Seven (plus or minus two) Clusters, A Monte Carlo Study

Larry Hoyle, Policy Research Institute, The University of Kansas

Abstract

This paper employs k-means clustering using PROC FASTCLUS to explore using from 2 to 12 clusters to represent univariate data in samples from five different distributions. Implications for shaded maps and an alternative to traditional histograms are also discussed.

The Monte Carlo Study

Given a maximum number of clusters desired, PROC FASTCLUS computes optimum clusters of observations. The default parameters produce clusters with the minimum possible within cluster variance. When clustering univariate data a small number of clusters can explain a large proportion of the target variable's variance.

The SAS® program listed in Appendix 1 generated 500 samples of 300 and of 1000 cases from each of the uniform, normal, and exponential distributions, as well as from bimodal and trimodal distributions.

The bimodal distribution was drawn as:

50% Normal(mean=0, variance=1)
50% Normal(mean=4, variance =1)

The bimodal distribution was drawn as:

33.33% Normal(mean=0, variance=1)
33.34% Normal(mean=5, variance =1)
33.33% Normal(mean=9, variance =1)

The program ran PROC FASTCLUS 11 times on each sample, specifying from 2 to 12 clusters. It then recorded R^2, the proportion of variance accounted for, for each clustering.

Figures 1 through 10 show scatter plots of the R^2 values by the number of clusters requested for different combinations of sample size and distribution. Each small dot represents the clustering on one sample. Each plot has 5500 dots, representing 5500 cluster analyses.

The variable on the x-axis, number of clusters, has a small random number added so as to make more points visible. In most of the plots, most of the individual points are so close together that they cannot be distinguished. In figure 5 you can see several individual points.

For the uniform distribution, the value of R^2 falls into tight ranges for each selection of the number of clusters. Even analyses with two clusters, always explained over 70% of the variance, and those with 5 clusters always explained about 95% of the variance or more.

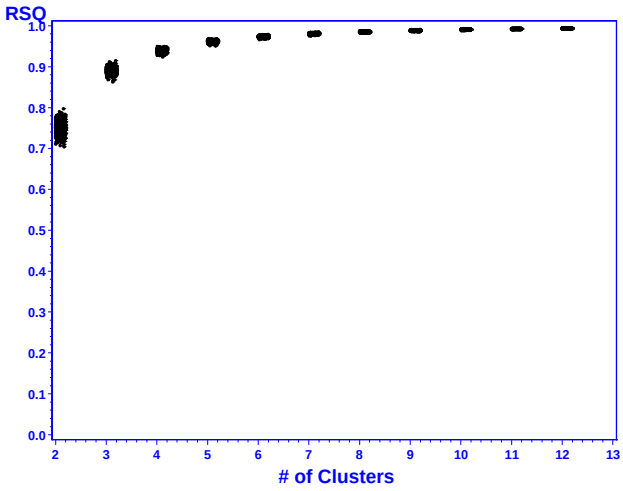


Figure 1 Uniform, samples of 300

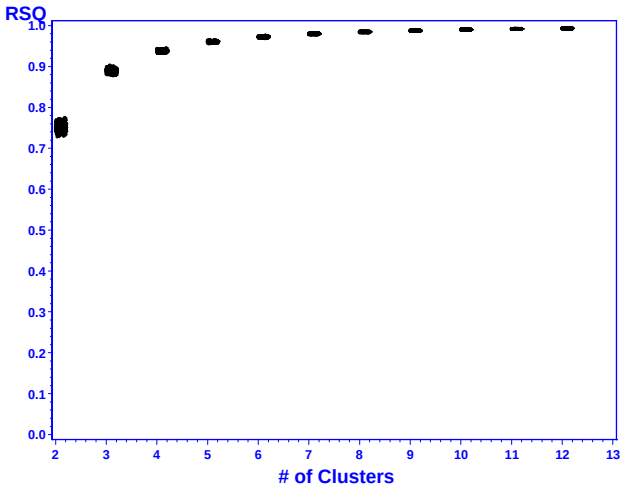


Figure 2 Uniform, samples of 1000

Ranges for the normal distribution are not so tight. It takes an extra cluster to explain more than 70% and it takes eight clusters to get into the 95% or more range.

