc(sasmsg(sashelp.dmin
      e, rpt_varclus_label_variable, noquote))";
7350  21631 +           label col1 = "%sysfunc(sasmsg(sashelp.dmi
      ne, rpt_correlation_vlabel, noquote))";
7351  21632 +
7352  21633 +       run ;
7353  21634 +    %end;
7354  21635 +    %else %do;
7355  21636 +        proc sort data=&corrplotds;
7356  21637 +           by _name_;
7357  21638 +        run ;
7358  21639 +        data &corrplotds(keep= _NAME_ _Y_ col1 renam
      e=(_NAME_=_X_ col1=Correlation)) ;
7359  21640 +           set &corrplotds;
7360  21641 +           label _NAME_ = "%sysfunc(sasmsg(sashelp.d
      mine, rpt_varclus_label_variable, noquote))" ;
7361  21642 +           label _Y_ = "%sysfunc(sasmsg(sashelp.dmin
      e, rpt_varclus_label_variable, noquote))" ;
7362  21643 +           label col1 = "%sysfunc(sasmsg(sashelp.dmi
      ne, rpt_correlation_vlabel, noquote))";
7363  21644 +
7364  21645 +        run ;
7365  21646 +    %end;
7366  21647 +    proc sort data=&corrplotds;
```

```
7367  21648 +       by _X_ _Y_;
7368  21649 +    run ;
7369  21650 +    proc datasets lib=work nolist mt=(DATA VIEW);
7370  21651 +       delete varclus_match;
7371  21652 +    run;
7372  21653 +    quit;
7373  21654 +
7374  21655 +%doendc:
7375  21656 +
7376  21657 +%mend MakeVarClusCorrData;
7377  21658 +
7378  21659 +%macro MakeStatPlotData(statds= , outstatplotds=);
7379  21660 +   %if %sysfunc(exist(&statds)) %then %do;
7380  21661 +
7381  21662 +       data varclus_tmp(drop=_NAME_ _NCL_) ;
7382  21663 +          set &statDs;
7383  21664 +          where _type_ in('MEAN', 'STD', 'N');
7384  21665 +       run ;
7385  21666 +       proc transpose data=varclus_tmp out=&outstatp
      lotds;
7386  21667 +          id _TYPE_ ;
7387  21668 +       run ;
7388  21669 +       data &outstatplotds;
7389  21670 +          set &outstatplotds(obs=1000);
7390  21671 +          label _name_= "%sysfunc(sasmsg(sashelp.dmi
      ne, rpt_varclus_label_name, noquote))";
7391  21672 +          label _label_="%sysfunc(sasmsg(sashelp.dmi
      ne, rpt_varclus_label_label, noquote))" ;
7392  21673 +          if MEAN ne 0 then SCALEDSTD= STD / MEAN ;
7393  21674 +          else SCALEDSTD= STD ;
7394  21675 +          label SCALEDSTD = "%sysfunc(sasmsg(sashelp
      .dmine, rpt_varclus_label_scaledstd, noquote))";
7395  21676 +       run ;
7396  21677 +       proc sort data=&outstatplotds;
7397  21678 +          by descending SCALEDSTD ;
7398  21679 +       run ;
```

```
7399 21680 +       proc datasets lib=work nolist mt=(DATA VIEW);
7400 21681 +          delete varclus_tmp;
7401 21682 +       run;
7402 21683 +       quit;
7403 21684 +    %end;
7404 21685 +
7405 21686 +%mend MakeStatPlotData;
7406 21687 +
7407 21688 +
7408 21689 +%macro CreateScoreCode(indata=, ncluscomp=, fileref=
     );
7409 21690 +     %EM_GETNAME(KEY=OUTSTATSCORE, type=DATA);
7410 21691 +    data &EM_USER_OUTSTATSCORE;
7411 21692 +          set &indata;
7412 21693 +          if (_TYPE_ in ('SCORE' 'MEAN' 'STD') and _
     NCL_ = &ncluscomp ) or (_TYPE_ in ('MEAN' 'STD'));
7413 21694 +          if _TYPE_ = 'MEAN' then _NAME_='MEAN';
7414 21695 +          if _TYPE_ = 'STD' then _NAME_='STD';
7415 21696 +          DROP _TYPE_ _NCL_;
7416 21697 +     run;
7417 21698 +
7418 21699 +     filename _file_  "&fileRef";
7419 21700 +
7420 21701 +     data _null_;
7421 21702 +        FILE _file_ MOD;
7422 21703 +        put ' ';
7423 21704 +        put '/*-------------------------------------
     ------------*/';
7424 21705 +        put '/* ' "%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_score_title_begin , noquote))" '*/';
7425 21706 +        put '/*-------------------------------------
     ------------*/';
7426 21707 +        put ' ';
7427 21708 +        %let dsid = %sysfunc(open(&EM_USER_OUTSTATSC
     ORE));
7428 21709 +
```

```
7429 21710 +        %let nvar = %sysfunc(attrn(&dsid, NVAR));
7430 21711 +        %let vn_name =%sysfunc(varnum(&dsid, _NAME_)
     );
7431 21712 +
7432 21713 +        %let k = 1;
7433 21714 +        %do %while(^%sysfunc(fetch(&dsid)));
7434 21715 +                %let _name = %sysfunc(getvarc(&dsid,
      &vn_name));
7435 21716 +                %if &k > 2 %then %do;
7436 21717 +                 %let cn = %eval(&k-2);
7437 21718 +                 put "&_name = 0 ; /*---" "%sysfunc(
     sasmsg(sashelp.dmine, rpt_varclus_score_cluscompnum, noquot
     e, &cn))"  "------ */";
7438 21719 +                %end;
7439 21720 +                %let k = %eval(&k+1);
7440 21721 +        %end;
7441 21722 +
7442 21723 +        %let rc = %sysfunc(rewind(&dsid));
7443 21724 +
7444 21725 +        %do i= 2 %to &nvar;
7445 21726 +            %let _varname =  %sysfunc(varname(&dsid,
      &i));
7446 21727 +            %do %while(^%sysfunc(fetch(&dsid)));
7447 21728 +                %let _name = %sysfunc(getvarc(&dsid,
      &vn_name));
7448 21729 +                %if &_name = MEAN %then
7449 21730 +                %let _mean = %sysfunc(getvarn(&dsid,
      &i));
7450 21731 +                %else %if &_name = STD %then
7451 21732 +                %let _std = %sysfunc(getvarn(&dsid,
     &i));
7452 21733 +                %else %do;
7453 21734 +                      %let coeff =  %sysfunc(getvarn
     (&dsid, &i));
7454 21735 +                      %let abscoeff = %sysfunc(abs(&
     coeff));
```

```
7455 21736 +                          %if &abscoeff >  0 %then %
     do;
7456 21737 +                       put "&_name = &_name+&coeff *
      (&_varname - &_mean)/&_std;";
7457 21738 +                          %end;
7458 21739 +                %end;
7459 21740 +             %end;
7460 21741 +             %let rc = %sysfunc(rewind(&dsid));
7461 21742 +          %end;
7462 21743 +
7463 21744 +        %let dsid= %sysfunc(close(&dsid));
7464 21745 +       run;
7465 21746 +%mend CreateScoreCode;
7466 21747 +
7467 21748 +
7468 21749 +
7469 21750 +/*-------------------------------------------------
     --------
7470 21751 +    Instead of using %MakeRSquareData,
7471 21752 +    %MakeVarClusResultTable at macro2.source is used
7472 21753 + +-----------------------------------------------
     --------*/
7473 21754 +
7474 21755 +
7475 21756 +%macro MakeRSquareData(indata=, inClusRSquare=, outd
     ata=, ncluster=);
7476 21757 +
7477 21758 +/* modifying  from ods rsquare = data */
7478 21759 +
7479 21760 +data &outdata(drop= ControlVar  NumberOfClusters Cur
     rentCluster);
7480 21761 +    Length Cluster $16;
7481 21762 +    length Variable $32;
7482 21763 +    Length VariableLabel $64;
7483 21764 +    set &indata; retain CurrentCluster;
7484 21765 +    if NumberOfClusters ^= &ncluster then delete;
```

```
7485 21766 +    if strip(Cluster) eq '' then Cluster = CurrentCl
     uster;
7486 21767 +    CurrentCluster = Cluster;
7487 21768 +   run;
7488 21769 +proc sort data =&outdata ;
7489 21770 +     by Cluster RsquareRatio;
7490 21771 +run;
7491 21772 +data _tmprsq(drop=index);
7492 21773 +     set &outdata; by Cluster;
7493 21774 +     if first.Cluster then do;
7494 21775 +     index = strip(scan(Cluster,2 ));
7495 21776 +     Variable = "Clus"||index;
7496 21777 +     VariableLabel = "Cluster Component "||index;
7497 21778 +     OwnCluster = 1;
7498 21779 +     NextClosest = .;
7499 21780 +     RsquareRatio = 0;
7500 21781 +     output;
7501 21782 +     end;
7502 21783 +run;
7503 21784 +
7504 21785 +proc sort data = _tmprsq ;
7505 21786 +     by Cluster RsquareRatio;
7506 21787 +run;
7507 21788 +data &outdata;
7508 21789 +     set &outdata _tmprsq;
7509 21790 +by Cluster;
7510 21791 +run;
7511 21792 +
7512 21793 +
7513 21794 +/* Just create the Selected variable with all YES */
7514 21795 +
7515 21796 +data &outdata;
7516 21797 +    set &outdata; by cluster;
7517 21798 +    length Selected $8;
7518 21799 +    Selected = 'YES';
7519 21800 +    label  OwnCluster = 'R-Sqaure with Cluster Compo
```

```
     nent';
7520 21801 +    label  NextClosest = 'R-Sqaure with Next Cluster
      Component';
7521 21802 +    rename OwnCluster = RSqWithClusterComp;
7522 21803 +    rename NextClosest = RSqWithNextClusComp;
7523 21804 +run;
7524 21805 +
7525 21806 +
7526 21807 +/* Selected = Y/N will be done %score section -----
7527 21808 +
7528 21809 +%if &EM_PROPERTY_EXPORTEDCOMP ne CLUSTERCOMP %then %
     do;
7529 21810 +data &outdata;
7530 21811 +    set &outdata; by cluster;
7531 21812 +    length Selected $8;
7532 21813 +    if first.Cluster then Selected = 'YES';
7533 21814 +    else Selected = 'NO';
7534 21815 +    label  OwnCluster = 'R-Sqaure with Cluster Compo
     nent';
7535 21816 +    label  NextClosest = 'R-Sqaure with Next Cluster
      Component';
7536 21817 +    rename OwnCluster = RSqWithClusterComp;
7537 21818 +    rename NextClosest = RSqWithNextClusComp;
7538 21819 +run;
7539 21820 +%end;
7540 21821 +%else %do;
7541 21822 +data &outdata;
7542 21823 +    set &outdata; by cluster;
7543 21824 +    if last.Cluster then Selected = 'YES';
7544 21825 +    else Selected = 'NO';
7545 21826 +    label  OwnCluster = 'R-Sqaure with Cluster Compo
     nent';
7546 21827 +    label  NextClosest = 'R-Sqaure with Next Cluster
      Component';
7547 21828 +    rename OwnCluster = RSqWithClusterComp;
7548 21829 +    rename NextClosest = RSqWithNextClusComp;
```

```
7549  21830 +run;
7550  21831 +%end;
7551  21832 +-----------------------------------------------------
      -----------*/
7552  21833 +
7553  21834 +%if %sysfunc(exist(&inClusRSquare)) %then %do;
7554  21835 +/* to calculate NextClosestClusRsq */
7555  21836 +proc transpose data = &inClusRSquare  out=_clusRsq;
7556  21837 +      by cluster;
7557  21838 +      run;
7558  21839 +data _clusRsq;
7559  21840 +     set _clusRsq;
7560  21841 +     if strip(upcase(Cluster)) eq strip(upcase(_NAME
      _)) then delete;
7561  21842 +run;
7562  21843 +
7563  21844 +proc sort data=_clusRsq;
7564  21845 +    by cluster col1;
7565  21846 +    run;
7566  21847 +data _clusRsq(drop=_NAME_ _LABEL_);
7567  21848 +     set _clusRsq; by cluster;
7568  21849 +     if last.Cluster then output;
7569  21850 +     label  COL1 = 'R-Sqaure with Next Cluster Compo
      nent';
7570  21851 +     rename COL1 = RSqWithNextClusComp;
7571  21852 +     rename Cluster = Variable;
7572  21853 +     label  Cluster = "Variable";
7573  21854 +run;
7574  21855 +
7575  21856 +proc sort data =&outdata;
7576  21857 +     by Variable;
7577  21858 +run;
7578  21859 +data &outdata;
7579  21860 +    merge &outdata _clusRsq;
7580  21861 +    by Variable;
7581  21862 +run;
```

