

CHAPTER

24

The GREDUCE Procedure

<i>Overview</i>	895
<i>Concepts</i>	897
<i>About the Input Map Data Set</i>	897
<i>About Unmatched Area Boundaries</i>	897
<i>Procedure Syntax</i>	897
<i>PROC GREDUCE Statement</i>	898
<i>ID Statement</i>	899
<i>Using the GREDUCE Procedure</i>	900
<i>Specifying Density Levels</i>	900
<i>Subsetting a Map Data Set</i>	901
<i>Examples</i>	902
<i>Example 1: Reducing the Map of Canada</i>	902
<i>References</i>	904

Overview

The GREDUCE procedure processes map data sets so that they can draw simpler maps with fewer boundary points. It creates an output map data set that contains all of the variables in the input map data set plus a new variable named DENSITY. For each observation in the input map data set, the procedure determines the significance of that point for maintaining a semblance of the original shape and gives the observation a corresponding DENSITY value.

You can then use the value of the DENSITY variable to create a subset of the original map data set. The observations in the subset can draw a map that retains the overall appearance of the original map but contains fewer points, requires considerably less storage space, and can be drawn much more quickly.

GREDUCE does not produce any graphics output. Instead, it produces an output map data set that can become either

- the input map data set for the GMAP procedure
- the input map data set for a DATA step that removes points from the map.

Figure 24.1 on page 896 and Figure 24.2 on page 896 illustrate the effect of reduction on a typical map data set. Figure 24.1 on page 896 uses observations with all DENSITY values as input to the GMAP procedure.

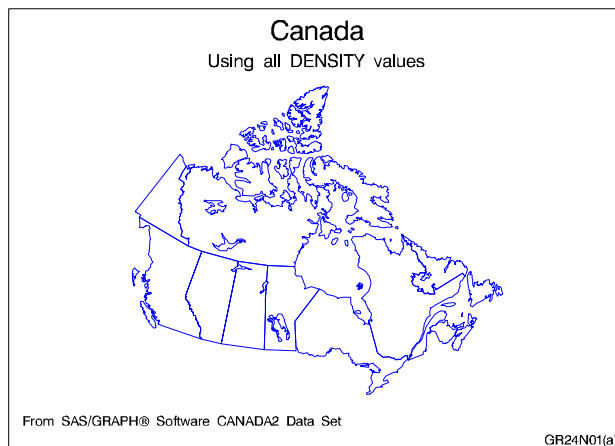
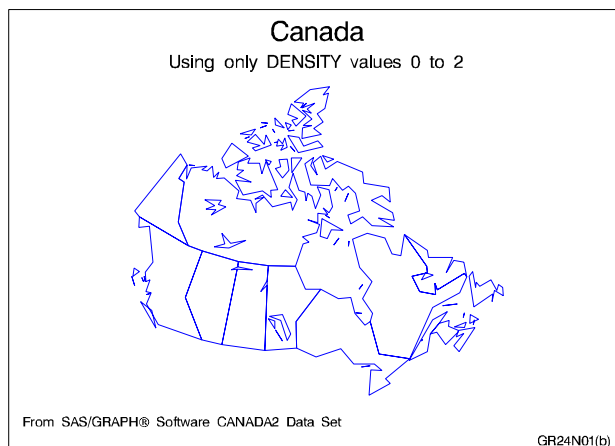
Figure 24.1 CANADA2 Map before Reduction (GR24N01(a))

Figure 24.2 on page 896 uses only those observations with a DENSITY value of 0 or 2 as input to the GMAP procedure.

Figure 24.2 CANADA2 Map after Reduction (GR24N01(b))

The program for these maps is in Example 1 on page 902.

The reduced map shown in Figure 24.2 on page 896 retains the overall shape of the original but requires only 463 observations compared to the 4302 observations that are needed to produce the map in Figure 24.1 on page 896.

Note: Many of the map data sets that are supplied by SAS Institute already have been processed by GREDUCE. If the map data set contains a DENSITY variable, you do not need to process the data set using GREDUCE. \triangle

See also Chapter 25, “The GREMOVE Procedure,” on page 905 for more information on how to

- combine groups of unit areas into larger unit areas to create regional maps
- remove some of the boundaries in a map and create a subset of a map that combines the original areas.