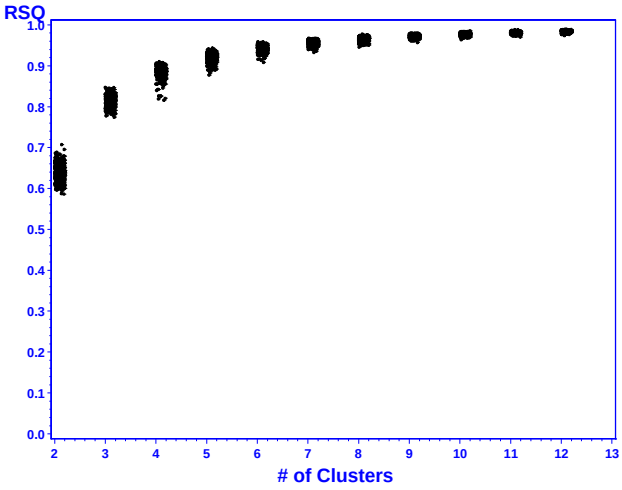


Figure 3 Normal samples of 300

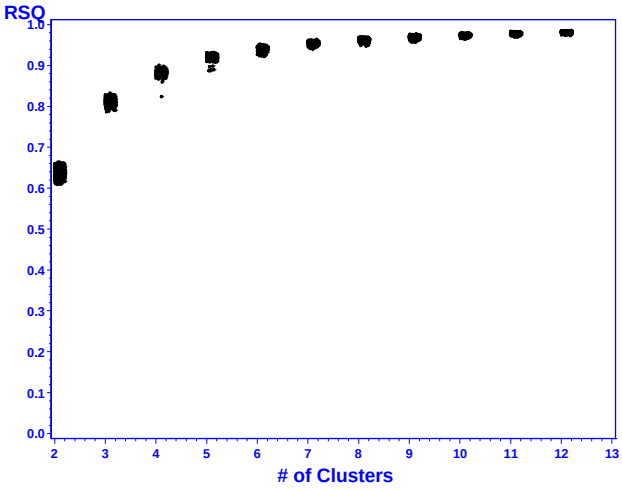


Figure 4 Normal samples of 1000

Ranges for the Exponential distribution are much wider, especially for small numbers of clusters. Like the normal distribution, it takes eight clusters to get into the 95% of variance explained range.
One curious feature of the plot of R^2 values for the exponential samples of size 300 is that there are a few samples with really low values of R^2. Figure 5a is a box plot showing the distribution of the sample with the lowest value, which has an R^2 of 0.2126 for 2 clusters.

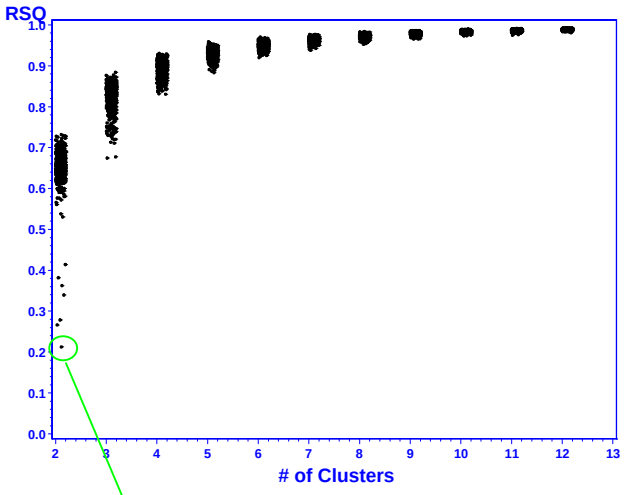


Figure 5 Exponential samples of 300

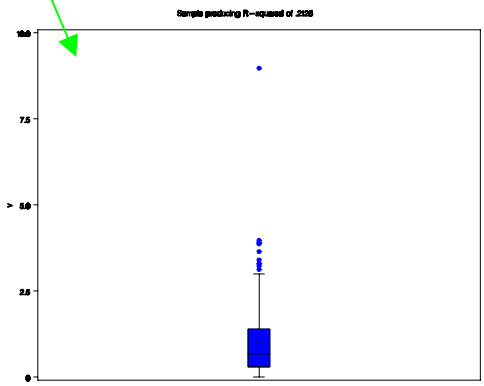


Figure 5a Distribution of the worst clustered sample

Note that the sample has an outlier at 8.97 and that all other observations are below 4. This outlier becomes one of the two clusters leaving the rest of the sample in one cluster with a mean of .98. Other samples without an extreme outlier can be represented by two cluster means and naturally have a higher R^2.

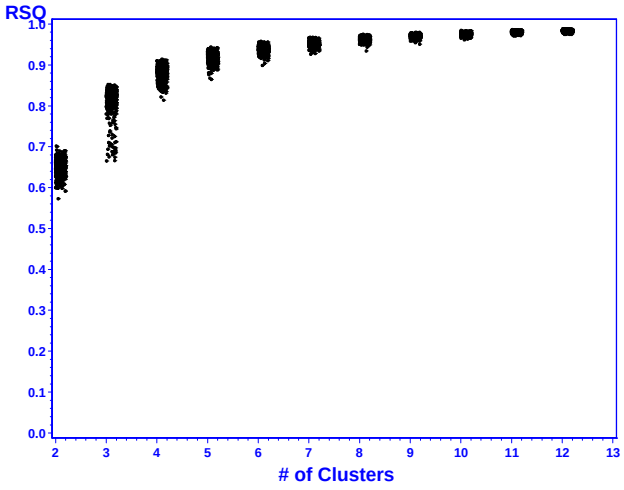


Figure 6 Exponential samples of 1000

Samples with Clusters

All of the preceding examples were with samples drawn from distributions that do not have underlying clusters. What happens if there really are clusters?