```
7582  21863 +proc sort data =&outdata;
7583  21864 +by Cluster  RsquareRatio;
7584  21865 +run;
7585  21866 +quit;
7586  21867 +%end;
7587  21868 +
7588  21869 +proc datasets lib = work nolist;
7589  21870 +     delete _tmprsq _clusRsq;
7590  21871 +     run;
7591  21872 +quit;
7592  21873 +
7593  21874 +%mend MakeRSquareData;
7594  21875 +
7595  21876 +
7596  21877 +/*-----------------------------------------------
      ----------------------*/
7597  21878 +
7598  21879 +
7599  21880 +
7600  21881 +%macro ModifyCorr(indata=,
7601  21882 +                   outdata=,
7602  21883 +                   rsquare = Y
7603  21884 +                   );
7604  21885 +    data corr_tmp;
7605  21886 +        set &indata;
7606  21887 +    run;
7607  21888 +     proc sql;
7608  21889 +           update &indata
7609  21890 +           set
7610  21891 +     %let dsid = %sysfunc(open(work.corr_tmp));
7611  21892 +     %let nvar = %sysfunc(attrn(&dsid, NVAR));
7612  21893 +          %do i = 4 %to &nvar;
7613  21894 +          %let _name = %sysfunc(varname(&dsid, &i));
7614  21895 +             %if &rsquare eq Y %then %let _name_md =
       &_name.**2;
7615  21896 +            %else  %let _name_md = &_name;
```

```
7616 21897 +            %if &i < &nvar %then %do;
7617 21898 +               &_name = 1- &_name_md ,
7618 21899 +            %end;
7619 21900 +            %else %do;
7620 21901 +               &_name = &_name_md where _TYPE_ conta
     ins 'CORR' ;
7621 21902 +            %end;
7622 21903 +          %end;
7623 21904 +      %let dsid= %sysfunc(close(&dsid));
7624 21905 +
7625 21906 +      select * from &indata;
7626 21907 +      run;
7627 21908 +      proc datasets lib = work nolist;
7628 21909 +           delete corr_tmp;
7629 21910 +      run;
7630 21911 +      quit;
7631 21912 +
7632 21913 + %mend ModifyCorr;
7633 21914 +
7634 21915 + %macro MakeClusStructCorrData(indata=,outdata=, ncl
     uster=, Rsquare=N);
7635 21916 +    data &outdata(drop= _NCL_  _TYPE_);
7636 21917 +       set &indata;
7637 21918 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
     E_) eq 'STRUCTUR') then delete;
7638 21919 +       rename _NAME_ = Cluster;
7639 21920 +       label _NAME_ = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_clustername,  noquote))";
7640 21921 +    run;
7641 21922 +    %if &RSquare eq Y %then %do;
7642 21923 +     data corr_tmp;
7643 21924 +        set &outdata;
7644 21925 +     run;
7645 21926 +
7646 21927 +     data &outdata(drop=i);
7647 21928 +          set &outdata;
```

```
7648 21929 +          %let dsid = %sysfunc(open(work.corr_tmp));
7649 21930 +          %let nvar = %sysfunc(attrn(&dsid, NVAR));
7650 21931 +          %do i = 2 %to &nvar;
7651 21932 +            %let _name = %sysfunc(varname(&dsid, &i)
     );
7652 21933 +            %let _name_md = &_name.**2;
7653 21934 +                &_name = &_name_md;
7654 21935 +          %end;
7655 21936 +       %let dsid= %sysfunc(close(&dsid));
7656 21937 +       run;
7657 21938 +       proc datasets lib = work nolist;
7658 21939 +            delete corr_tmp;
7659 21940 +       run;
7660 21941 +
7661 21942 +    %end;
7662 21943 +     quit;
7663 21944 +%mend MakeClusStructCorrData;
7664 21945 +
7665 21946 +%macro MakeInterClusCorrData(indata=, outdata=, nclu
     ster=, RSquare=N, makeplotds=N, plotds=);
7666 21947 +    data &outdata(drop= _NCL_  _TYPE_);
7667 21948 +       set &indata;
7668 21949 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
     E_) eq 'CCORR') then delete;
7669 21950 +       rename _NAME_ = Cluster;
7670 21951 +       label _NAME_ = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_clustername,  noquote))";
7671 21952 +    run;
7672 21953 +    data corr_tmp;
7673 21954 +        set &outdata;
7674 21955 +    run;
7675 21956 +
7676 21957 +    %let dsid = %sysfunc(open(work.corr_tmp));
7677 21958 +    %let nclus2= %eval(&ncluster+1);
7678 21959 +    data &outdata;
7679 21960 +         set &outdata;
```

```
7680  21961 +          %do i = 2 %to &nclus2;
7681  21962 +          %let i_1 = %eval(&i-1);
7682  21963 +            %let _name = %sysfunc(varname(&dsid, &i)
      );
7683  21964 +            %let _newName = Clus&i_1;
7684  21965 +                rename &_name = &_newName; ;
7685  21966 +                *label &_name ="Cluster &i_1";
7686  21967 +                label &_name = "%sysfunc(sasmsg(sash
      elp.dmine, rpt_varclus_label_clusternum, noquote,  &i_1))";
7687  21968 +          %end;
7688  21969 +          keep Cluster
7689  21970 +          %do i = 2 %to &nclus2;
7690  21971 +                %let _name = %sysfunc(varname(&dsid,
       &i));
7691  21972 +                &_name
7692  21973 +          %end;
7693  21974 +          ;
7694  21975 +     %let dsid= %sysfunc(close(&dsid));
7695  21976 +      run;
7696  21977 +      quit;
7697  21978 +
7698  21979 +     %if &RSquare eq Y %then %do;
7699  21980 +
7700  21981 +        data corr_tmp;
7701  21982 +         set &outdata;
7702  21983 +      run;
7703  21984 +
7704  21985 +      data &outdata(drop=i);
7705  21986 +           set &outdata;
7706  21987 +           %let dsid = %sysfunc(open(work.corr_tmp));
7707  21988 +           %let nvar = %sysfunc(attrn(&dsid, NVAR));
7708  21989 +           %do i = 2 %to &nvar;
7709  21990 +             %let _name = %sysfunc(varname(&dsid, &i)
      );
7710  21991 +             %let _name_md = &_name.**2;
7711  21992 +                 &_name = &_name_md;
```

```
7712  21993 +          %end;
7713  21994 +       %let dsid= %sysfunc(close(&dsid));
7714  21995 +       run;
7715  21996 +    %end;
7716  21997 +
7717  21998 +    %if &makeplotds eq Y %then %do;
7718  21999 +     proc transpose data = &outdata
7719  22000 +          out=&plotds(drop=_LABEL_ rename=(_NAME_ =
      Y Cluster=X Col1= Correlation));
7720  22001 +          by cluster;
7721  22002 +     run;
7722  22003 +     data &plotds;
7723  22004 +          set  &plotds;
7724  22005 +          label x="%sysfunc(sasmsg(sashelp.dmine, rp
      t_varclus_label_cluster, noquote))";
7725  22006 +          label Y="%sysfunc(sasmsg(sashelp.dmine, rp
      t_varclus_label_cluster, noquote))";
7726  22007 +     run;
7727  22008 +     %end;
7728  22009 +     proc datasets lib = work nolist;
7729  22010 +           delete corr_tmp;
7730  22011 +     run;
7731  22012 +     quit;
7732  22013 +%mend MakeInterClusCorrData;
7733  22014 +
7734  22015 +
7735  22016 +%macro MakeClusConstellData(indata=, outlink=, outno
      de=);
7736  22017 +
7737  22018 +data &outlink(drop = Selected);
7738  22019 +     set &indata;
7739  22020 +     LINKID = _N_;
7740  22021 +     label LINKID = "%sysfunc(sasmsg(sashelp.dmine,
      rpt_varclus_label_linkid,  noquote))";
7741  22022 +     if strip(upcase(Cluster)) eq strip(upcase(Varia
      ble)) then Variable = ClosestCluster;
```

```
7742 22023 +run;
7743 22024 +data &outnode(keep=NODEID TYPE LABEL);
7744 22025 +    set &indata;
7745 22026 +    length TYPE $16;
7746 22027 +    rename Variable = NODEID;
7747 22028 +    label  Variable= "%sysfunc(sasmsg(sashelp.dmine,
      rpt_varclus_label_nodeidvar, noquote))";
7748 22029 +    if strip(upcase(Cluster)) eq strip(upcase(Variab
     le))
7749 22030 +    then  TYPE = "CLUSTER";
7750 22031 +    else  TYPE="VARIABLE";
7751 22032 +    label TYPE = "%sysfunc(sasmsg(sashelp.dmine, rpt
     _varclus_label_nodetype, noquote))";
7752 22033 +run;
7753 22034 +quit;
7754 22035 +%mend MakeClusConstellData;
7755 22036 +
7756 22037 +
7757 22038 +
7758 22039 +%macro MakeClusConstellData(indata=, outlink=, outno
     de=);
7759 22040 +
7760 22041 +data &outlink(drop = Selected);
7761 22042 +     set &indata;
7762 22043 +     LINKID = _N_;
7763 22044 +     label LINKID = "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_linkid,  noquote))";
7764 22045 +     if strip(upcase(Cluster)) eq strip(upcase(Varia
     ble)) then Variable = ClosestCluster;
7765 22046 +run;
7766 22047 +data &outnode(keep=NODEID TYPE LABEL);
7767 22048 +    set &indata;
7768 22049 +    length TYPE $16;
7769 22050 +    rename Variable = NODEID;
7770 22051 +    label  Variable= "%sysfunc(sasmsg(sashelp.dmine,
      rpt_varclus_label_nodeidvar, noquote))";
```

```
7771 22052 +    if strip(upcase(Cluster)) eq strip(upcase(Variab
     le))
7772 22053 +    then  TYPE = "CLUSTER";
7773 22054 +    else  TYPE="VARIABLE";
7774 22055 +    label TYPE = "%sysfunc(sasmsg(sashelp.dmine, rpt
     _varclus_label_nodetype, noquote))";
7775 22056 +run;
7776 22057 +quit;
7777 22058 +%mend MakeClusConstellData;
7778 22059 +
7779 22060 +
7780 22061 +/*--- This will work only when inds is not a view da
     ta -------
7781 22062 +
7782 22063 +%macro getNVarNObs(inds=, nvar=, nobs=);
7783 22064 +
7784 22065 +    %global &nvar;
7785 22066 +    %global &nobs;
7786 22067 +    data _null_;
7787 22068 +         dsid = open("&inds");
7788 22069 +         nv = attrn(dsid, 'NVAR');
7789 22070 +         no = attrn(dsid, 'NOBS');
7790 22071 +         dsid = close(dsid);
7791 22072 +         call symput("&nvar", nv);
7792 22073 +         call symput("&nobs", no);
7793 22074 +    run;
7794 22075 +    quit;
7795 22076 +%mend  getNVarNObs;
7796 22077 +
7797 22078 ++-------------------------------------------------
     ------------*/
7798 22079 +
7799 22080 +
7800 22081 +%macro getNVar(inds=, nvar=);
7801 22082 +    %global &nvar;
7802 22083 +    data _null_;
```

```
7803  22084 +         dsid = open("&inds");
7804  22085 +         nv = attrn(dsid, 'NVAR');
7805  22086 +         dsid = close(dsid);
7806  22087 +         call symput("&nvar", nv);
7807  22088 +    run;
7808  22089 +    quit;
7809  22090 +%mend  getNVar;
7810  22091 +
7811  22092 +
7812  22093 +
7813  22094 +%macro getNObs(inds=, nobs=);
7814  22095 +    %global &nobs;
7815  22096 +    data _null_;
7816  22097 +        set &inds end=eof;
7817  22098 +        if eof then call symput("&nobs", _N_);
7818  22099 +    run;
7819  22100 +    quit;
7820  22101 +%mend  getNObs;
7821  22102 +
7822  22103 +%Macro CreateVarclusMeta(trainnum=);
7823  22104 +   %EM_GETNAME(KEY=VARCLUSMETA, TYPE=DATA);
7824  22105 +    data &EM_USER_VARCLUSMETA;
7825  22106 +         length TrainNum 8.;
7826  22107 +         length NewTrain $8;
7827  22108 +         length NGCluster 8.;
7828  22109 +         length ExportedComp $16;
7829  22110 +         length HideVariable $8;
7830  22111 +         TrainNum = &trainnum;
7831  22112 +         NewTrain = "Y";
7832  22113 +         ExportedComp = "&EM_PROPERTY_EXPORTEDCOMP";
7833  22114 +         HideVariable = "&EM_PROPERTY_HIDEVARIABLE";
7834  22115 +         NGCluster = 0; /* zero means no twostage */
7835  22116 +   run;
7836  22117 +   quit;
7837  22118 +%mend CreateVarclusMeta;
7838  NOTE: %INCLUDE (level 1) ending.
```