Concepts

About the Input Map Data Set

The input map data set must be a map data set and contain these variables:

- a numeric variable named X that contains the horizontal coordinates of the map boundary points.
- a numeric variable named Y that contains the vertical coordinates of the map boundary points.
- one or more *identification variables* that uniquely identify the unit areas in the map. These variables are listed in the ID statement.

It also can contain

- one or more variables that identify groups of unit areas (for BY-group processing)
- the variable SEGMENT, which distinguishes nonconterminous segments of the unit areas.

Any other variables in the input map data set do not affect the GREDUCE procedure.

About Unmatched Area Boundaries

If you are using map data sets in which area boundaries do not match precisely (for example, if the boundaries were digitized with a different set of points), PROC GREDUCE will not be able to identify common boundaries properly, and this results in abnormalities in your maps. These abnormalities include mismatched borders, missing vertex points, stray lines, gaps, and distorted polygons.

If the points in the area boundaries match up except for precision differences, round each X and Y value in your map data set accordingly, using the DATA step function ROUND before using PROC GREDUCE. (See *SAS Language Reference: Dictionary* for information on the ROUND function.)

For example, if the map data set APPROX has horizontal and vertical coordinate values for interior boundaries of unit areas that are exactly equal only to three decimal places, then this DATA step creates a new map data set, EXACT, that will be better suited for use with PROC GREDUCE:

```
data exact;
    set approx;
    if x ne . then x=round(x,.001);
    if y ne . then y=round(y,.001);
run;
```

See “About Map Data Sets” on page 735 for additional information on map data sets.

Procedure Syntax

Requirements: Exactly one ID statement is required.

Reminder: The procedure can include the BY statement.

PROC GREDUCE <option(s)>;

ID *id-variable(s)*;

PROC GREDUCE Statement

Identifies the input and output map data sets. Optionally specifies the reduction criteria.

Requirements: An input map data set is required.

Syntax

PROC GREDUCE *<option(s)>*;

option(s) can be one or more options from any or all of the following categories:

□ data set options:

DATA=*input-map-data-set*

OUT=*output-map-data-set*

□ level options:

E1=*min-distance*

E2=*min-distance*

E3=*min-distance*

E4=*min-distance*

E5=*min-distance*

N1=*max-points*

N2=*max-points*

N3=*max-points*

N4=*max-points*

N5=*max-points*

Options

DATA=*input-map-data-set*

identifies the map data set that you want to process. By default, the procedure uses the most recently created SAS data set.

See also: “About the Input Map Data Set” on page 897 and “SAS Data Sets” on page 25

E1=*min-distance*

E2=*min-distance*

E3=*min-distance*

E4=*min-distance*

E5=*min-distance*

specify the minimum distance that a point must lie from a straight line segment to be included at density level 1, 2, 3, 4, or 5, respectively. That is, in a reduced curve of

three points, the middle point is at least a distance that is *min-distance* from a straight line between the two outside points.

Express *min-distance* values in the units for the coordinate system of the input map data set. For example, if the input map data set contains coordinates that are expressed in radians, express the *min-distance* values in radians.

Specify the $E_n=$ values in decreasing order. For example, the $E_2=$ value should be less than the $E_1=$ value and so on.

$N_1=$ max-points

$N_2=$ max-points

$N_3=$ max-points

$N_4=$ max-points

$N_5=$ max-points

specify that for density level 1, 2, 3, 4, or 5, the boundary of a unit area should contain no more than *max-points* points.

Specify the $N_n=$ values in increasing order. For example, the $N_2=$ value should be greater than or equal to the $N_1=$ value and so on.

By default, if you omit $N_n=$ and $E_n=$, the GREDUCE procedure calculates values for the five $N_n=$ parameters using this formula:

$$N_n = n^2 \times N_{\max}/36$$

Here N_{\max} is the maximum number of points in any unit area in the input map data set. However, the restriction that the number of points for any level cannot be less than the number of points in level 0 still applies.