For these examples, samples were drawn from either a bimodal distribution or a trimodal distribution.

R^2 was .78 or higher with just 2 clusters for samples drawn from the bimodal distribution. The samples also formed very tight groups at each level of # of clusters.

As might be expected, R^2 takes a jump between 2 and 3 clusters when there really are three clusters in the sample. The samples also form very tight groups at 3 and above clusters.

In all four cases where there are underlying clusters, 5 clusters account for over 90% of the sample variance.

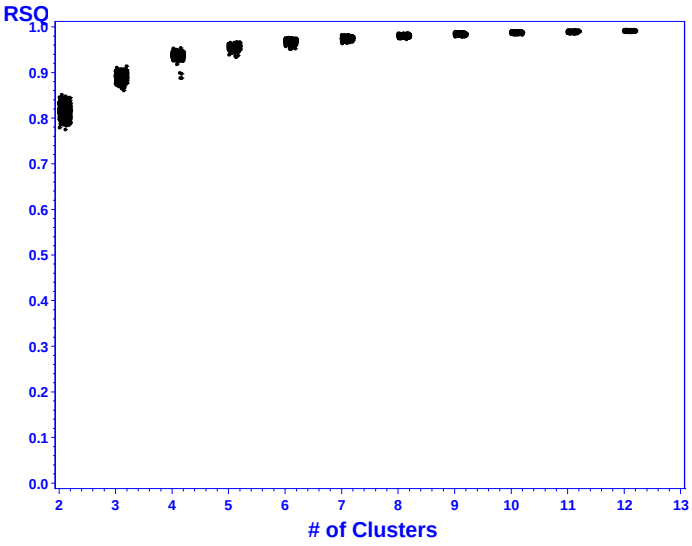


Figure 7 Bimodal samples of 300

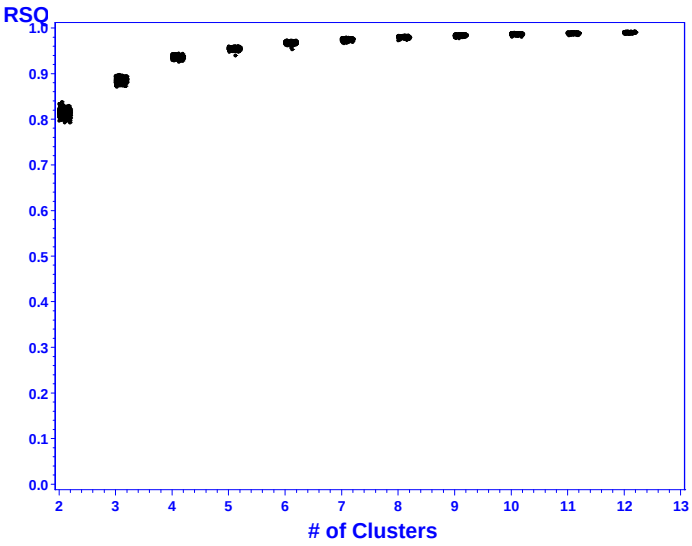


Figure 8 Bimodal samples of 1000

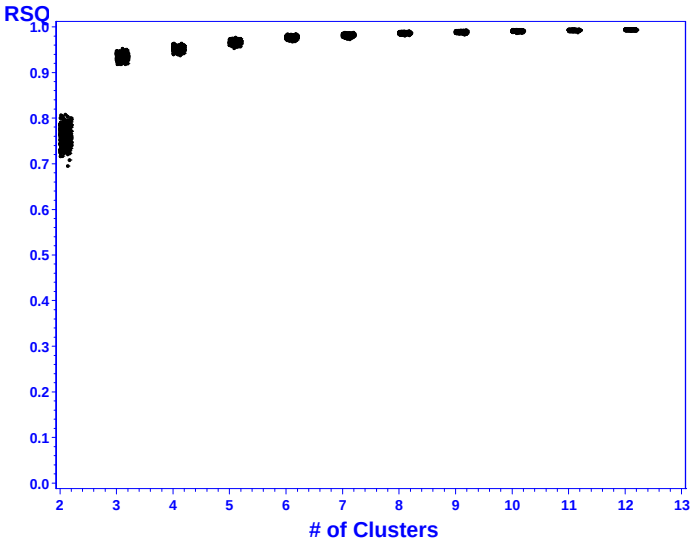


Figure 9 Trimodal samples of 300

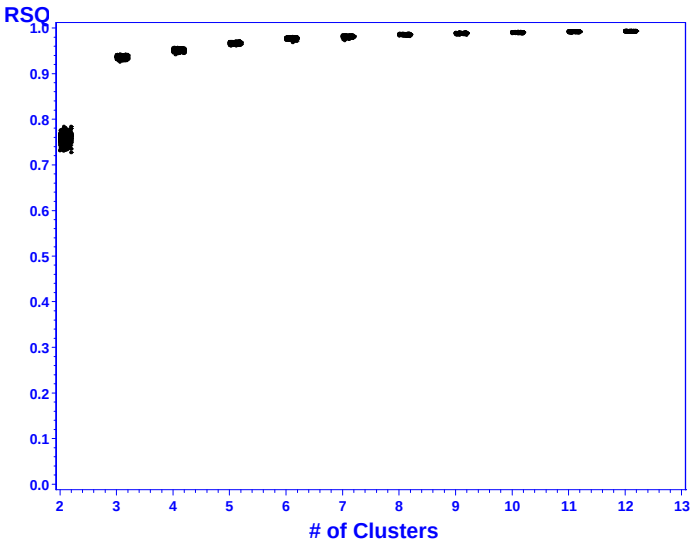


Figure 10 Trimodal samples of 1000

Table 1 shows the minimum R^2 for the three unimodal distributions by each number of clusters requested.

Table 1, Minimum R squared by sample size and distribution						
	Exponential		Normal		Uniform	
	300	1000	300	1000	300	1000
Clusters						
2	0.213	0.573	0.586	0.610	0.704	0.729
3	0.674	0.665	0.774	0.787	0.863	0.878
4	0.831	0.814	0.816	0.824	0.924	0.932
5	0.883	0.865	0.877	0.887	0.951	0.956
6	0.920	0.899	0.908	0.922	0.966	0.969
7	0.938	0.926	0.933	0.940	0.975	0.978
8	0.953	0.934	0.945	0.947	0.982	0.982
9	0.966	0.951	0.957	0.957	0.985	0.986
10	0.973	0.961	0.964	0.964	0.988	0.989
11	0.975	0.971	0.970	0.969	0.990	0.991
12	0.981	0.975	0.974	0.973	0.992	0.992

Each number in Table 1 is the minimum across the 500 trials at that combination of sample size and number of clusters selected. Note that across all distributions, at least 86.5% of the variance is accounted for by just five clusters. Nine clusters always accounted for at least 95% of the variance.
In other words, we could recode the original values in any sample with just 9 cluster means and lose very little information as measured by squared deviation from the sample mean.

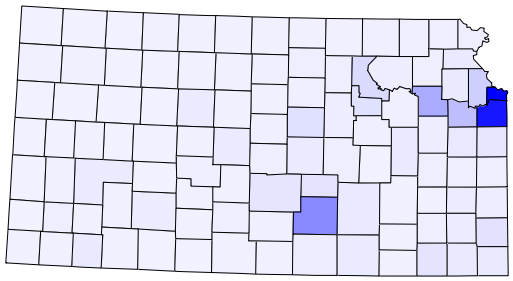
Maps
This phenomenon has allowed cartographers to use clustering to better assign shading for choroplethic maps, instead of the more commonly seen assignment by equal intervals or equal frequencies. Clustering into 7 or so groups represents the data well even when there are no underlying groups.

Consider a map of population density of Kansas counties for 2000. The distribution is highly skewed, with the two counties in the Kansas City metro area having much higher density than all the others.
Compare the following Figures:

- 11 – shading proportionally (unclassed)
- 12 – shading proportionally by seven clusters
- 13 – equal spaced shading by seven equal intervals
- 14 – equal spaced shading by septiles