```
7839 NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMEXPL.V
     ARIABLECLUSTERING_MACROS2.SOURCE.
7840 22119 +
7841 22120 +%macro MakeInterClusCorrData(indata=, outdata=, nclu
     ster=, globalclusid=, RSquare=N, makeplotds=N, plotds=);
7842 22121 +    data &outdata(drop= _NCL_  _TYPE_);
7843 22122 +       set &indata;
7844 22123 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
     E_) eq 'CCORR') then delete;
7845 22124 +       %if &globalclusid ne %then %do;
7846 22125 +       _NAME_ = "GC&globalclusid._"||upcase(_NAME_);
7847 22126 +       rename _NAME_ = Cluster;
7848 22127 +       %end;
7849 22128 +       %else %do;
7850 22129 +        _NAME_ = upcase(_NAME_);
7851 22130 +       rename _NAME_ = Cluster;
7852 22131 +       %end;
7853 22132 +       label _NAME_ = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_clustername,  noquote))";
7854 22133 +    run;
7855 22134 +    data corr_tmp;
7856 22135 +        set &outdata;
7857 22136 +    run;
7858 22137 +
7859 22138 +    %let dsid = %sysfunc(open(work.corr_tmp));
7860 22139 +    %let nclus2= %eval(&ncluster+1);
7861 22140 +    data &outdata;
7862 22141 +         set &outdata;
7863 22142 +          %do i = 2 %to &nclus2;
7864 22143 +          %let i_1 = %eval(&i-1);
7865 22144 +            %let _name = %sysfunc(varname(&dsid, &i)
     );
7866 22145 +            %if &globalclusid ne %then
7867 22146 +                %do; %let _newName = GC&globalclusid
     ._CLUS&i_1;
7868 22147 +                    rename &_name = &_newName;
```

```
7869  22148 +                    *label &_name ="GC &globalclusi
      d : Cluster &i_1";
7870  22149 +                    label &_name = "%sysfunc(sasmsg
      (sashelp.dmine, rpt_varclus_label_gc_clusternum, noquote,
      &globalclusid, &i_1))";
7871  22150 +                %end;
7872  22151 +              %else
7873  22152 +                %do; %let _newName = CLUS&i_1;
7874  22153 +                     rename &_name = &_newName;
7875  22154 +                     *label &_name ="Cluster &i_1";
7876  22155 +                     label &_name = "%sysfunc(sasmsg
      (sashelp.dmine, rpt_varclus_label_clusternum, noquote,  &i_
      1))";
7877  22156 +                %end;
7878  22157 +           %end;
7879  22158 +           keep Cluster
7880  22159 +           %do i = 2 %to &nclus2;
7881  22160 +                %let _name = %sysfunc(varname(&dsid,
       &i));
7882  22161 +                &_name
7883  22162 +           %end;
7884  22163 +           ;
7885  22164 +    %let dsid= %sysfunc(close(&dsid));
7886  22165 +     run;
7887  22166 +     quit;
7888  22167 +
7889  22168 +    %if &RSquare eq Y %then %do;
7890  22169 +
7891  22170 +       data corr_tmp;
7892  22171 +        set &outdata;
7893  22172 +     run;
7894  22173 +
7895  22174 +     data &outdata;
7896  22175 +          set &outdata;
7897  22176 +          %let dsid = %sysfunc(open(work.corr_tmp));
7898  22177 +          %let nvar = %sysfunc(attrn(&dsid, NVAR));
```

```
7899  22178 +          %do i = 2 %to &nvar;
7900  22179 +            %let _name = %sysfunc(varname(&dsid, &i)
      );
7901  22180 +            %let _name_md = &_name.**2;
7902  22181 +                &_name = &_name_md;
7903  22182 +          %end;
7904  22183 +       %let dsid= %sysfunc(close(&dsid));
7905  22184 +       run;
7906  22185 +    %end;
7907  22186 +
7908  22187 +    %if &makeplotds eq Y %then %do;
7909  22188 +     proc transpose data = &outdata
7910  22189 +          out=&plotds(drop=_LABEL_ rename=(_NAME_ =
      Y Cluster=X Col1= Correlation));
7911  22190 +          by cluster;
7912  22191 +     run;
7913  22192 +     data &plotds;
7914  22193 +          set  &plotds;
7915  22194 +          label x="%sysfunc(sasmsg(sashelp.dmine, rp
      t_varclus_label_cluster, noquote))";
7916  22195 +          label Y="%sysfunc(sasmsg(sashelp.dmine, rp
      t_varclus_label_cluster, noquote))";
7917  22196 +          label  Correlation="%sysfunc(sasmsg(sashel
      p.dmine, rpt_correlation_vlabel, noquote))";
7918  22197 +     run;
7919  22198 +     %end;
7920  22199 +     proc datasets lib = work nolist;
7921  22200 +           delete corr_tmp;
7922  22201 +     run;
7923  22202 +     quit;
7924  22203 +%mend MakeInterClusCorrData;
7925  22204 +
7926  22205 +%macro MakeOwnRSquare(indata=, outdata=, ncluster=,
      globalclusid=);
7927  22206 +    data _tmpds(drop= _NCL_);
7928  22207 +       set &indata;
```

```
7929 22208 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
     E_) in ('GROUP','RSQUARED')) then delete;
7930 22209 +       %if &globalclusid ne %then %do;
7931 22210 +           _NAME_ = "GC&globalclusid.";
7932 22211 +       rename _NAME_ = Cluster;
7933 22212 +       %end;
7934 22213 +       %else %do;
7935 22214 +        _NAME_ = "CLUS";
7936 22215 +        rename _NAME_ = Cluster;
7937 22216 +       %end;
7938 22217 +       label _NAME_ = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_clustername,  noquote))";
7939 22218 +    run;
7940 22219 +    proc transpose data =_tmpds out =&outdata;
7941 22220 +    run;
7942 22221 +
7943 22222 +    data &outdata(drop=COL1);
7944 22223 +        %if &globalclusid ne %then %do;
7945 22224 +        length GCluster $16;
7946 22225 +        %end;
7947 22226 +        length Cluster $32;
7948 22227 +        length _NAME_ $32;
7949 22228 +         set &outdata;
7950 22229 +         _NAME_ = upcase(_NAME_);
7951 22230 +         rename _NAME_=Variable;
7952 22231 +         *label _NAME_="Variable";
7953 22232 +         label _NAME_ = "%sysfunc(sasmsg(sashelp.dmi
     ne, rpt_varclus_label_variable, noquote))" ;
7954 22233 +         label Cluster = "%sysfunc(sasmsg(sashelp.dm
     ine, rpt_varclus_label_cluster, noquote))" ;
7955 22234 +         label GCluster = "%sysfunc(sasmsg(sashelp.d
     mine, rpt_varclus_label_gcluster, noquote))" ;
7956 22235 +
7957 22236 +         %if &globalclusid ne %then %do;
7958 22237 +          GCluster = "GC&globalclusid";
7959 22238 +          Cluster = "GC&globalclusid._CLUS"||strip(C
```

```
     OL1);
7960 22239 +         %end;
7961 22240 +         %else %do;
7962 22241 +         Cluster = "CLUS"||strip(COL1);
7963 22242 +        %end;
7964 22243 +         rename COL2 = RSqWithOwnClusComp;
7965 22244 +         *label COL2 = "R-Square With Own Cluster Co
     mponent";
7966 22245 +         label COL2 = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_ownrsq, noquote))" ;
7967 22246 +
7968 22247 +   run;
7969 22248 +   proc sort data =&outdata;
7970 22249 +        by Cluster RSqWithOwnClusComp;
7971 22250 +   run;
7972 22251 +   proc datasets lib = work nolist;
7973 22252 +           delete _tmpds;
7974 22253 +   run;
7975 22254 +   quit;
7976 22255 +%mend MakeOwnRSquare;
7977 22256 +
7978 22257 +%macro MakeClusStructCorrData(indata=, outdata=, glo
     balclusid=, ncluster=, Rsquare=N);
7979 22258 +    data &outdata(drop= _NCL_  _TYPE_);
7980 22259 +       %if &globalclusid ne %then %do;
7981 22260 +        length GCluster $16;
7982 22261 +        %end;
7983 22262 +       set &indata;
7984 22263 +       if ^(strip(_NCL_) eq &ncluster and strip(_TYP
     E_) eq 'STRUCTUR') then delete;
7985 22264 +       %if &globalclusid ne %then %do;
7986 22265 +         GCluster = "GC&globalclusid";
7987 22266 +         _NAME_ = "GC&globalclusid._"||upcase(_NAME_
     );
7988 22267 +         rename _NAME_ = Cluster;
7989 22268 +         label _NAME_ = "%sysfunc(sasmsg(sashelp.dmi
```

```
     ne, rpt_varclus_label_cluster, noquote))" ;
7990 22269 +         label GCluster = "%sysfunc(sasmsg(sashelp.d
     mine, rpt_varclus_label_gcluster, noquote))" ;
7991 22270 +
7992 22271 +       %end;
7993 22272 +       %else %do;
7994 22273 +         _NAME_ = upcase(_NAME_);
7995 22274 +         rename _NAME_ = Cluster;
7996 22275 +         label _NAME_ = "%sysfunc(sasmsg(sashelp.dmi
     ne, rpt_varclus_label_cluster, noquote))" ;
7997 22276 +         label GCluster = "%sysfunc(sasmsg(sashelp.d
     mine, rpt_varclus_label_gcluster, noquote))" ;
7998 22277 +
7999 22278 +       %end;
8000 22279 +     run;
8001 22280 +    %if &RSquare eq Y %then %do;
8002 22281 +     data corr_tmp;
8003 22282 +          set &outdata;
8004 22283 +     run;
8005 22284 +     %let istart = 2;
8006 22285 +     %if &globalclusid ne %then %let istart = 3;
8007 22286 +     data &outdata;
8008 22287 +          set &outdata;
8009 22288 +          %let dsid = %sysfunc(open(work.corr_tmp));
8010 22289 +          %let nvar = %sysfunc(attrn(&dsid, NVAR));
8011 22290 +          %do i =&istart %to &nvar;
8012 22291 +            %let _name = %sysfunc(varname(&dsid, &i)
     );
8013 22292 +            %let _name_md = &_name.**2;
8014 22293 +                &_name = &_name_md;
8015 22294 +          %end;
8016 22295 +      %let dsid= %sysfunc(close(&dsid));
8017 22296 +      run;
8018 22297 +      proc datasets lib = work nolist;
8019 22298 +           delete corr_tmp;
8020 22299 +      run;
```

```
8021 22300 +    %end;
8022 22301 +     quit;
8023 22302 +%mend MakeClusStructCorrData;
8024 22303 +
8025 22304 +/*
8026 22305 +%MakeClusStructCorrData(indata=playpen._outstat, out
     data=_structrsq , ncluster=7, Rsquare=Y);
8027 22306 +*/
8028 22307 +
8029 22308 +%macro FindNextClosestClusByVar(indata=, outdata=, g
     lobalclusid=, ncluster=);
8030 22309 +
8031 22310 +     /* The indata should be the outdata
8032 22311 +        from %MakeClusStructCorrData(indata=, outdat
     a=, ); */
8033 22312 +
8034 22313 +     proc sort data =&indata out=_tmpclusRsq;
8035 22314 +     by cluster;
8036 22315 +     run;
8037 22316 +
8038 22317 +     proc transpose data =_tmpclusRsq out=_tmpclusRs
     q;
8039 22318 +      by cluster;
8040 22319 +     run;
8041 22320 +
8042 22321 +     proc sort data=_tmpclusRsq;
8043 22322 +        by _NAME_ COL1;
8044 22323 +     run;
8045 22324 +
8046 22325 +     data _tmpclusRsq;
8047 22326 +         length _NAME_ $32;
8048 22327 +         set _tmpclusRsq; by _NAME_;
8049 22328 +          _NAME_ = upcase(_NAME_);
8050 22329 +          %if &ncluster ne 1 %then %do;
8051 22330 +             if last._NAME_ then delete;
8052 22331 +          %end;
```

```
8053  22332 +          %else %do;
8054  22333 +             COL1 = 0;
8055  22334 +          %end;
8056  22335 +     run;
8057  22336 +     /* need to sort again */
8058  22337 +     proc sort data=_tmpclusRsq;
8059  22338 +        by _NAME_ COL1;
8060  22339 +     run;
8061  22340 +
8062  22341 +     data &outdata;
8063  22342 +         set _tmpclusRsq; by _NAME_;
8064  22343 +         Cluster = upcase(Cluster);
8065  22344 +         if last._NAME_ then output;
8066  22345 +         *label  COL1 = 'R-Sqaure with Next Cluster
      Component';
8067  22346 +         label COL1 = "%sysfunc(sasmsg(sashelp.dmine
      , rpt_varclus_label_nextrsq, noquote))" ;
8068  22347 +         rename COL1 = RSqWithNextClusComp;
8069  22348 +         Cluster = upcase(Cluster);
8070  22349 +         rename Cluster = ClosestCluster;
8071  22350 +         *label  Cluster = "Next Closest Cluster";
8072  22351 +         label Cluster = "%sysfunc(sasmsg(sashelp.dm
      ine, rpt_varclus_label_nextclus, noquote))" ;
8073  22352 +         rename _NAME_ = Variable;
8074  22353 +         label  _NAME_ = "%sysfunc(sasmsg(sashelp.dm
      ine, rpt_varclus_label_variable, noquote))" ;
8075  22354 +    run;
8076  22355 +
8077  22356 +
8078  22357 +    %if &globalclusid ne %then %do;
8079  22358 +    data &outdata;
8080  22359 +       length GCluster $16;
8081  22360 +       set &outdata;
8082  22361 +       GCluster = "GC&globalclusid";
8083  22362 +       run;
8084  22363 +    %end;
```