OUT=output-data-set

names the new map data set, which contains all of the observations and variables in the original map data set plus the new DENSITY variable. If the input map data set contains a variable named DENSITY, the GREDUCE procedure replaces the values of the variable in the output map data set. The original values of the DENSITY variable from the input map data set are not included in the output map data set.

By default, the GREDUCE procedure names the new data set that uses the $DATA_n$ naming convention. That is, the procedure uses the name $WORK.DATA_n$, where n is the next unused number in sequence. Thus, the first automatically named data set is $DATA_1$, the second is $DATA_2$, and so on.

ID Statement

Identifies the variable or variables that define the hierarchy of the current unit areas in the input map data set.

Requirements: At least one *id-variable* is required.

Featured in: Example 1 on page 902

Syntax

ID *id-variable(s)*;

Required Arguments

id-variable(s)

specifies one or more variables in the input map data set that identify unit areas. *Id-variable(s)* can be either numeric or character.

Each group of observations with a different ID variable value is evaluated as a separate unit area.

Using the GREDUCE Procedure

Specifying Density Levels

GREDUCE uses default criteria for determining the appropriate DENSITY variable value for each observation in the input map data set. If you do not want to use the default criteria, use PROC GREDUCE options to select

- the maximum number of observations for each DENSITY level
- the minimum distance that an intermediate point must lie from a line between two end points to be included in the level.

If you do not explicitly specify criteria, the procedure computes and uses default values.

GREDUCE creates seven density levels, numbered 0 through 6. Specify criteria for density levels 1 through 5. You cannot define criteria for level 0, which is reserved for map vertex points, such as common corners of unit areas. You also cannot define criteria for level 6, which is assigned to those points that do not meet the criteria for any lower level.

Specify the maximum number of observations per density level using $Nn=$ in the PROC GREDUCE statement, and specify the minimum point distance using $En=$. You must have knowledge of the X and Y variable values in the particular input map data set to determine appropriate values for $En=$. See $En=$ and $Nn=$ on page 898 for details.

Figure 24.3 on page 900 illustrates how to use the minimum distance parameter to determine which points belong in a particular density level. At density level n , only point C lies at a distance greater than the $En=$ value (70) from a line between points A and B. Thus, after reduction only point C remains between points A and B at density level n , and the resulting reduced boundary is shown in Figure 24.4 on page 901. See Douglas and Peucker (1973) for details of the algorithm used.

Figure 24.3 Points in Data Set before Reduction

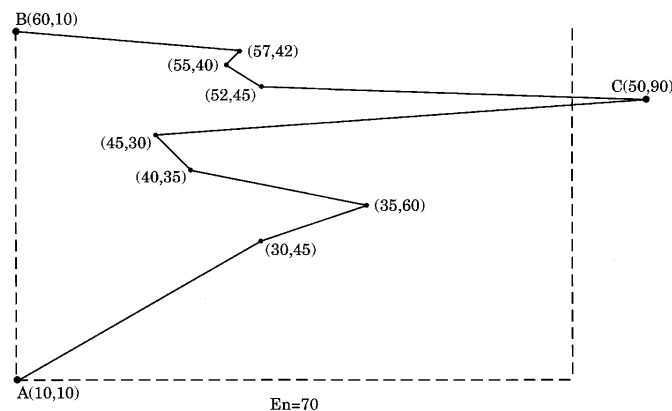
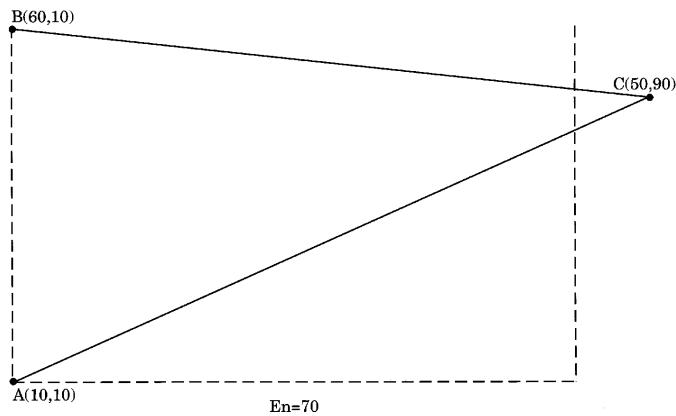