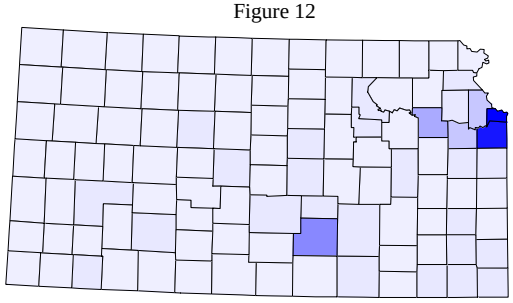
Proportional Shading

Figure 11



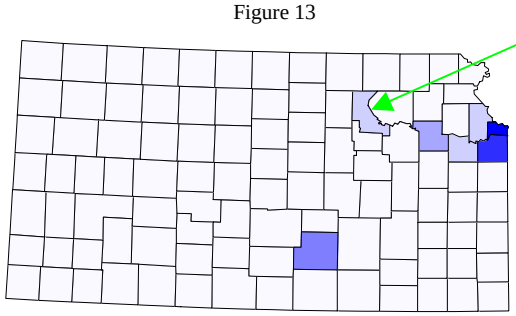
Cluster Means

Figure 12



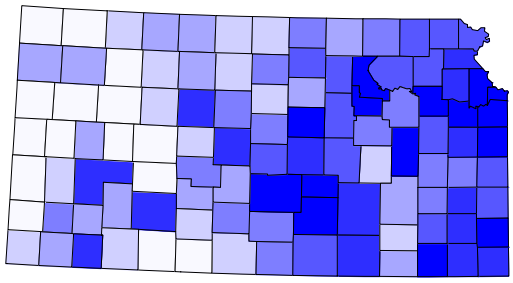
Equal Intervals

Figure 13



Equal Numbers (septiles)

Figure 14



The unclassed map and the map using the clusters appear very similar. The equal intervals map doesn't display some of the subtle differences among the western counties and overemphasizes the difference between Riley and the adjacent counties. The equal numbers map shows a very different pattern from the others.

Histograms

Assignment of shading categories to a map is analogous to dividing the x-axis into intervals to do a histogram. When we draw a bar the width of the interval and the height of the count of the number of observations in that interval, we are essentially using the midpoint of the interval to represent every value in that interval. In a typical histogram we are therefore creating equally spaced clusters. This is very unlikely to be the optimum clustering. Suppose instead that we first perform a k-means clustering and plot 9 vertical bars centered on each cluster mean so that the bars are not equally spaced. Alternatively, we could start at 2 clusters and perform clusterings with increasing numbers of clusters until we account for a predetermined proportion of variance, say 95% or 99%. We could also draw the bars in such a way as to show the number of observations composing them, like little weights on a balance beam.

Figure 15 shows a scatter plot of a sample of 200 observations drawn from a normal distribution. The vertical dimension is a random number assigned to allow better viewing of individual points. Note that for this particular sample the two lowest values are somewhat separated from the rest.

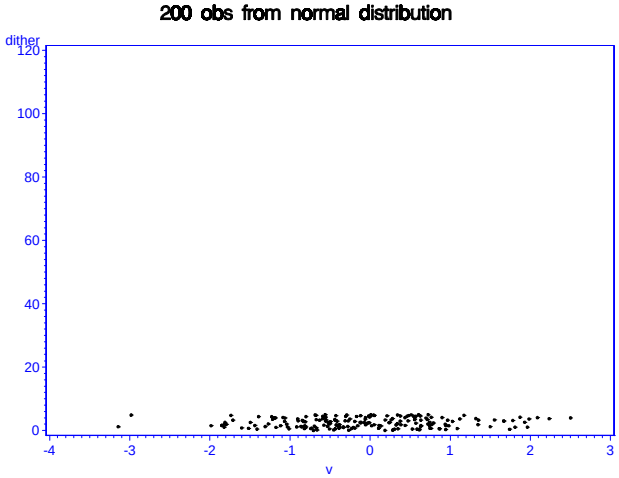


Figure 16 at the right shows histograms of the population density data using from 5 to 9 intervals in the left hand column, and shows needle plots of the frequencies of the corresponding number of cluster means in the right hand column.