```
8085  22364 +    proc datasets lib = work nolist;
8086  22365 +           delete _tmpclusRsq;
8087  22366 +    run;
8088  22367 +    quit;
8089  22368 +%mend FindNextClosestClusByVar;
8090  22369 +
8091  22370 +
8092  22371 +%macro FindNextClosestClusByCluster(indata=, outdata
      =, globalclusid=, ncluster=);
8093  22372 +     /* The indata should be the outdata from %MakeI
      nterClusCorrData(indata=, outdata=, ); */
8094  22373 +     proc sort data =&indata out=_tmpclusRsq;
8095  22374 +     by cluster;
8096  22375 +     run;
8097  22376 +     proc transpose data =_tmpclusRsq out=_tmpclusRs
      q;
8098  22377 +     by cluster;
8099  22378 +     run;
8100  22379 +     proc sort data=_tmpclusRsq;
8101  22380 +        by _NAME_ col1;
8102  22381 +     run;
8103  22382 +     data _tmpclusRsq;
8104  22383 +         length _NAME_ $32;
8105  22384 +         set _tmpclusRsq; by _NAME_;
8106  22385 +          _NAME_ = upcase(_NAME_);
8107  22386 +          %if &ncluster ne 1 %then %do;
8108  22387 +             if last._NAME_ then delete;
8109  22388 +          %end;
8110  22389 +          %else %do;
8111  22390 +             COL1 = 0;
8112  22391 +          %end;
8113  22392 +     run;
8114  22393 +     data &outdata;
8115  22394 +         set _tmpclusRsq; by _NAME_;
8116  22395 +         Cluster = upcase(Cluster);
8117  22396 +         if last._NAME_ then output;
```

```
8118 22397 +         *label  COL1 = 'R-Sqaure with Next Cluster
     Component';
8119 22398 +         label COL1 = "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_nextrsq, noquote))" ;
8120 22399 +         rename COL1 = RSqWithNextClusComp;
8121 22400 +         Cluster = upcase(Cluster);
8122 22401 +         rename Cluster = ClosestCluster;
8123 22402 +         *label  Cluster = "Next Closest Cluster";
8124 22403 +         label Cluster = "%sysfunc(sasmsg(sashelp.dm
     ine, rpt_varclus_label_nextclus, noquote))" ;
8125 22404 +         rename _NAME_ = Variable;
8126 22405 +         *label  _NAME_ = "Variable";
8127 22406 +         label  _NAME_ = "%sysfunc(sasmsg(sashelp.dm
     ine, rpt_varclus_label_variable, noquote))" ;
8128 22407 +
8129 22408 +    run;
8130 22409 +    %if &globalclusid ne %then %do;
8131 22410 +    data &outdata;
8132 22411 +       length GCluster $16;
8133 22412 +       set &outdata;
8134 22413 +       GCluster = "GC&globalclusid";
8135 22414 +       run;
8136 22415 +    %end;
8137 22416 +
8138 22417 +    proc datasets lib = work nolist;
8139 22418 +           delete _tmpclusRsq;
8140 22419 +    run;
8141 22420 +
8142 22421 +    quit;
8143 22422 +%mend FindNextClosestClusByCluster;
8144 22423 +
8145 22424 +%macro MakeVarClusResultTable(indata1=, indata2=, in
     data3=, outdata=, globalclusid=, ncluster=, selectedcomp=cl
     ustercomp);
8146 22425 +/*----
8147 22426 +  indata1=_ownRsq, indata2=_nextVarRsq, indata3=_nex
```

```
     tClusRSq,
8148 22427 +-----------*/
8149 22428 +
8150 22429 +proc sort data =&indata1;
8151 22430 +    by Variable;
8152 22431 +run;
8153 22432 +proc sort data =&indata2;
8154 22433 +    by Variable;
8155 22434 +run;
8156 22435 +data &outdata;
8157 22436 +    merge &indata1 &indata2;
8158 22437 +    by Variable;
8159 22438 +    length Type $16;
8160 22439 +    Type = 'Variable';
8161 22440 +    *label Type ='Type';
8162 22441 +    label Type = "%sysfunc(sasmsg(sashelp.dmine, rpt
     _varclus_label_type, noquote))" ;
8163 22442 +run;
8164 22443 +
8165 22444 +
8166 22445 +data &indata3;
8167 22446 +    set  &indata3 ;
8168 22447 +    length RSqWithOwnClusComp 8.;
8169 22448 +    Cluster = Variable;
8170 22449 +    RSqWithOwnClusComp = 1;
8171 22450 +    *label RSqWithOwnClusComp = "R-Square With Own C
     luster Component";
8172 22451 +    label RSqWithOwnClusComp = "%sysfunc(sasmsg(sash
     elp.dmine, rpt_varclus_label_ownrsq, noquote))" ;
8173 22452 +    length Type $16;
8174 22453 +    Type = 'ClusterComp';
8175 22454 +    label Type = "%sysfunc(sasmsg(sashelp.dmine, rpt
     _varclus_label_type, noquote))" ;
8176 22455 +
8177 22456 +;
8178 22457 +run;
```

```
8179  22458 +
8180  22459 +proc sort data=&outdata;
8181  22460 +    by Cluster;
8182  22461 +run;
8183  22462 +proc sort data =&indata3;
8184  22463 +    by Cluster;
8185  22464 +run;
8186  22465 +
8187  22466 +data &outdata;
8188  22467 +     set &outdata &indata3;
8189  22468 +     by Cluster;
8190  22469 +run;
8191  22470 +
8192  22471 +
8193  22472 +/* Create the Selected variable with all YES */
8194  22473 +
8195  22474 +data &outdata;
8196  22475 +     set &outdata;
8197  22476 +     length RsqRatio 8.;
8198  22477 +     length Selected $8;
8199  22478 +     *label RSqRatio = "1-R**2 Ratio";
8200  22479 +     label RSqRatio =  "%sysfunc(sasmsg(sashelp.dmin
      e, rpt_varclus_label_oneminusrsq, noquote))" ;
8201  22480 +     *label Selected = "Variable Selected";
8202  22481 +     label Selected = "%sysfunc(sasmsg(sashelp.dmine
      , rpt_varclus_label_varselected, noquote))" ;
8203  22482 +     RsqRatio = (1-RSqWithOwnClusComp)/(1-RSqWithNex
      tClusComp);
8204  22483 +     Selected ='YES';
8205  22484 +     rename _LABEL_ = Label;
8206  22485 +     label _LABEL_ =  "%sysfunc(sasmsg(sashelp.dmine
      , rpt_varclus_label_label, noquote))" ;
8207  22486 +run;
8208  22487 +
8209  22488 +
8210  22489 +/*----  Selected = Y/N will be assigened at the %sco
```

```
     re                    -------+
8211 22490 +        Just create the Selected variable with all Y
     ES at the step above
8212 22491 + +-----------------------------------------------
     ----------------------------+
8213 22492 +
8214 22493 +proc sort data=&outdata;
8215 22494 +    by Cluster RsqRatio;
8216 22495 +run;
8217 22496 +
8218 22497 +%if &selectedcomp eq CLUSTERCOMP %then %do;
8219 22498 +data &outdata;
8220 22499 +     set &outdata; by Cluster;
8221 22500 +     length Selected $8;
8222 22501 +     label Selected = "Variable Selected";
8223 22502 +     if  first.Cluster then Selected ='Yes';
8224 22503 +     else Selected = 'No';
8225 22504 +    run;
8226 22505 +%end;
8227 22506 +%else %do;
8228 22507 +data &outdata(drop = _var _varchange);
8229 22508 +     set &outdata; retain _var 0; by Cluster;
8230 22509 +     length Selected $8;
8231 22510 +     label Selected = "Variable Selected";
8232 22511 +     if first.Cluster then  _varchange = 0;
8233 22512 +     else _varchange =1;
8234 22513 +     if _var ne _varchange then  Selected  = 'Yes';
8235 22514 +     else Selected = 'No';
8236 22515 +     if last.cluster then  _var = 0;
8237 22516 +     else _var = _varchange;
8238 22517 +run;
8239 22518 +%end;
8240 22519 +
8241 22520 +-----------------------------------------------
     ----------*/
8242 22521 +
```

```
8243  22522 +quit;
8244  22523 +%mend MakeVarClusResultTable;
8245  22524 +
8246  22525 +%Macro MakePlotDataFromCorrTable(indata=, outdata=,
      globalclusid=);
8247  22526 +     proc sort data =&indata;
8248  22527 +        by cluster;
8249  22528 +     run;
8250  22529 +     proc transpose data =&indata
8251  22530 +          out=&outdata(drop=_LABEL_ rename=(_NAME_ =
       Y Cluster=X Col1= Correlation));
8252  22531 +          by cluster;
8253  22532 +     run;
8254  22533 +     data &outdata;
8255  22534 +          set  &outdata;
8256  22535 +          label x= "%sysfunc(sasmsg(sashelp.dmine, r
      pt_varclus_label_cluster, noquote))" ;
8257  22536 +          label Y= "%sysfunc(sasmsg(sashelp.dmine, r
      pt_varclus_label_cluster, noquote))" ;
8258  22537 +          label Correlation = "%sysfunc(sasmsg(sashe
      lp.dmine, rpt_correlation_vlabel, noquote))" ;
8259  22538 +     run;
8260  22539 +     %if &globalclusid ne %then %do;
8261  22540 +     data &outdata;
8262  22541 +          Length GCluster $16;
8263  22542 +          label GCluster = "%sysfunc(sasmsg(sashelp.
      dmine, rpt_varclus_label_gcluster, noquote))" ;
8264  22543 +          set  &outdata;
8265  22544 +          GCluster ="GC&globalclusid.";
8266  22545 +          run;
8267  22546 +     %end;
8268  22547 +
8269  22548 +%Mend MakePlotDataFromCorrTable;
8270  22549 +
8271  22550 +
8272  22551 +%macro MakeCorrelation(indata=,
```

```
8273  22552 +                        outstat=_tmpoutstat,
8274  22553 +                        corrmatrix=N,
8275  22554 +                        outcorr=_tmpoutcorr,
8276  22555 +                        includeclassvar=N,
8277  22556 +                        target=,
8278  22557 +                        freq=,
8279  22558 +                        weight=);
8280  22559 +   %if &target eq  %then %do;
8281  22560 +
8282  22561 +       proc varclus data=&indata outstat=&outstat hi
       maxclusters=1 noprint;
8283  22562 +            var %EM_INTERVAL_INPUT %EM_INTERVAL_REJE
      CTED
8284  22563 +       %if &includeclassvar eq Y %then %do;
8285  22564 +          %let dsid = %sysfunc(open(&EM_USER_OUTDUMM
      Y));
8286  22565 +          %let nvar = %sysfunc(attrn(&dsid, NVAR));
8287  22566 +          %do i = 2 %to &nvar;
8288  22567 +          %let varname = %sysfunc(varname(&dsid, &i)
      );
8289  22568 +          &varname
8290  22569 +          %end;
8291  22570 +       %end;
8292  22571 +       ;
8293  22572 +      %if &freq ne  %then %do;
8294  22573 +          freq &freq;
8295  22574 +      %end;
8296  22575 +      %if &weight ne  %then %do;
8297  22576 +          weight &weight;
8298  22577 +      %end;
8299  22578 +
8300  22579 +      run;
8301  22580 +      %if &corrmatrix eq Y %then %do;
8302  22581 +      data  &outcorr (drop = _NCL_ _TYPE_);
8303  22582 +           set  &outstat ;
8304  22583 +           if _TYPE_ ='CORR' then output;
```

```
8305  22584 +      run;
8306  22585 +      %end;
8307  22586 +    %end;
8308  22587 +    %else %do;
8309  22588 +       proc corr data=&indata outp=&outstat noprint;
8310  22589 +            var
8311  22590 +       %let dsid = %sysfunc(open(&indata));
8312  22591 +       %let nvar = %sysfunc(attrn(&dsid, NVAR));
8313  22592 +          %do i = 1 %to &nvar;
8314  22593 +              %let _name = %sysfunc(varname(&dsid, &
      i));
8315  22594 +              %if &_name ne &target %then;
8316  22595 +              &_name
8317  22596 +           %end;
8318  22597 +        %let dsid= %sysfunc(close(&dsid));
8319  22598 +        ;
8320  22599 +        with &target;
8321  22600 +        run;
8322  22601 +    %end;
8323  22602 +    quit;
8324  22603 +%mend MakeCorrelation;
8325  22604 +
8326  22605 +
8327  22606 +%macro MakeCorrelationDistance(indata=,
8328  22607 +                                outdata=,
8329  22608 +                                rsquare = N
8330  22609 +                                );
8331  22610 +    data corr_tmp;
8332  22611 +        set &indata;
8333  22612 +        if _N_ = 1 then do;
8334  22613 +           output;
8335  22614 +           stop;
8336  22615 +         end;
8337  22616 +    run;
8338  22617 +    %if &outdata ne  %then %let  _outdata = &outdata
      ;
```