Figure 24.4 Points in Data Set at Density n after Reduction

GREDUCE uses the usual Euclidean distance formula to determine the distance between points. For example, the distance d between the points (x_0, y_0) and (x_1, y_1) is

GREDUCE uses the usual Euclidean distance formula to determine the distance between points. For example, the distance d between the points (x_0, y_0) and (x_1, y_1) is

$$d = \sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2}$$

If this distance function is not suitable for the coordinate system in your input map data set, transform the X and Y values to an appropriate coordinate system before using GREDUCE. An example of inappropriate coordinates is latitude and longitude values around one of the poles. In this case, the data values should be projected before they are reduced. See Chapter 23, “The GPROJECT Procedure,” on page 873 for more information on map projection.

If you specify both $Nn=$ and $En=$ values for a density level, GREDUCE attempts to satisfy both criteria. However, the number of points for any level is never reduced below the number of points in density level 0. If you specify a combination of $Nn=$ or $En=$ values such that the resulting DENSITY values are not in order of increasing density, a note is printed in the SAS log, and the DENSITY values are calculated in increasing order of density.

Subsetting a Map Data Set

A map data set that is processed by GREDUCE does not automatically result in a map that uses fewer points. By default, the GMAP procedure produces a map that uses all of the points in the map data set, even if the data set has been processed by the GREDUCE procedure. To decrease the number of points that produce the map, you must create a subset of the original data set using a DATA step or the WHERE= data set option. For example, to create a subset of a map that uses only the DENSITY values 0, 1, and 2, use this DATA step:

```
data smallmap;
  set map;
  if density <= 2;
```

```
run;
```

Alternatively, you can use WHERE= in the PROC GMAP statement:

```
proc gmap map=map(where=(density<=2))  
        data=response;
```

Note: GREduce does not reduce the size of the output map data set compared to the input map data set. By default, the output map data set from PROC GREduce will be larger than the input map data set because it contains all of the variables and observations from the original data set, with the addition of the DENSITY variable if it was not present in the original data set. If the input map data set already had a DENSITY variable, the output map data set will be the same size as the input map data set. \triangle

Examples

The following example illustrates major features of the GREduce procedure. Because the example uses one of the map data sets that are supplied with SAS/GRAPH, you may need to replace *SAS-data-library* in the LIBNAME statement with the actual location of the SAS data library that contains the Institute-supplied map data sets on your system. Contact your SAS Software Consultant for the location of the map data sets at your site. If your site automatically assigns the libref MAPS to the SAS data library that contains the Institute-supplied map data sets, delete the LIBNAME statement in this example.

Example 1: Reducing the Map of Canada

Procedure features:

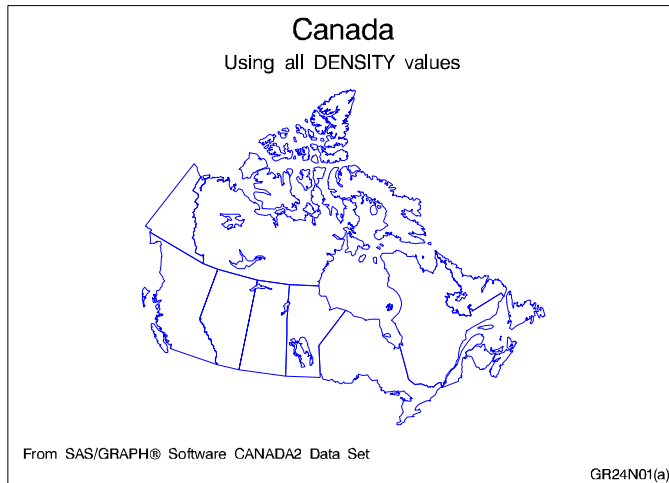
ID statement

Other features:

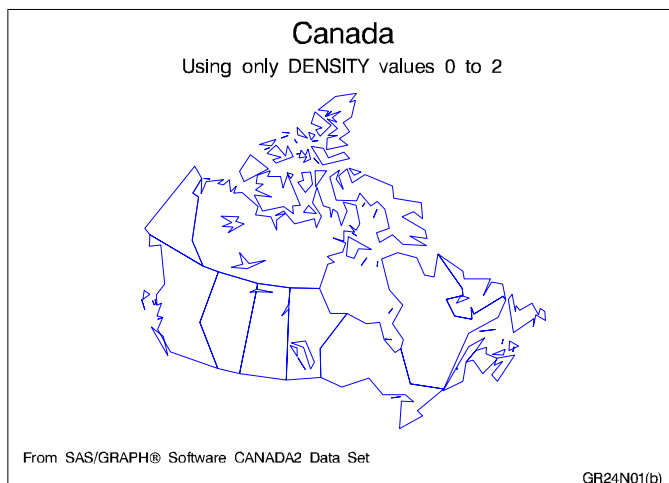
PROC GMAP option:

WHERE=

Sample library member: GR24N01



In this example, the GREDUCE procedure creates the DENSITY variable for the CANADA2 map data set that is provided with SAS/GRAPH. First, the map is displayed at its original density by using the GMAP procedure. Second, the map is displayed by using density values of 0 to 2.



Assign the librefs and set the graphics environment.

```
libname reflib 'SAS-data-library';
libname maps 'SAS-data-library';
goptions reset=global gunit=pct border cback=white
          colors=(black blue green red)
          ftext=swiss htitle=6 htext=3;
```

Define titles and footnotes for the first map.

```
title1 'Canada';
title2 h=4 'Using all DENSITY values';
```

```

footnote1 j=1 ' From SAS/GRAPH
' '02'x
' Software CANADA2 Data Set';
footnote2 j=r 'GR24N01(a) ';

```

Define pattern characteristics.

```

pattern value=memory repeat=12 color=blue;

```

Show the unreduced map. The ID statement specifies the variable in the map data set that defines unit areas.

```

proc gmap map=maps.canada2 data=maps.canada2 all;
  id province;
  choro province / nolegend;
run;

```

The GREDUCE procedure creates a new map data set, REFLIB.CAN2, containing a DENSITY variable. The ID statement specifies the variable in the map data set that defines unit areas.

```

proc greduce data=maps.canada2 out=reflib.can2;
  id province;
run;

```

Define title and footnote for the second map.

```

title2 h=4 'Using only DENSITY values 0 to 2';
footnote2 j=r 'GR24N01(b) ';

```

Show reduced map with density levels 0-2. WHERE= selects map coordinates with the appropriate DENSITY values.

```

proc gmap map=reflib.can2(where=(density<3))
  data=reflib.can2 all;
  id province;
  choro province / nolegend;
run;
quit;

```

References

Douglas, D.H. and Peucker, T.K. (1973), "Algorithms for the Reduction of the Number of Points Required to Represent a Digitized Line or Its Caricature," *The Canadian Cartographer*, 10, 112–122.

The correct bibliographic citation for this manual is as follows: SAS Institute Inc., *SAS/GRAPH® Software: Reference, Version 8*, Cary, NC: SAS Institute Inc., 1999.

SAS/GRAPH® Software: Reference, Version 8

Copyright © 1999 by SAS Institute Inc., Cary, NC, USA.

ISBN 1-58025-525-6

All rights reserved. Printed in the United States of America.

U.S. Government Restricted Rights Notice. Use, duplication, or disclosure of the software by the government is subject to restrictions as set forth in FAR 52.227-19 Commercial Computer Software-Restricted Rights (June 1987).

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513.

1st printing, October 1999

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

OS/2®, OS/390®, and IBM® are registered trademarks or trademarks of International Business Machines Corporation.

Other brand and product names are registered trademarks or trademarks of their respective companies.

The Institute is a private company devoted to the support and further development of its software and related services.