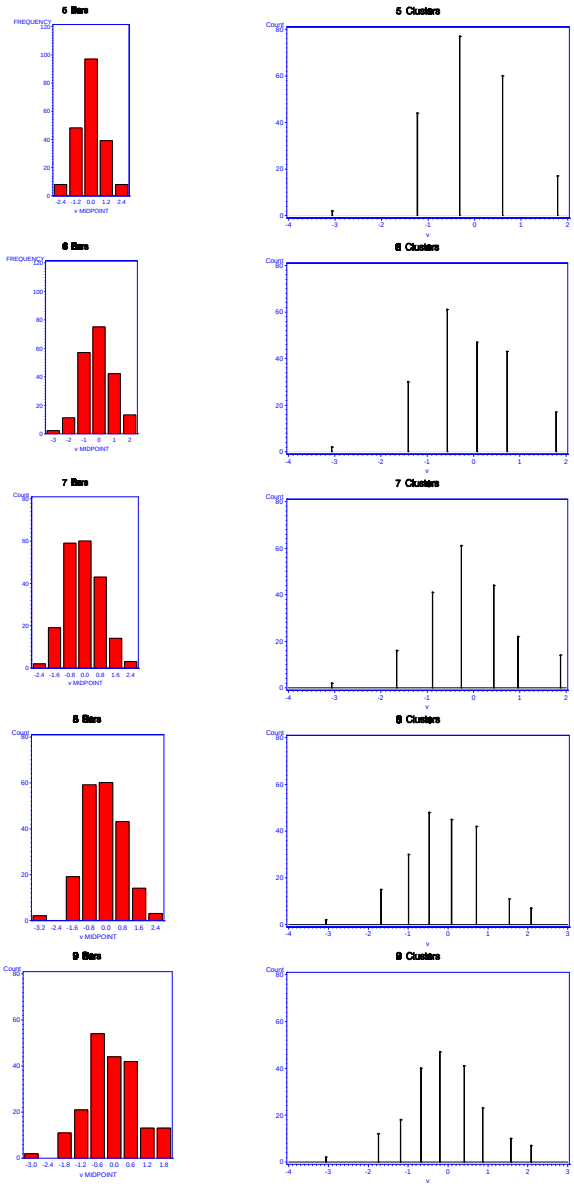


Figure 16, a comparison of bar and needle plots

Note that the bar charts do not show the separation of the two lowest values until there are 8 bars. The needle charts of the cluster means show that separation even at 5 groups.

Appendix 2 shows the code used to generate these charts.

Postscript:

In his classic 1956 paper,"The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information", George Miller discussed human ability to process unidimensional stimuli. It seems we have a capacity to distinguish at most 5 to 9 levels of a stimulus or perceptual dimension. Perhaps it is more than a fortunate coincidence that quantizing an input into the right 9 levels can do a very good job of representing the underlying continuous variable.