```
8339 22618 +    %else %let _outdata = &indata;
8340 22619 +
8341 22620 +    data &_outdata;
8342 22621 +         set &indata;
8343 22622 +
8344 22623 +         %let dsid = %sysfunc(open(work.corr_tmp));
8345 22624 +         %let nvar = %sysfunc(attrn(&dsid, NVAR));
8346 22625 +          %do i = 2 %to &nvar;
8347 22626 +              %let _name = %sysfunc(varname(&dsid, &
     i));
8348 22627 +              %if &rsquare eq Y %then %let _name_md
     = &_name.**2;
8349 22628 +              %else  %let _name_md = &_name;
8350 22629 +              &_name = 1- &_name_md;
8351 22630 +          %end;
8352 22631 +      %let dsid= %sysfunc(close(&dsid));
8353 22632 +      run;
8354 22633 +      proc datasets lib = work nolist;
8355 22634 +           delete corr_tmp;
8356 22635 +      run;
8357 22636 +      quit;
8358 22637 +  %mend MakeCorrelationDistance;
8359 22638 +
8360 22639 +
8361 22640 +%macro UpdateOutStatCorrToDistance(indata=, /* indat
     a should be a outstat from proc varclus */
8362 22641 +                                   rsquare = N
8363 22642 +                                   );
8364 22643 +    data corr_tmp;
8365 22644 +        set &indata;
8366 22645 +    run;
8367 22646 +    proc sql noprint;
8368 22647 +           update &indata
8369 22648 +           set
8370 22649 +     %let dsid = %sysfunc(open(work.corr_tmp));
8371 22650 +     %let nvar = %sysfunc(attrn(&dsid, NVAR));
```

```
8372  22651 +          %do i = 4 %to &nvar;
8373  22652 +          %let _name = %sysfunc(varname(&dsid, &i));
8374  22653 +             %if &rsquare eq Y %then %let _name_md =
       &_name.**2;
8375  22654 +            %else  %let _name_md = &_name;
8376  22655 +            %if &i < &nvar %then %do;
8377  22656 +               &_name = 1- &_name_md ,
8378  22657 +            %end;
8379  22658 +            %else %do;
8380  22659 +               &_name = &_name_md where _TYPE_ eq 'C
      ORR' ;
8381  22660 +            %end;
8382  22661 +          %end;
8383  22662 +       %let dsid= %sysfunc(close(&dsid));
8384  22663 +
8385  22664 +       select * from &indata;
8386  22665 +       run;
8387  22666 +     data &indata( drop = _NCL_);
8388  22667 +          set &indata;
8389  22668 +          if _TYPE_ not in ('CORR', 'STD', 'N', 'ME
      AN') then delete;
8390  22669 +          if _TYPE_ ='CORR' then _TYPE_ ='DISTANCE'
      ;
8391  22670 +     run;
8392  22671 +     data &indata(DROP = _NCL_);
8393  22672 +          set &indata;
8394  22673 +          if _TYPE_ = 'CORR' then _TYPE_ ='DISTANCE
      ';
8395  22674 +          if _TYPE_ not in ('DISTANCE', 'N', 'STD',
       'MEAN') then delete;
8396  22675 +          rename _NAME_ = _VAR_;
8397  22676 +     run;
8398  22677 +     proc datasets lib = work nolist;
8399  22678 +          delete corr_tmp;
8400  22679 +     run;
8401  22680 +     quit;
```

```
8402  22681 + %mend UpdateOutStatCorrToDistance;
8403  22682 +
8404  22683 +
8405  22684 +%macro HierClusWithCorr(indata= ,
8406  22685 +                         ncluster=,
8407  22686 +                         method = Ward,
8408  22687 +                         outtree = _outtree,
8409  22688 +                         idvar =_VAR_,
8410  22689 +                         outdata=,
8411  22690 +                         rescore = N,
8412  22691 +                         newncluster=
8413  22692 +                         );
8414  22693 +      %global &newncluster;
8415  22694 +      %if &rescore ne Y %then %do;
8416  22695 +      proc cluster data=&indata(type=Distance where=
      (upcase(strip(_TYPE_)) = "DISTANCE"))
8417  22696 +                   method=&method outtree=&outtree n
      oprint;
8418  22697 +           id &idvar;
8419  22698 +      run;
8420  22699 +      %end;
8421  22700 +      proc tree data=&outtree nclusters = &ncluster
      out=&outdata noprint;
8422  22701 +      run;
8423  22702 +      /* ----- Check some variables like CL1, CL5...
      , remove them ----*/
8424  22703 +      proc contents data =&indata out=_outcontent(ke
      ep=NAME) noprint;
8425  22704 +      run;
8426  22705 +      data _outcontent;
8427  22706 +          set _outcontent;
8428  22707 +          if NAME in ('_TYPE_' , '_VAR_') then delet
      e;
8429  22708 +          index = 1;
8430  22709 +          rename NAME = _NAME_;
8431  22710 +      run;
```

```
8432  22711 +      proc sort data=_outcontent;
8433  22712 +          by _NAME_;
8434  22713 +      run;
8435  22714 +      proc sort data =&outdata;
8436  22715 +           by _NAME_;
8437  22716 +      run;
8438  22717 +      data &outdata(drop=index);
8439  22718 +          merge &outdata _outcontent;
8440  22719 +          by _NAME_;
8441  22720 +          if index = . then delete;
8442  22721 +      run;
8443  22722 +      /*-----------------------------------------
      ---------------*/
8444  22723 +      data &outdata;
8445  22724 +           length CLUSNAME $16;
8446  22725 +           set &outdata;
8447  22726 +           if CLUSTER > &ncluster then delete;
8448  22727 +           CLUSNAME='GC'||strip(CLUSTER);
8449  22728 +           *label CLUSNAME = "Cluster Name";
8450  22729 +           label CLUSNAME  = "%sysfunc(sasmsg(sashel
      p.dmine, rpt_varclus_label_clustername, noquote))" ;
8451  22730 +           rename _NAME_ = VARIABLE ;
8452  22731 +           *label _NAME_ = "Variable";
8453  22732 +           *label CLUSTER = "Cluster";
8454  22733 +           label _NAME_ ="%sysfunc(sasmsg(sashelp.dm
      ine, rpt_varclus_label_variable, noquote))" ;
8455  22734 +           label CLUSTER ="%sysfunc(sasmsg(sashelp.d
      mine, rpt_varclus_label_cluster, noquote))" ;
8456  22735 +      run;
8457  22736 +      proc sort data=&outdata out=&outdata;
8458  22737 +           by CLUSTER;
8459  22738 +      run;
8460  22739 +      proc means data =&outdata noprint;
8461  22740 +           output out=_meanout;
8462  22741 +      run;
8463  22742 +      data _null_;
```

```
8464 22743 +            set _meanout;
8465 22744 +            if strip(_STAT_) eq 'MAX' then do;
8466 22745 +            call symput("&newncluster", CLUSTER);
8467 22746 +            stop;
8468 22747 +            end;
8469 22748 +      run;
8470 22749 +
8471 22750 +      proc datasets lib = work nolist;
8472 22751 +           delete _outcontent _meanout;
8473 22752 +      run;
8474 22753 +      quit;
8475 22754 +%mend  HierClusWithCorr;
8476 22755 +
8477 22756 +%macro CreateScoreCode2(indata=, ncluscomp=, globalc
     lusid=, fileref=);
8478 22757 +
8479 22758 +     data _tmpindata;
8480 22759 +          set &indata;
8481 22760 +          if (_TYPE_ in ('SCORE' 'MEAN' 'STD') and _
     NCL_ = &ncluscomp ) or (_TYPE_ in ('MEAN' 'STD'));
8482 22761 +          if _TYPE_ = 'MEAN' then _NAME_='MEAN';
8483 22762 +          if _TYPE_ = 'STD' then _NAME_='STD';
8484 22763 +          if _TYPE_ = 'SCORE' then _NAME_=upcase("GC
     &globalclusid._"||_NAME_);
8485 22764 +          DROP _TYPE_ _NCL_;
8486 22765 +     run;
8487 22766 +
8488 22767 +     filename _file_  "&fileRef";
8489 22768 +
8490 22769 +     data _null_;
8491 22770 +        FILE _file_ MOD;
8492 22771 +        put ' ';
8493 22772 +        put "/*-------------------------------------
     ------------*/";
8494 22773 +        put '/* ' "%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_score_title_gclus, noquote, &globalclusid))" '*/'
```

```
     ;
8495 22774 +        put "/*-----------------------------------
     ------------*/";
8496 22775 +        put ' ';
8497 22776 +        %let dsid = %sysfunc(open(work._tmpindata));
8498 22777 +
8499 22778 +        %let nvar = %sysfunc(attrn(&dsid, NVAR));
8500 22779 +        %let vn_name =%sysfunc(varnum(&dsid, _NAME_)
     );
8501 22780 +        %let k = 1;
8502 22781 +        %do %while(^%sysfunc(fetch(&dsid)));
8503 22782 +               %let _name = %sysfunc(getvarc(&dsid,
     &vn_name));
8504 22783 +               %if &k > 2 %then %do;
8505 22784 +                %let cn = %eval(&k-2);
8506 22785 +                 put "&_name = 0 ; /*---" "%sysfunc(
     sasmsg(sashelp.dmine, rpt_varclus_score_gcluscompnum, noquo
     te, &globalclusid, &cn))" "------ */";
8507 22786 +                %end;
8508 22787 +                %let k = %eval(&k+1);
8509 22788 +
8510 22789 +        %end;
8511 22790 +        %let rc = %sysfunc(rewind(&dsid));
8512 22791 +
8513 22792 +        %do j= 2 %to &nvar;
8514 22793 +            %let _varname = %sysfunc(varname(&dsid,
     &j));
8515 22794 +            %do %while(^%sysfunc(fetch(&dsid)));
8516 22795 +                %let _name = %sysfunc(getvarc(&dsid,
      &vn_name));
8517 22796 +                %if &_name = MEAN %then
8518 22797 +                %let _mean = %sysfunc(getvarn(&dsid,
      &j));
8519 22798 +                %else %if &_name = STD %then
8520 22799 +                %let _std = %sysfunc(getvarn(&dsid,
     &j));
```

```
8521 22800 +                %else %do;
8522 22801 +                      %let coeff =  %sysfunc(getvarn
     (&dsid, &j));
8523 22802 +                      %let abscoeff = %sysfunc(abs(&
     coeff));
8524 22803 +                          %if &abscoeff >  0 %then %
     do;
8525 22804 +                       put "&_name = &_name+&coeff *
      (&_varname - &_mean)/&_std;";
8526 22805 +                           %end;
8527 22806 +                %end;
8528 22807 +             %end;
8529 22808 +             %let rc = %sysfunc(rewind(&dsid));
8530 22809 +         %end;
8531 22810 +
8532 22811 +        %let dsid= %sysfunc(close(&dsid));
8533 22812 +
8534 22813 +     run;
8535 22814 +
8536 22815 +     filename _file_;
8537 22816 +     proc datasets lib = work nolist;
8538 22817 +           delete _tmpindata;
8539 22818 +     run;
8540 22819 +     quit;
8541 22820 +%mend CreateScoreCode2;
8542 22821 +
8543 22822 +
8544 22823 +%macro MakeDeltaCode2(groupds=,deltacodefile=);
8545 22824 +
8546 22825 +     /*--- Build Code to Modify Metadata ---*/
8547 22826 +     filename X "&deltacodefile";
8548 22827 +     data _null_;
8549 22828 +        FILE X;
8550 22829 +        set &groupds end=eof;
8551 22830 +        if _N_=1 then do;
8552 22831 +           %if &EM_PROPERTY_INCLUDECLASSVAR eq Y %th
```

```
en %do;
8553 22832 +            put "if upcase(strip(ROLE)) ='INPUT' and
      upcase(strip(LEVEL)) ^='INTERVAL' then ROLE ='REJECTED' ;"
     ;
8554 22833 +           %end;
8555 22834 +           put "if upcase(strip(ROLE))='INPUT' and u
     pcase(strip(LEVEL))='INTERVAL' then do;";
8556 22835 +           put "if upcase(strip(NAME)) in (";
8557 22836 +        end;
8558 22837 +        if Strip(upcase(Selected)) eq 'YES' then do;
8559 22838 +           string = '"'!!trim(left(VARIABLE))!!'"';
8560 22839 +           put string;
8561 22840 +        end;
8562 22841 +        if eof then do;
8563 22842 +           put ') then ROLE="INPUT";';
8564 22843 +           put 'else ROLE="REJECTED";';
8565 22844 +           put 'end;';
8566 22845 +
8567 22846 +           %if %upcase(&EM_PROPERTY_HIDEVARIABLE) eq
      Y %then %do;
8568 22847 +             put 'if upcase(strip(ROLE)) = "REJECTED
     " then delete ;';
8569 22848 +           %end;
8570 22849 +        end;
8571 22850 +     run;
8572 22851 +     quit;
8573 22852 +
8574 22853 +     filename X;
8575 22854 +     quit;
8576 22855 +%mend MakeDeltaCode2;
8577 22856 +
8578 22857 +%macro getInitialGClusterNumber(indata=, ninput=, nd
     ummy=0, div=100, ngc=);
8579 22858 +  %global &ngc;
8580 22859 +  data _null_;
8581 22860 +  %if &indata ne %then %do;
```

