

Using R in the SAS® System

Alan Mitchell, MS, Cancer Research and Biostatistics, Seattle, WA

ABSTRACT

If you use both SAS and R, then you've probably run into a situation where your data are saved as SAS® data sets but the analysis tools you want to use are only available in R. Fortunately, a feature of SAS/IML allows users to exchange SAS data with R and execute R statements using SUBMIT/ENDSUBMIT statements and CALL routines. SAS users can create R data frames from SAS data sets and have access to R's graphics tools along with a large collection of add-on packages. An interactive R programming environment can be initiated from within SAS that is useful for writing and debugging R scripts. In addition to the basic mechanics of the interaction between PROC IML and R, a discussion of some of the pitfalls and a demonstration of some of the tools available will be presented. SAS/IML 9.22 or later is required. However, to use more recent versions of R, newer versions of SAS/IML are required.

INTRODUCTION

The SAS/IML interface to R allows users to take advantage of R within the SAS environment. If you're one of the many analysts, researchers or statisticians that have already learned to use R, you can take advantage of this interface without a lot of extra effort. Using SUBMIT/ENDSUBMIT statements in PROC IML and a few CALL routines, you can create R data frames from SAS data sets and execute R statements within your SAS programs. To create visualizations on the fly, SAS macros can be created for specific plotting functions.

GETTING STARTED

Fortunately, the setup is pretty easy. You will need SAS/IML 9.22 or later, a supported version of R and an environment variable "R_HOME" that points to your R installation (for example, R_HOME=C:\R-2.15.2). You will also need to start SAS from the command line with the -RLANG option or add '-RLANG' to your SASV9.CFG file. Once you've completed these steps, you can execute the code below to make sure SAS is ready to communicate with R.

```
PROC OPTIONS OPTION=RLANG;  
RUN;
```

SAS/IML Version	Supported R Versions
SAS/IML 9.22	R 2.9.1 – 2.11.1
SAS/IML 12.1 or later	R 2.12.1 or later

Table 1. R software supported by different versions of SAS/IML

USING THE INTERFACE

Interaction between SAS and R is not new. Users have done this before, but the solution was not trivial and required exporting SAS data to a delimited text file (See Holland, 'SAS to R to SAS'). With the SAS/IML interface, the process of exchanging data has become much simpler and the execution of code is more interactive. Data exchange is handled by four call routines – ExportDataSetToR, ExportMatrixToR, ImportDataSetFromR and ImportMatrixFromR. In the example below, a matrix is created in PROC IML and made available in R along with the SASHELP.CLASS data set. Note the different use of quotes for IML matrices and SAS data sets. SAS data set names should be enclosed in quotes; IML objects should not.

```
PROC IML;  
X = {1 2 3, 4 5 6, 7 8 9};  
CALL ExportMatrixToR(X, "X");  
CALL ExportDataSetToR("SASHELP.CLASS", "SASData");
```

The SUBMIT and ENDSUBMIT statements define sections of code to be executed outside PROC IML. The area between these two statements is called a SUBMIT block. Any code within a SUBMIT block will be executed in R. The end result is a process half-way between line-by-line execution and batch execution. The R workspace stays open until you execute the QUIT statement and the results of multiple submit blocks are cumulative within an IML session. A scatterplot of height versus weight by sex is generated below from the data previously exported to R. If you are using the SAS Display Manager, a new plot window will be created just as it would in R.

```
SUBMIT /R;
plot(x=SASData$Weight,y=SASData$Height,col=as.factor(SASData$Sex),pch=19)
legend('bottomright',inset=.05,legend= c('F','M'),pch=19,col=c(1,2))
ENDSUBMIT;
```