Appendix 1– 55000 FASTCLUS runs

```
option notes;
%global libroot;

/* %let libroot=d:\projects\sugs\mwsug2001\sasdata\final; */

%let libroot=c:\users\lhoyle\projects\sugs\mwsug2001\sasdata\final\multi;

libname projects "&libroot";

libname samples "&libroot\samples";

                    /* create the logging datasets */

%MACRO INITLOG(sampsiz=100, distrfn=uni);
proc sql;
  create table projects.&distrfn.&sampsiz.rsq
  (distrfn char(4),
   nclusts num,
   sampsiz num,
   rsq num);

   create table work.&distrfn.&sampsiz.cntr
    (distrfn char(4),
     nclusts num,
                sampsiz num,
                v num);
quit;

%mend initlog;

        /* macro to run 11 clusterings on one sample  */

%MACRO RUNCLUST(sampsiz=100, distrfn=uni, run=1, dset=foo);

            /* extract one sample from the  dataset of all samples */

data onev;
  keep v;
  start=1 + (&run-1)*&sampsiz;    /* skip run-1 sets of sampsiz observations */
  end=&run*&sampsiz;
  put start= end=;
  do ixp=start to end;
    set samples.&dset point=ixp;
    output;
  end;
  stop;
run;
sasfile onev load;       /*  load the file into memory */

     /*   run a fastclus for each of 2 to 12 clusters */
%DO IXF=2 %TO 12;
  proc fastclus data=onev
              maxclusters=&ixf
              cluster=clust&ixf
              least=2
                          /* mean=outmn&ixf */
                          /* out=out&ixf */
              outstat=stat&ixf
              noprint
              ;
       var v;
  run;
                        /*  log the r-squares   */
  proc sql;
    insert into projects.&distrfn.&sampsiz.rsq
        set nclusts=&ixf,
        sampsiz=&sampsiz,
        distrfn="&distrfn",
        rsq=(select v from stat&ixf where _type_='RSQ')
        ;

                        /* log the cluster means */
    create table t as
     select "&distrfn" as distrfn,
              &ixf as nclusts,
              &sampsiz as sampsiz,
              v
       from stat&ixf
       where _type_='CENTER'
       order by v;

insert into work.&distrfn.&sampsiz.cntr
                  (distrfn ,
                   nclusts ,
                   sampsiz ,
                   v )
                  select distrfn ,
                  nclusts ,
                   sampsiz ,
                   v
       from t;
  quit;

%END;

sasfile onev close;      /*  unload  from memory */


%MEND RUNCLUST;
                /*  spin the wheel - a lot    */

%MACRO RUNNER(dist=uni, sampsz=100, nruns=50, seed=123841) ;
%let thissamp=s&nruns&dist._&sampsz._&seed;
                 /* create a dataset with all samples in it for this set of runs */
data samples.&thissamp;
  keep s v;
  seed=&seed;
  totaln=&sampsz*&nruns;
  do ix=1 to totaln;
     s=1+floor((ix-1)/&sampsz);
     %IF "&dist" eq "bi" %THEN %DO;
      call rannor(seed,v);
                  call ranuni(seed,u);
                  if u>.5 then v=v+4;
                 %END;
     %ELSE  %IF "&dist" eq "tri" %THEN %DO;
      call rannor(seed,v);
                  call ranuni(seed,u);
                  if u>.3333 then v=v+4;
                  if u>.6667 then v=v+5;
                 %END;
     %ELSE   %DO;
      call ran&dist.(seed,v);
                  %END;
    output;
  end;
run;

%INITLOG(sampsiz=&sampsz, distrfn=&dist);

%DO ixr=1 %TO &nruns;
%RUNCLUST(sampsiz=&sampsz, distrfn=&dist, run=&ixr, dset=&thissamp);
  %put _____run  &ixr  ends;
%END;

     /* renumber the clusters lowest value to highest */
data projects.&&dist.&sampsz.cntr;
  set work.&&dist.&sampsz.cntr;
  retain nc 0;
  retain cluster;
  drop nc;
  if nc ne nclusts then do;
    cluster=0;
  end;
  cluster=cluster+1;
  nc=nclusts;
  output;
run;
%MEND RUNNER;
```

```
/* here is where we specify which samples to run */