```
8582 22861 +      %let dsid = %sysfunc(open(&indata));
8583 22862 +           %let nvar = %sysfunc(attrn(&dsid, NVAR));
8584 22863 +      %let dsid = %sysfunc(close(&dsid));
8585 22864 +  %end;
8586 22865 +  %else %do;
8587 22866 +       %let nvar = %eval(&ninput+&ndummy); ;
8588 22867 +  %end;
8589 22868 +  %let numgc = %eval(&nvar/&div+2);
8590 22869 +  %let &ngc = &numgc;
8591 22870 +   run;
8592 22871 +  quit;
8593 22872 +%mend getInitialGClusterNumber;
8594 22873 +
8595 22874 +
8596 22875 +%macro MakeGobalConstellData(indata=, outlink=, outn
     ode=);
8597 22876 +data &outlink(drop = Selected);
8598 22877 +     set &indata;
8599 22878 +     LINKID = _N_;
8600 22879 +     label LINKID = "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_linkid, noquote))" ;
8601 22880 +run;
8602 22881 +data &outnode(keep=NODEID TYPE LABEL);
8603 22882 +    set &indata;
8604 22883 +    length TYPE $16;
8605 22884 +    rename VARIABLE = NODEID;
8606 22885 +    *label  CLUSNAME="Node ID";
8607 22886 +    label CLUSNAME= "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_nodeidvar, noquote))" ;
8608 22887 +    TYPE = "VARIABLE";
8609 22888 +    *label TYPE = "Node Type";
8610 22889 +    label TYPE =  "%sysfunc(sasmsg(sashelp.dmine, rp
     t_varclus_label_nodetype, noquote))" ;
8611 22890 +    run;
8612 22891 +data _tmp(keep=NODEID TYPE LABEL);
8613 22892 +    set &indata;
```

```
8614 22893 +    length TYPE $16;
8615 22894 +    rename CLUSNAME = NODEID;
8616 22895 +    label  CLUSNAME= "%sysfunc(sasmsg(sashelp.dmine,
      rpt_varclus_label_nodeidvar, noquote))" ;
8617 22896 +    TYPE = "GCLUSTER";
8618 22897 +    label TYPE = "%sysfunc(sasmsg(sashelp.dmine, rpt
     _varclus_label_nodetype, noquote))" ;
8619 22898 +    run;
8620 22899 +proc sort data=_tmp;
8621 22900 +    by NODEID;
8622 22901 +run;
8623 22902 +data _tmp;
8624 22903 +    set _tmp; by NODEID;
8625 22904 +    if first.NODEID then output;
8626 22905 +run;
8627 22906 +proc sort data=&outnode;
8628 22907 +    by NODEID;
8629 22908 +run;
8630 22909 +data  &outnode;
8631 22910 +    set _tmp &outnode;
8632 22911 +run;
8633 22912 +proc datasets lib = work nolist;
8634 22913 +     delete _tmp;
8635 22914 +run;
8636 22915 +quit;
8637 22916 +%mend MakeGobalConstellData;
8638 22917 +
8639 22918 +/* Make contellation plot data among GCLUSTERS */
8640 22919 +
8641 22920 +%Macro MakeGClusterConstData(indata=, inoutrsq=, out
     node=, outlink=);
8642 22921 +
8643 22922 +data &outlink(keep = _NAME_ _PARENT_ _LABEL_ LINKID)
     ;
8644 22923 +     set &indata;
8645 22924 +     LINKID = _N_;
```

```
8646 22925 +     if upcase(substr(strip(_NAME_),1, 2))="CL" then
      do;
8647 22926 +        _NAME_ = "ROOT"||upcase(substr(strip(_NAME_)
     ,5));
8648 22927 +     end;
8649 22928 +     if _PARENT_ ne " " and upcase(substr(strip(_PAR
     ENT_),1, 2))="CL" then do;
8650 22929 +        _PARENT_ = "ROOT"||upcase(substr(strip(_PARE
     NT_),5));
8651 22930 +     end;
8652 22931 +     if upcase(substr(strip(_LABEL_),1, 2))="CL" the
     n do;
8653 22932 +        _LABEL_ = "ROOT"||upcase(substr(strip(_LABEL
     _),5));
8654 22933 +    end;
8655 22934 +run;
8656 22935 +
8657 22936 +data _tmp_outrsquare;
8658 22937 +        set &inoutrsq;
8659 22938 +        if upcase(strip(TYPE)) = 'CLUSTERCOMP' then
     delete;
8660 22939 +run;
8661 22940 +
8662 22941 +proc freq data =_tmp_outrsquare noprint;
8663 22942 +         tables GCluster/out=_tmp_GCLUSFREQ(rename=(
     GCLUSTER=_NAME_));
8664 22943 +run;
8665 22944 +
8666 22945 +data &outnode(keep=_NAME_ TYPE LABEL);
8667 22946 +    set &outlink;
8668 22947 +    length TYPE $16;
8669 22948 +    length LABEL $100;
8670 22949 +    /*label CLUSNAME= "%sysfunc(sasmsg(sashelp.dmine
     , rpt_varclus_label_nodeidvar, noquote))" ;*/
8671 22950 +
8672 22951 +    if upcase(substr(strip(_NAME_),1, 2))='GC' then
```

```
     do;
8673 22952 +    TYPE = "GCLUSTER";
8674 22953 +    LABEL = "%sysfunc(sasmsg(sashelp.dmine, rpt_varc
     lus_label_gcluster, noquote)):"||_NAME_;
8675 22954 +    end;
8676 22955 +    else do;
8677 22956 +    TYPE= "ROOT";
8678 22957 +    LABEL= _NAME_;
8679 22958 +    end;
8680 22959 +    label TYPE  =  "%sysfunc(sasmsg(sashelp.dmine, r
     pt_varclus_label_nodetype, noquote))" ;
8681 22960 +    label LABEL =  "%sysfunc(sasmsg(sashelp.dmine, r
     pt_varclus_label_label, noquote))" ;
8682 22961 +    label _NAME_ =  "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_nodeidvar, noquote))" ;
8683 22962 +    run;
8684 22963 +
8685 22964 +proc sort data=&outnode;
8686 22965 +     by _NAME_;
8687 22966 +proc sort data=_tmp_GCLUSFREQ;
8688 22967 +     by _NAME_;
8689 22968 +run;
8690 22969 +
8691 22970 +data  &outnode;
8692 22971 +     merge &outnode _tmp_GCLUSFREQ; by _NAME_;
8693 22972 +     if COUNT=. then COUNT=1;
8694 22973 +run;
8695 22974 +
8696 22975 +proc datasets lib = work nolist;
8697 22976 +     delete _tmp_outrsquare _tmp_GCLUSFREQ;
8698 22977 +run;
8699 22978 +
8700 22979 +quit;
8701 22980 +%Mend MakeGClusterConstData;
8702 22981 +
8703 22982 +
```

```
8704 22983 +%macro CreateGClusterScoreCode(indata=,  globalclusi
     d=, fileref=);
8705 22984 +
8706 22985 +    data _gscoretmpds;
8707 22986 +          set &indata;
8708 22987 +          if (_TYPE_ in ('SCORE' 'MEAN' 'STD') and _
     NCL_ = 1 ) or (_TYPE_ in ('MEAN' 'STD'));
8709 22988 +          if _TYPE_ = 'MEAN' then _NAME_='MEAN';
8710 22989 +          if _TYPE_ = 'STD' then _NAME_='STD';
8711 22990 +          if _TYPE_ = 'SCORE' then _NAME_ = "GC"||st
     rip(&globalclusid);
8712 22991 +          DROP _TYPE_ _NCL_;
8713 22992 +     run;
8714 22993 +
8715 22994 +    /*  %let gscorefile =  %bquote(&EM_NODEDIR)&EM_D
     SEP.gclusterscore.sas;
8716 22995 +        GCluster Component &globalclusid ------ */
8717 22996 +
8718 22997 +    filename _file_  "&fileref";
8719 22998 +
8720 22999 +    data _null_;
8721 23000 +        %if &globalclusid eq 1 %then %do;
8722 23001 +          FILE _file_;
8723 23002 +        %end;
8724 23003 +        %else %do;
8725 23004 +          FILE _file_ MOD;
8726 23005 +        %end;
8727 23006 +
8728 23007 +        %let dsid = %sysfunc(open(work._gscoretmpds)
     );
8729 23008 +        %let nvar = %sysfunc(attrn(&dsid, NVAR));
8730 23009 +        %let vn_name =%sysfunc(varnum(&dsid, _NAME_)
     );
8731 23010 +
8732 23011 +        %let k = 1;
8733 23012 +        %do %while(^%sysfunc(fetch(&dsid)));
```

```
8734  23013 +                %let _name = %sysfunc(getvarc(&dsid,
       &vn_name));
8735  23014 +                %if &k > 2 %then %do;
8736  23015 +                put "&_name = 0 ; ";
8737  23016 +                %end;
8738  23017 +                %let k = %eval(&k+1);
8739  23018 +        %end;
8740  23019 +
8741  23020 +        %let rc = %sysfunc(rewind(&dsid));
8742  23021 +        %do i= 2 %to &nvar;
8743  23022 +            %let _varname =  %sysfunc(varname(&dsid,
       &i));
8744  23023 +            %do %while(^%sysfunc(fetch(&dsid)));
8745  23024 +                %let _name = %sysfunc(getvarc(&dsid,
       &vn_name));
8746  23025 +                %if &_name = MEAN %then
8747  23026 +                %let _mean = %sysfunc(getvarn(&dsid,
       &i));
8748  23027 +                %else %if &_name = STD %then
8749  23028 +                %let _std = %sysfunc(getvarn(&dsid,
      &i));
8750  23029 +                %else %do;
8751  23030 +                      %let coeff =  %sysfunc(getvarn
      (&dsid, &i));
8752  23031 +                      %let abscoeff = %sysfunc(abs(&
      coeff));
8753  23032 +                          %if &abscoeff >  0 %then %
      do;
8754  23033 +                       put "&_name = &_name+&coeff *
       (&_varname - &_mean)/&_std;";
8755  23034 +                           %end;
8756  23035 +                %end;
8757  23036 +             %end;
8758  23037 +             %let rc = %sysfunc(rewind(&dsid));
8759  23038 +
8760  23039 +         %end;
```

```
8761  23040 +
8762  23041 +        %let dsid= %sysfunc(close(&dsid));
8763  23042 +       run;
8764  23043 +
8765  23044 +
8766  23045 +       proc datasets lib=work nolist;
8767  23046 +            delete _gscoretmpds;
8768  23047 +       run;
8769  23048 +      quit;
8770  23049 +
8771  23050 +%mend CreateGClusterScoreCode;
8772  23051 +
8773  23052 +
8774  23053 +%macro MakeGClusterCorrelation(Indata=, ngcluster=,
      gscorecode=, outrsquare=);
8775  23054 +
8776  23055 +   %EM_REGISTER(KEY=GSCORE, TYPE=DATA);
8777  23056 +   %EM_GETNAME(KEY=GSCORE, TYPE=DATA);
8778  23057 +   %EM_REGISTER(KEY=GSCORESTAT, TYPE=DATA);
8779  23058 +   %EM_GETNAME(KEY=GSCORESTAT, TYPE=DATA);
8780  23059 +   %EM_REGISTER(KEY=GSCORETREE, TYPE=DATA);
8781  23060 +   %EM_GETNAME(KEY=GSCORETREE, TYPE=DATA);
8782  23061 +   %EM_REGISTER(KEY=GSCORECORR, TYPE=DATA);
8783  23062 +   %EM_GETNAME(KEY=GSCORECORR, TYPE=DATA);
8784  23063 +   %EM_REGISTER(KEY=GSCORECORRPLOT, TYPE=DATA);
8785  23064 +   %EM_GETNAME(KEY=GSCORECORRPLOT, TYPE=DATA);
8786  23065 +   %EM_REGISTER(KEY=GCLUSLINK, TYPE=DATA);
8787  23066 +   %EM_GETNAME(KEY=GCLUSLINK, TYPE=DATA);
8788  23067 +   %EM_REGISTER(KEY=GCLUSNODE, TYPE=DATA);
8789  23068 +   %EM_GETNAME(KEY=GCLUSNODE, TYPE=DATA);
8790  23069 +
8791  23070 +   filename gsfile "&gscorecode";
8792  23071 +
8793  23072 +   data &EM_USER_GSCORE;
8794  23073 +              set &indata;
8795  23074 +              %include  gsfile;
```