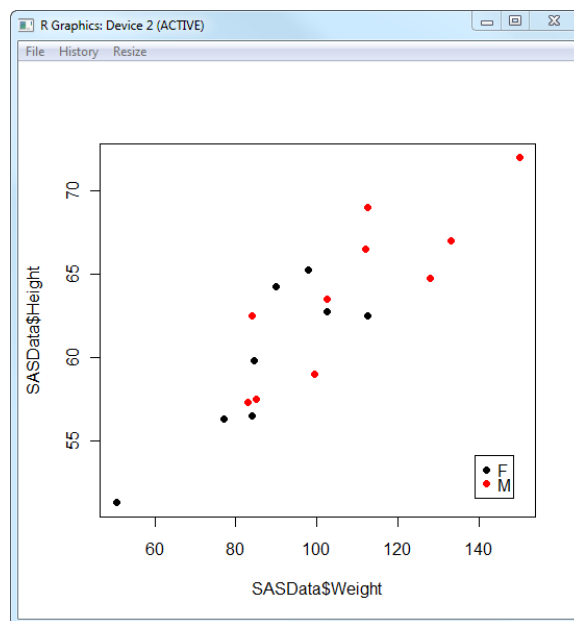


Figure 1. Scatterplot of Height versus Weight by Sex

SAS MACROS CONTAINING R SCRIPTS

SAS macros can be used to provide access to common plotting functions in R functions like `plot()` to make quick visualizations. There's a catch though; `SUBMIT/ENDSUBMIT` statements are not allowed in SAS macros. Try it yourself and you'll find the following error in your LOG.

ERROR: Submit block cannot be directly placed in a macro. Instead, place the submit block into a file first and then use `%include` to include the file within a macro definition.

This solution is easy to implement, although probably not ideal. In the macro definition below, IML is started and the data is exported to R within the macro. This allows you to create new plots quickly. You can use R expressions like `log(x)` or `sin(x)` rather than variable names to explore data transformations without an additional DATA step. The "PlotParms" variable lets you specify other plot parameters without having to define each variable in the macro.

```
%MACRO PLOT(ds=, x=, y= ,PlotParms="");
PROC IML;

CALL EXPORTDATASETTOR("&ds","rdata");
ds="&ds";
x="&x";
y="&y";
plotparms = &PlotParms;
print x y;

%INCLUDE "H:\R in SAS Demo\plotr.sas";

%MEND PLOT;
```

The variables defined in the `%PLOT` macro are stored as scalar variables in IML. The IML variables are passed to R through the `SUBMIT` statement. Additional processing of these variables can be performed in IML prior to passing the values to R. For instance, you might use `TRANSLATE` to make paths interpretable by R (convert `"\"` to `"/`). The values passed to R should be referenced with the same ampersand syntax used for macro variables in SAS.

```

SUBMIT ds x y PlotParms /R;
names(rdata) = tolower(names(rdata))
attach(rdata)
plot(x=&x, y=&y, main=toupper("&ds"),&plotparms)
detach(rdata);
ENDSUBMIT;

```

Creating a simple scatterplot requires only three parameters. The QUIT statement following the macro call is intentionally commented out. Executing the QUIT statement would terminate the IML and R sessions and close the plot window. This way, it is still easy to execute the QUIT statement at any time, but it takes much more deliberate action.

```

%PLOT(ds=sashelp.class, x=height, y=weight);
*quit;

```

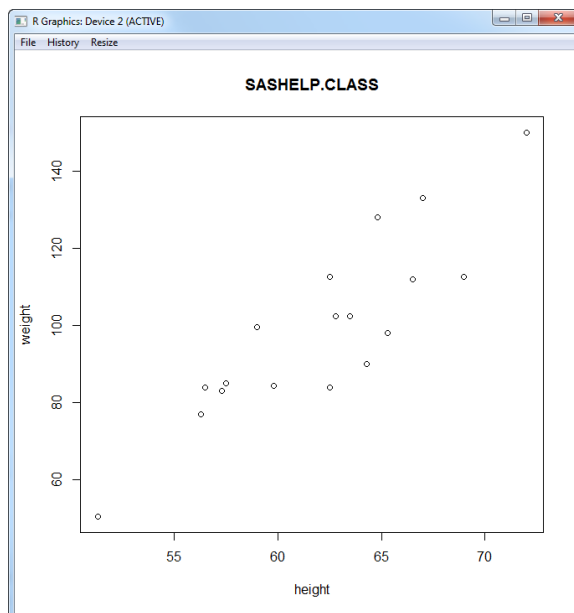


Figure 2. Scatterplot of Height versus Weight using %PLOT macro

DEBUGGING IN AN INTERACTIVE ENVIRONMENT

When debugging, you sometimes need to run your code line by line to find the problem. SUBMIT blocks aren't really conducive to this type of work. One way to work around this is to use the R Commander environment. This is an interactive interface built as a package in R and it's great for debugging R code from SAS.

```

submit /R;
library(Rcmdr)
endsubmit;

```

Loading this library will bring up a new window with the R Commander environment. Any data exported from SAS to R using the PROC IML CALL routines will be available immediately and the R Commander environment will reflect any previously executed statements. The package was originally designed for use as an educational tool for introductory statistics courses, but since it can be opened from a SUBMIT block, I have found it to be a generally useful interface. A picture of the R Commander window is shown in the Appendix.