/* capture the time to measure elapsed wall clock */
data _null_;
  start=datetime();
  cstart=put(start,datetime21.3);
  call symput("stmv",cstart);
  put cstart=;
  run;

option nonotes;
title "with sasfile";

%RUNNER(dist=bi, sampsz=300, nruns=500, seed=5398742);
%RUNNER(dist=tri, sampsz=300, nruns=500, seed=8434685);


%RUNNER(dist=bi, sampsz=1000, nruns=500, seed=7831249);
%RUNNER(dist=tri, sampsz=1000, nruns=500, seed=875961);
%RUNNER(dist=uni, sampsz=1000, nruns=500, seed=293897);
  /*   */
option notes;
                /* print out the elapsed wall clock */

 data _null_;

 start=input("&stmv",datetime23.);
 now=datetime();
 interval=now-start;
 put "this program took " interval " seconds";
run;



                  /* table all the minimum rsq */

%MACRO TABMINS(sampsiz=1000);
proc sql;
  create table e&sampsiz.min as
  select nclusts, min(rsq) as e&sampsiz.min
  from projects.exp&sampsiz.rsq
   group by nclusts;

  create table n&sampsiz.min as
  select nclusts, min(rsq) as n&sampsiz.min
  from projects.nor&sampsiz.rsq
   group by nclusts;

 create table u&sampsiz.min as
  select nclusts, min(rsq) as u&sampsiz.min
  from projects.uni&sampsiz.rsq
   group by nclusts;


  create table projects.mins&sampsiz as
  select e&sampsiz.min.nclusts,
        e&sampsiz.min.e&sampsiz.min,
        n&sampsiz.min.n&sampsiz.min,
                                u&sampsiz.min.u&sampsiz.min
  from e&sampsiz.min, n&sampsiz.min, u&sampsiz.min
  where e&sampsiz.min.nclusts=n&sampsiz.min.nclusts and
e&sampsiz.min.nclusts=u&sampsiz.min.nclusts;
  quit;

    /* export to an Excel Spreadsheet */
PROC EXPORT DATA= Projects.Mins&sampsiz
          OUTFILE= "&libroot\mins&sampsiz..xls"
          DBMS=EXCEL2000 REPLACE;
RUN;

%MEND TABMINS;

/*
%TABMINS(sampsiz=1000);
%TABMINS(sampsiz=300);

proc sql;
  create table projects.mins as
  select mins300.nclusts,
        mins300.e300min, mins1000.e1000min,
        mins300.n300min, mins1000.n1000min,
        mins300.u300min, mins1000.u1000min
  from projects.mins1000, projects.mins300
  where mins300.nclusts=mins1000.nclusts;
quit;


proc means data=PROJECTS.UNI1000RSQ     vardef=DF
    MIN MAX   MEAN   ;
  var rsq ;
  class nclusts ;
  ;
run;

proc means data=PROJECTS.UNI300RSQ     vardef=DF
    MIN MAX   MEAN   ;
  var rsq ;
  class nclusts ;
  ;
run;
*/


          /* scatter plot # clusters by RSQ */

%MACRO SPLOT(dist=exp, sampsiz=300);
          /* dither the number of clusters */
data &dist.&sampsiz.RSQ;
  set projects.&dist.&sampsiz.RSQ;
  ncfudged=nclusts+(ranuni(0)*.2);
run;

filename spfile "&libroot.\pics\&dist.&sampsiz.RSQ.cgm";

                          /* cgmmwwc cgmof97p cgmof97l */
goptions reset=(axis, legend, pattern, symbol, title, footnote) norotate
        hpos=0 vpos=0 htext=  ctext= target= gaccess=  ;

goptions targetdevice=cgmof97l device=cgmof97L  ctext=blue  ftext='HelveticaBold'
        gsfname=spfile   gsfmode=replace
        graphrc interpol=join;

symbol1 c=DEFAULT i=NONE v=DOT h=1 PCT ;

axis1   color=blue width=2.0 style=1
   label=( height=3 pct  "# of Clusters")
   ;
axis2   color=blue  width=2.0
    label=( height=3 pct  " RSQ")
    order=(0 to 1 by .1)
  ;
axis3
   color=blue
   width=2.0
   ;

proc gplot data=&dist.&sampsiz.RSQ;
   plot rsq * ncfudged /
      haxis=axis1
      vaxis=axis2
      frame   ;
run; quit;
%MEND SPLOT;
              /* one plot for each combination */

%SPLOT(dist=uni, sampsiz=300);
%SPLOT(dist=uni, sampsiz=1000);
%SPLOT(dist=exp, sampsiz=300);
%SPLOT(dist=exp, sampsiz=1000);
%SPLOT(dist=nor, sampsiz=300);
%SPLOT(dist=nor, sampsiz=1000);
%SPLOT(dist=bi, sampsiz=300);
%SPLOT(dist=bi, sampsiz=1000);
%SPLOT(dist=tri, sampsiz=300);
%SPLOT(dist=tri, sampsiz=1000);
```