```
8796 23075 +        keep
8797 23076 +        %do i=1 %to &ngcluster;
8798 23077 +         %let gcvarname = GC&i;
8799 23078 +         &gcvarname
8800 23079 +        %end;
8801 23080 +     ;
8802 23081 +   run;
8803 23082 +
8804 23083 +   proc varclus data=&EM_USER_GSCORE outstat=&EM_USE
     R_GSCORESTAT outtree=&EM_USER_GSCORETREE
8805 23084 +    %if %upcase(&EM_PROPERTY_CLUSCOMP) eq CENTROID %
     then %do; centroid %end;
8806 23085 +    %if %upcase(&EM_PROPERTY_CLUSSOURCE) eq COV %the
     n %do; cov %end;
8807 23086 +    %if %upcase(&EM_PROPERTY_CLUSHIERACHY) eq Y %the
     n %do; hi %end;
8808 23087 +    noprint  ;
8809 23088 +     var
8810 23089 +       %do i=1 %to &ngcluster;
8811 23090 +         %let gcvarname = GC&i;
8812 23091 +         &gcvarname
8813 23092 +        %end;
8814 23093 +   ;
8815 23094 +   run;
8816 23095 +
8817 23096 +
8818 23097 +   %MakeVarClusCorrData(statds=&EM_USER_GSCORESTAT,
     corrds=&EM_USER_GSCORECORR, corrplotds=&EM_USER_GSCORECORRP
     LOT );
8819 23098 +   data &EM_USER_GSCORECORRPLOT ;
8820 23099 +        set &EM_USER_GSCORECORRPLOT;
8821 23100 +        rename _X_ = X;
8822 23101 +        rename _Y_ = Y;
8823 23102 +        label _X_ = "%sysfunc(sasmsg(sashelp.dmine,
     rpt_varclus_label_gcluster, noquote))" ;
8824 23103 +        label _Y_ = "%sysfunc(sasmsg(sashelp.dmine,
```

```
     rpt_varclus_label_gcluster, noquote))" ;
8825 23104 +   run;
8826 23105 +
8827 23106 +   %MakeGClusterConstData(indata=&EM_USER_GSCORETREE
     , inoutrsq=&outrsquare, outnode=&EM_USER_GCLUSNODE, outlink
     =&EM_USER_GCLUSLINK);
8828 23107 +
8829 23108 +   data &EM_USER_GSCORETREE;
8830 23109 +        length _NAME_ $32;
8831 23110 +        length _LABEL_ $100;
8832 23111 +        set &EM_USER_GSCORETREE(DROP=_LABEL_);
8833 23112 +        if upcase(substr(strip(_NAME_),1, 2))='GC' t
     hen do;
8834 23113 +         _LABEL_ = "%sysfunc(sasmsg(sashelp.dmine, r
     pt_varclus_label_gcluster, noquote)):"||_NAME_;
8835 23114 +        end;else do;
8836 23115 +         _LABEL_ = _NAME_;
8837 23116 +        end;
8838 23117 +       label _LABEL_ = "%sysfunc(sasmsg(sashelp.dmin
     e, rpt_varclus_label_variable, noquote))";
8839 23118 +
8840 23119 +   run;
8841 23120 +
8842 23121 +   quit;
8843 23122 +
8844 23123 +%mend MakeGClusterCorrelation;
8845 23124 +
8846 23125 +
8847 NOTE: %INCLUDE (level 1) ending.
8848 NOTE: Fileref TEMP has been deassigned.
8849 NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMEXPL.V
     ARIABLECLUSTERING_REPORT.SOURCE.
8850 23126 +%macro report;
8851 23127 +    %EM_GETNAME(key=VARCLUSMETA, type=DATA) ;
8852 23128 +
8853 23129 +    data _null_;
```

```
8854  23130 +         set &EM_USER_VARCLUSMETA;
8855  23131 +         if _N_=1 then
8856  23132 +         call symput('_trainnum', TrainNum);
8857  23133 +    run;
8858  23134 +   %if (&_trainnum = 1 ) or %upcase(&EM_PROPERTY_TWO
      STAGECLUS) = NO %then %do;
8859  23135 +           filename temp catalog 'sashelp.emexpl.var
      iableclustering_report1.source';
8860  23136 +           %include temp;
8861  23137 +           filename temp;
8862  23138 +           %report1;
8863  23139 +       %end;
8864  23140 +   %if (&_trainnum = 2 ) or %upcase(&EM_PROPERTY_TWO
      STAGECLUS) = YES %then %do;
8865  23141 +           filename temp catalog 'sashelp.emexpl.var
      iableclustering_report2.source';
8866  23142 +           %include temp;
8867  23143 +           filename temp;
8868  23144 +           %report2;
8869  23145 +    %end;
8870  23146 +
8871  23147 +   /* End of all runs, change the flag of newtrain t
      o N */
8872  23148 +    %EM_GETNAME(KEY=VARCLUSMETA, TYPE=DATA);
8873  23149 +    data &EM_USER_VARCLUSMETA;
8874  23150 +         set &EM_USER_VARCLUSMETA;
8875  23151 +         NewTrain = "N";
8876  23152 +    run;
8877  23153 +    quit;
8878  23154 +%mend report;
8879  23155 +
8880  23156 +
8881  NOTE: %INCLUDE (level 1) ending.
8882  NOTE: Fileref TEMP has been deassigned.
8883
8884  NOTE: Numeric values have been converted to character value
```

```
     s at the places given by: (Line):(Column).
8885       84545:166
8886 NOTE: There were 1 observations read from the data set EMWS
     4.VARCLUS2_VARCLUSMETA.
8887 NOTE: DATA statement used (Total process time):
8888       real time           0.00 seconds
8889       user cpu time       0.00 seconds
8890       system cpu time     0.00 seconds
8891       memory              235458.71k
8892       OS Memory           245980.00k
8893       Timestamp           07/01/2024 05:54:49 AM
8894       Step Count                        1  Switch Count  0
8895       Page Faults                       0
8896       Page Reclaims                     62
8897       Page Swaps                        0
8898       Voluntary Context Switches        8
8899       Involuntary Context Switches      0
8900       Block Input Operations            288
8901       Block Output Operations           0
8902
8903
8904 NOTE: %INCLUDE (level 1) file TEMP is file SASHELP.EMEXPL.V
     ARIABLECLUSTERING_REPORT1.SOURCE.
8905 23157 +%macro report1;
8906 23159 +   %EM_GETNAME(key=OUTRSQUARE, type=DATA) ;
8907 23160 +   %EM_GETNAME(key=OUTVARSEL, type=DATA) ;
8908 23161 +   %EM_GETNAME(key=CLUSFREQ, type=DATA) ;
8909 23163 +   data &EM_USER_OUTVARSEL;
8910 23164 +        set &EM_USER_OUTRSQUARE;
8911 23165 +        if upcase(strip(SELECTED)) = 'YES' then outp
     ut;
8912 23166 +   run;
8913 23168 +   data _tmp_outrsquare;
8914 23169 +        set &EM_USER_OUTRSQUARE;
8915 23170 +        if upcase(strip(TYPE)) = 'CLUSTERCOMP' then
     delete;
```

```
8916  23171 +   run;
8917  23173 +   proc freq data =_tmp_outrsquare noprint;
8918  23174 +        tables cluster/out=&EM_USER_CLUSFREQ sparse;
8919  23175 +   run;
8920  23177 +   proc datasets lib = work nolist;
8921  23178 +     delete _tmp_outrsquare;
8922  23179 +   run;
8923  23181 +   data &EM_USER_CLUSFREQ;
8924  23182 +       set &EM_USER_CLUSFREQ;
8925  23183 +       label
8926  23184 +           cluster = "%sysfunc(sasmsg(sashelp.dmine,
      rpt_varclus_label_cluster, NOQUOTE))"
8927  23185 +           count   = "%sysfunc(sasmsg(sashelp.dmine,
      rpt_varclus_label_freqcnt, NOQUOTE))"
8928  23186 +           percent = "%sysfunc(sasmsg(sashelp.dmine,
      rpt_varclus_label_freqpct, NOQUOTE))"
8929  23187 +       ;
8930  23188 +   run;
8931  23190 +   %EM_REPORT(key=OUTVARSEL, viewtype=DATA, block=MO
      DEL, autodisplay=Y, description=selectedvariables);
8932  23191 +   %EM_REPORT(key=CLUSFREQ, viewtype=DATA, block=MOD
      EL, autodisplay=Y, description=varfreqbycluster);
8933  23192 +   %EM_REPORT(key=OUTRSQUARE, viewtype=DATA, block=M
      ODEL, autodisplay=Y, description=varseltable);
8934  23194 +   %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);
8935  23195 +   %EM_REPORT(KEY=OUTTREE, VIEWTYPE=DENDROGRAM,  aut
      odisplay=Y, block =MODEL, Name=NAME, PARENT=PARENT, HEIGHT=
      Proportion , TipText = Variable, description=dendrogram );
8936  23196 +   %EM_REPORT(key=OUTRSQUARE, view = 1, viewtype=SCA
      TTER, x=Cluster, y=RSqRatio, autodisplay=N, group=Selected,
       block=MODEL, description=rsqplot);
8937  23197 +   %EM_REPORT(view = 1, y=RsqWithOwnClusComp);
8938  23198 +   %EM_REPORT(view=1, y=RsqWithNextClusComp);
```

```
8939 23199 +   %EM_REPORT(key=OUTCORRPLOT, viewtype=HISTOGRAM, X
     =_X_,Y=_Y_, FREQ=correlation, autodisplay=N, block=MODEL, d
     escription=varcorr);
8940 23200 +   %EM_REPORT(key=OUTCLUSCORRPLOT, viewtype=HISTOGRA
     M, X=X,Y=Y, FREQ=Correlation, autodisplay=N, block=MODEL, d
     escription=cluscorr);
8941 23201 +   %EM_REPORT(key=OUTCLUSSTRUCT, view=2, viewtype=BA
     R, block=MODEL, x=Cluster, freq=_anynumeric_ , autodisplay=
     N,  description=corrstruct) ;
8942 23203 +   /*
8943 23204 +   %EM_REPORT(KEY=OUTTREE, VIEWTYPE=DENDROGRAM,  aut
     odisplay=N, block =Dendrogram, Name=_NAME_, Parent=_PARENT_
     , Height=_VAREXP_, TipText=Variable, description=totvar);
8944 23205 +   %EM_REPORT(key=OUTCORR, viewtype=DATA, block=MODE
     L,description=corrmatrix) ;
8945 23206 +   %EM_REPORT(key=OUTSTATPLOT, viewtype=DATA, block=
     MODEL, description=sumstat) ; */
8946 23209 +%mend report1;
8947 NOTE: %INCLUDE (level 1) ending.
8948 NOTE: Fileref TEMP has been deassigned.
8949
8950 NOTE: There were 89 observations read from the data set EMW
     S4.VARCLUS2_OUTRSQUARE.
8951 NOTE: The data set EMWS4.VARCLUS2_OUTVARSEL has 28 observat
     ions and 9 variables.
8952 NOTE: DATA statement used (Total process time):
8953       real time           0.00 seconds
8954       user cpu time       0.01 seconds
8955       system cpu time     0.00 seconds
8956       memory              235458.71k
8957       OS Memory           245980.00k
8958       Timestamp           07/01/2024 05:54:49 AM
8959       Step Count                        1  Switch Count  0
8960       Page Faults                       0
8961       Page Reclaims                     127
8962       Page Swaps                        0
```

```
8963        Voluntary Context Switches        20
8964        Involuntary Context Switches      0
8965        Block Input Operations            0
8966        Block Output Operations           264
8967
8968
8969
8970  NOTE: There were 89 observations read from the data set EMW
      S4.VARCLUS2_OUTRSQUARE.
8971  NOTE: The data set WORK._TMP_OUTRSQUARE has 61 observations
       and 9 variables.
8972  NOTE: DATA statement used (Total process time):
8973        real time           0.00 seconds
8974        user cpu time       0.00 seconds
8975        system cpu time     0.00 seconds
8976        memory              235458.71k
8977        OS Memory           245980.00k
8978        Timestamp           07/01/2024 05:54:49 AM
8979        Step Count                        1  Switch Count  0
8980        Page Faults                       0
8981        Page Reclaims                     128
8982        Page Swaps                        0
8983        Voluntary Context Switches        3
8984        Involuntary Context Switches      0
8985        Block Input Operations            0
8986        Block Output Operations           264
8987
8988
8989
8990  NOTE: There were 61 observations read from the data set WOR
      K._TMP_OUTRSQUARE.
8991  NOTE: The data set EMWS4.VARCLUS2_CLUSFREQ has 28 observati
      ons and 3 variables.
8992  NOTE: PROCEDURE FREQ used (Total process time):
8993        real time           0.00 seconds
8994        user cpu time       0.00 seconds
```