TAKING ADVANTAGE OF ADDITIONAL PACKAGES

The R package system, including a voluminous collection of packages developed by the user community, is one of the most attractive features of R. In addition to being a convenient way to create collections of tools, R packages provide a framework for documentation that makes the tools easier to use. Most packages tend to have a very narrow scope, but may contain a large number of tools to perform a specific analysis or accomplish a specific task. The RgoogleMaps package is an excellent example of one of the highly specialized sets of tools available through the R package system.

A cholera outbreak in London in 1854 led to the creation of a remarkable map by the physician John Snow. This map has been recreated multiple times by multiple people because it is such a great example of how important the graphical representation of data can be. Given the locations of the nearby pumps and the locations and frequencies of deaths, this map can be recreated using satellite images available from Google Maps. The data for this example was compiled by Robin Wilson and is available from his blog. SAS code is supplied in the appendix that will recreate the data sets used for this example.

The first step to using a new package is installation. This is a rather simple process and can be done from within SAS. After beginning a PROC IML session, you can use a SUBMIT block to run the R function that installs the package. If you do not specify a repository and you haven't defined a default repository in your R system options you will be asked to choose from a long list of potential mirrors. The code below uses the default repository, but finding the closest mirror to you is considered a best practice. This step does not need to be repeated until you upgrade to a new version of R.

```
proc iml;

submit /R;
install.packages('RgoogleMaps', repos="http://cran.us.r-project.org");
endsubmit;
```

After installing the package, you are ready to export the SAS data sets created using the code in the appendix. For convenience, the data frames in R will have the same name as the data sets from SAS.

```
call ExportDataSetToR('pumps','pumps');
call ExportDataSetToR('deaths','deaths');
```

Unlike SAS, R is a case sensitive language. The first thing I like to do with any SAS data set exported to R is to change all the variable names to lower case. You can see this in the first two lines of the SUBMIT block below. Before starting the actual map making, the RgoogleMaps package must be loaded and the working directory set so we know where to look for any output later. This is much easier than using full paths for every file reference.

After the preliminaries are done, the GetMap function queries a server to download a map image based on a location and zoom you provide. You can find this information ahead of time by finding the map you want on the Google Maps web page. The latitude and longitude coordinates as well as the zoom will be in the URL. The next few calls to PlotOnStaticMap load the map image and plot the deaths and pumps on the map. A few adjustments were necessary to the longitude and latitude coordinates to make a better fit on the map.

```
submit /R;

names(pumps) = tolower(names(pumps));
names(deaths) = tolower(names(deaths));

library('RgoogleMaps')
setwd("C:/Users/AlanM/SnowMap")

mymap = GetMap(center=c(51.513,-0.135), zoom = 16, destfile = "map.png",
                format="PNG", maptype = "hybrid", frame=TRUE);

png('SnowMap.png',width=1200,height=1200,pointsize=18)
PlotOnStaticMap(mymap)

PlotOnStaticMap(mymap,FUN=points,add=TRUE,
                lat=deaths$lat-.00015,
                lon=deaths$long-.00015,
                col=rgb(1, 1, 0, .5),pch=19,cex=deaths$psize)
PlotOnStaticMap(mymap,FUN=points,add=TRUE,
                lat=pumps$lat-.00015,
                lon=pumps$long-.00015,
                col=rgb(1, 0, 0),pch=18,cex=2.5)

dev.off()

endsubmit;
```