Appendix 2 Generating the needle charts

```
%let sampsiz=200;

%global libroot;
%let libroot=d:\projects\sugs\mwsug2001;
options mprint;

data forneedl;
  keep v dither;
  retain  seed 912837;
  retain  seed2 0;
  do ix=1 to &sampsiz;
     call rannor(seed,v);
    dither=5*ranuni(seed2);
    output;
  end;
run;

goptions reset=(axis, legend, pattern, symbol, title, footnote) norotate
        hpos=0 vpos=0 htext= ftext= ctext= target= gaccess= gsfname= colors=
       ctitle=black;

axis1
   color=blue width=2.;
axis2
   color=blue  width=2.0
   order=(0 to 80 by 20) ;
axis3
   color=blue   width=2.0  ;

symbol1 c=DEFAULT i=NONE v=DOT h=1 PCT ;
 title "200 obs from normal distribution";

 filename needle "&libroot.\dither.cgm";
goptions targetdevice=cgmof97l device=cgmof97L;
goptions gsfname=needle gsfmode=replace;

proc gplot data=work.forneedl;
   plot dither * v /
      haxis=axis1
      vaxis=axis2
      frame;
run; quit;

symbol1 c=DEFAULT
   i=needle
   color=black
   width=8;
axis1
   color=blue   width=2.0 ;
axis2
   color=blue  width=2.0
   order=(0 to 80 by 20)
   label=("Count");
axis3
   color=blue     width=2.0 ;

%MACRO CLUSTER;
%DO IXF=2 %TO 16;
  proc fastclus data=forneedl
              maxclusters=&ixf
              cluster=clust&ixf
              least=2
              mean=ndlmn&ixf
                           /* out=out&ixf */
              outstat=ndlstat&ixf
              noprint;
        var v;
  run;

filename needle "&libroot.\needle&ixf..cgm";

goptions gsfname=needle gsfmode=replace;
title "&IXF Clusters";

proc sort data=WORK.ndlmn&ixf;
 by v;

proc gplot data=WORK.ndlmn&ixf;
   plot _FREQ_ * v /
      haxis=axis1
      vaxis=axis2
      frame ;
run; quit;

filename bar "&libroot.\bar&ixf..cgm";

goptions gsfname=bar gsfmode=replace;

title "&IXF Bars";

proc gchart data=work.forneedl;
   vbar v /
      maxis=axis1
      raxis=axis2
      frame
      type=FREQ
      levels=&IXF
     patternid=by;
run; quit;

%END;
%MEND CLUSTER;
%CLUSTER;
```

Trademarks:

References:

Jenks, George F. Generalization in Statistical Mapping. *Annals of the Association of American Geographers*, Vol 53, No. 1 March 1963 pp. 15-26

Jenks, George F. and Duane S. Knos. 1961. The Use of Shading Patterns in Graded Series. *Annals of the Association of American Geographers* 51: 316-334.

Miller, George A. The Magical Number Seven, Plus or Minus Two: Some Limits in our Capacity for Processing Information. *The Psychological Review*, 1956, vol. 63 pp. 81-97.

Slocum, Terry A.; *Thematic cartography and visualization* Upper Saddle River, N.J., Prentice Hall, 1998.

Contact Information:

Larry Hoyle
Policy Research Institute
The University of Kansas
LarryHoyle@ku.edu