```
8995        system cpu time     0.00 seconds
8996        memory              235458.71k
8997        OS Memory           245980.00k
8998        Timestamp           07/01/2024 05:54:49 AM
8999        Step Count                        1  Switch Count  0
9000        Page Faults                       0
9001        Page Reclaims                     198
9002        Page Swaps                        0
9003        Voluntary Context Switches        15
9004        Involuntary Context Switches      0
9005        Block Input Operations            0
9006        Block Output Operations           528
9007
9008
9009
9010  NOTE: Deleting WORK._TMP_OUTRSQUARE (memtype=DATA).
9011
9012  NOTE: PROCEDURE DATASETS used (Total process time):
9013        real time           0.00 seconds
9014        user cpu time       0.00 seconds
9015        system cpu time     0.00 seconds
9016        memory              235458.71k
9017        OS Memory           245980.00k
9018        Timestamp           07/01/2024 05:54:49 AM
9019        Step Count                        1  Switch Count  0
9020        Page Faults                       0
9021        Page Reclaims                     50
9022        Page Swaps                        0
9023        Voluntary Context Switches        0
9024        Involuntary Context Switches      0
9025        Block Input Operations            0
9026        Block Output Operations           8
9027
9028
9029
9030  NOTE: There were 28 observations read from the data set EMW
```

```
      S4.VARCLUS2_CLUSFREQ.
9031  NOTE: The data set EMWS4.VARCLUS2_CLUSFREQ has 28 observati
      ons and 3 variables.
9032  NOTE: DATA statement used (Total process time):
9033        real time           0.01 seconds
9034        user cpu time       0.00 seconds
9035        system cpu time     0.00 seconds
9036        memory              235458.71k
9037        OS Memory           245980.00k
9038        Timestamp           07/01/2024 05:54:49 AM
9039        Step Count                        1  Switch Count  0
9040        Page Faults                       0
9041        Page Reclaims                     470
9042        Page Swaps                        0
9043        Voluntary Context Switches        38
9044        Involuntary Context Switches      0
9045        Block Input Operations            288
9046        Block Output Operations           264
9047
9048
9049
9050  NOTE: The data set WORK.EM_USER_REPORT has 132 observations
       and 4 variables.
9051  NOTE: DATA statement used (Total process time):
9052        real time           0.02 seconds
9053        user cpu time       0.03 seconds
9054        system cpu time     0.00 seconds
9055        memory              235458.71k
9056        OS Memory           245980.00k
9057        Timestamp           07/01/2024 05:54:49 AM
9058        Step Count                        1  Switch Count  0
9059        Page Faults                       0
9060        Page Reclaims                     173
9061        Page Swaps                        0
9062        Voluntary Context Switches        0
9063        Involuntary Context Switches      0
```

```
9064        Block Input Operations            0
9065        Block Output Operations           264
9066
9067
9068
9069  NOTE: There were 132 observations read from the data set WO
      RK.EM_USER_REPORT.
9070  NOTE: The data set WORK.EM_USER_REPORT has 264 observations
       and 4 variables.
9071  NOTE: DATA statement used (Total process time):
9072        real time           0.02 seconds
9073        user cpu time       0.03 seconds
9074        system cpu time     0.00 seconds
9075        memory              235458.71k
9076        OS Memory           245980.00k
9077        Timestamp           07/01/2024 05:54:49 AM
9078        Step Count                        1  Switch Count  0
9079        Page Faults                       0
9080        Page Reclaims                     198
9081        Page Swaps                        0
9082        Voluntary Context Switches        0
9083        Involuntary Context Switches      0
9084        Block Input Operations            0
9085        Block Output Operations           272
9086
9087
9088
9089  NOTE: There were 264 observations read from the data set WO
      RK.EM_USER_REPORT.
9090  NOTE: The data set WORK.EM_USER_REPORT has 396 observations
       and 4 variables.
9091  NOTE: DATA statement used (Total process time):
9092        real time           0.02 seconds
9093        user cpu time       0.03 seconds
9094        system cpu time     0.00 seconds
9095        memory              235458.71k
```

```
9096        OS Memory           245980.00k
9097        Timestamp           07/01/2024 05:54:49 AM
9098        Step Count                      1  Switch Count  0
9099        Page Faults                     0
9100        Page Reclaims                   196
9101        Page Swaps                      0
9102        Voluntary Context Switches      0
9103        Involuntary Context Switches    0
9104        Block Input Operations          0
9105        Block Output Operations         520
9106
9107
9108
9109 NOTE: There were 396 observations read from the data set WO
     RK.EM_USER_REPORT.
9110 NOTE: The data set WORK.EM_USER_REPORT has 528 observations
      and 4 variables.
9111 NOTE: DATA statement used (Total process time):
9112        real time           0.02 seconds
9113        user cpu time       0.03 seconds
9114        system cpu time     0.00 seconds
9115        memory              235458.71k
9116        OS Memory           245980.00k
9117        Timestamp           07/01/2024 05:54:49 AM
9118        Step Count                      1  Switch Count  0
9119        Page Faults                     0
9120        Page Reclaims                   196
9121        Page Swaps                      0
9122        Voluntary Context Switches      0
9123        Involuntary Context Switches    0
9124        Block Input Operations          0
9125        Block Output Operations         520
9126
9127
9128
9129 NOTE: There were 528 observations read from the data set WO
```

```
      RK.EM_USER_REPORT.
9130  NOTE: The data set WORK.EM_USER_REPORT has 660 observations
       and 4 variables.
9131  NOTE: DATA statement used (Total process time):
9132        real time           0.02 seconds
9133        user cpu time       0.03 seconds
9134        system cpu time     0.00 seconds
9135        memory              235458.71k
9136        OS Memory           245980.00k
9137        Timestamp           07/01/2024 05:54:49 AM
9138        Step Count                        1  Switch Count  0
9139        Page Faults                       0
9140        Page Reclaims                     196
9141        Page Swaps                        0
9142        Voluntary Context Switches        0
9143        Involuntary Context Switches      0
9144        Block Input Operations            0
9145        Block Output Operations           776
9146
9147
9148
9149  NOTE: There were 660 observations read from the data set WO
      RK.EM_USER_REPORT.
9150  NOTE: The data set WORK.EM_USER_REPORT has 793 observations
       and 4 variables.
9151  NOTE: DATA statement used (Total process time):
9152        real time           0.02 seconds
9153        user cpu time       0.03 seconds
9154        system cpu time     0.00 seconds
9155        memory              235458.71k
9156        OS Memory           245980.00k
9157        Timestamp           07/01/2024 05:54:49 AM
9158        Step Count                        1  Switch Count  0
9159        Page Faults                       0
9160        Page Reclaims                     228
9161        Page Swaps                        0
```

```
9162        Voluntary Context Switches        0
9163        Involuntary Context Switches      0
9164        Block Input Operations            0
9165        Block Output Operations           776
9166
9167
9168
9169  NOTE: There were 793 observations read from the data set WO
      RK.EM_USER_REPORT.
9170  NOTE: The data set WORK.EM_USER_REPORT has 926 observations
       and 4 variables.
9171  NOTE: DATA statement used (Total process time):
9172        real time           0.02 seconds
9173        user cpu time       0.02 seconds
9174        system cpu time     0.00 seconds
9175        memory              235458.71k
9176        OS Memory           245980.00k
9177        Timestamp           07/01/2024 05:54:49 AM
9178        Step Count                        1  Switch Count  0
9179        Page Faults                       0
9180        Page Reclaims                     228
9181        Page Swaps                        0
9182        Voluntary Context Switches        0
9183        Involuntary Context Switches      0
9184        Block Input Operations            0
9185        Block Output Operations           1032
9186
9187
9188
9189  NOTE: There were 926 observations read from the data set WO
      RK.EM_USER_REPORT.
9190  NOTE: The data set WORK.EM_USER_REPORT has 1059 observation
      s and 4 variables.
9191  NOTE: DATA statement used (Total process time):
9192        real time           0.04 seconds
9193        user cpu time       0.04 seconds
```

```
9194        system cpu time     0.00 seconds
9195        memory              235458.71k
9196        OS Memory           245980.00k
9197        Timestamp           07/01/2024 05:54:49 AM
9198        Step Count                        1  Switch Count  0
9199        Page Faults                       0
9200        Page Reclaims                     260
9201        Page Swaps                        0
9202        Voluntary Context Switches        0
9203        Involuntary Context Switches      0
9204        Block Input Operations            0
9205        Block Output Operations           1032
9206
9207
9208
9209 NOTE: There were 1059 observations read from the data set W
     ORK.EM_USER_REPORT.
9210 NOTE: The data set WORK.EM_USER_REPORT has 1192 observation
     s and 4 variables.
9211 NOTE: DATA statement used (Total process time):
9212        real time           0.02 seconds
9213        user cpu time       0.03 seconds
9214        system cpu time     0.00 seconds
9215        memory              235458.71k
9216        OS Memory           245980.00k
9217        Timestamp           07/01/2024 05:54:49 AM
9218        Step Count                        1  Switch Count  0
9219        Page Faults                       0
9220        Page Reclaims                     260
9221        Page Swaps                        0
9222        Voluntary Context Switches        0
9223        Involuntary Context Switches      0
9224        Block Input Operations            0
9225        Block Output Operations           1288
9226
9227
```

```
9228
9229 NOTE: There were 1192 observations read from the data set W
     ORK.EM_USER_REPORT.
9230 NOTE: The data set WORK.EM_USER_REPORT has 1325 observation
     s and 4 variables.
9231 NOTE: DATA statement used (Total process time):
9232       real time           0.02 seconds
9233       user cpu time       0.02 seconds
9234       system cpu time     0.00 seconds
9235       memory              235458.71k
9236       OS Memory           245980.00k
9237       Timestamp           07/01/2024 05:54:49 AM
9238       Step Count                        1  Switch Count  0
9239       Page Faults                       0
9240       Page Reclaims                     292
9241       Page Swaps                        0
9242       Voluntary Context Switches        0
9243       Involuntary Context Switches      0
9244       Block Input Operations            0
9245       Block Output Operations           1288
9246
9247
9248
9249 NOTE: There were 1325 observations read from the data set W
     ORK.EM_USER_REPORT.
9250 NOTE: The data set WORK.EM_USER_REPORT has 1457 observation
     s and 4 variables.
9251 NOTE: DATA statement used (Total process time):
9252       real time           0.02 seconds
9253       user cpu time       0.03 seconds
9254       system cpu time     0.01 seconds
9255       memory              235458.71k
9256       OS Memory           245980.00k
9257       Timestamp           07/01/2024 05:54:49 AM
9258       Step Count                        1  Switch Count  0
9259       Page Faults                       0
```

```
9260        Page Reclaims                       292
9261        Page Swaps                          0
9262        Voluntary Context Switches          0
9263        Involuntary Context Switches        0
9264        Block Input Operations              0
9265        Block Output Operations             1544
9266
9267
9268
9269  NOTE: There were 1 observations read from the data set EMWS
      4.VARCLUS2_VARCLUSMETA.
9270  NOTE: The data set EMWS4.VARCLUS2_VARCLUSMETA has 1 observa
      tions and 5 variables.
9271  NOTE: DATA statement used (Total process time):
9272        real time           0.01 seconds
9273        user cpu time       0.01 seconds
9274        system cpu time     0.00 seconds
9275        memory              235458.71k
9276        OS Memory           245980.00k
9277        Timestamp           07/01/2024 05:54:49 AM
9278        Step Count                       1  Switch Count  0
9279        Page Faults                      0
9280        Page Reclaims                    127
9281        Page Swaps                       0
9282        Voluntary Context Switches       32
9283        Involuntary Context Switches     2
9284        Block Input Operations           0
9285        Block Output Operations          264
9286
9287
9288  23212  *-------------------------------------------------
      ---------*;
9289  23213  * End REPORT: VarClus2;
9290  23214  *-------------------------------------------------
      ---------*;
9291  23215
```

```
9292
9293  23216  /* Reset EM Options */
9294  23217  options formchar="|----|+|---+=|-/\<>*";
9295  23218  options nocenter ls=256 ps=10000;
9296  23219  goptions reset=all device=GIF NODISPLAY;
9297
9298  23220  proc sort data=WORK.EM_USER_REPORT;
9299  23221  by ID VIEW;
9300  23222  run;
9301
9302  NOTE: There were 1457 observations read from the data set W
      ORK.EM_USER_REPORT.
9303  NOTE: The data set WORK.EM_USER_REPORT has 1457 observation
      s and 4 variables.
9304  NOTE: PROCEDURE SORT used (Total process time):
9305        real time           0.00 seconds
9306        user cpu time       0.00 seconds
9307        system cpu time     0.00 seconds
9308        memory              235458.71k
9309        OS Memory           245980.00k
9310        Timestamp           07/01/2024 05:54:49 AM
9311        Step Count                        1  Switch Count  0
9312        Page Faults                       0
9313        Page Reclaims                     359
9314        Page Swaps                        0
9315        Voluntary Context Switches        0
9316        Involuntary Context Switches      0
9317        Block Input Operations            0
9318        Block Output Operations           1544
9319
9320
```