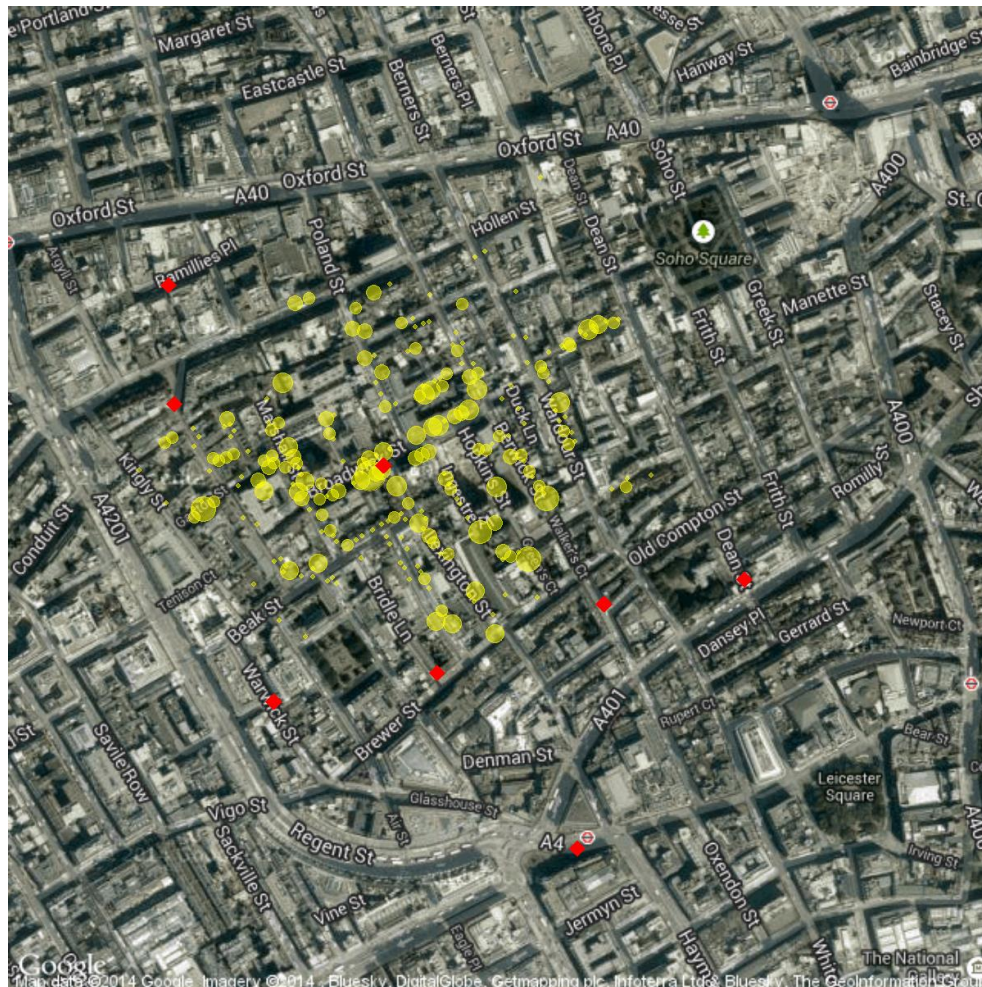


Figure 3. Recreation of John Snow's cholera map overlaid on a map of modern London

CONCLUSION

The SAS/IML interface to R can be used to generate quick visualizations and publication-ready graphics. SAS macros can provide quick access to R's plotting functions and more complex visualizations can be created using SUBMIT blocks directly in your SAS programs. Whether you are looking for an additional graphical tool or a new suite of analysis tools, the SAS/IML interface to R allows users to take advantage of everything R has to offer.

REFERENCES

R Core Team (2013). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL <http://www.R-project.org/>.

Loecher, Markus (2014). RgoogleMaps: Overlays on Google map tiles in R. R package version 1.2.0.6. <http://CRAN.R-project.org/package=RgoogleMaps>

Wilson, R. "Updated Snow GIS data." Robin's Blog. (September 25, 2012) Available at <http://blog.rtwilson.com/updated-snow-gis-data/>

Google Maps (2014). [Hybrid map]. Retrieved from <http://maps.googleapis.com/maps/api/staticmap?center=51.513,-0.135&zoom=16&size=1200x1200&mapttype=hybrid>

RECOMMENDED READING

- R Graphics, Second Edition, CRC Press, Paul Murrell
- Calling Functions in the R Language, SAS/IML® 9.22-13.1 User's Guide

CONTACT INFORMATION

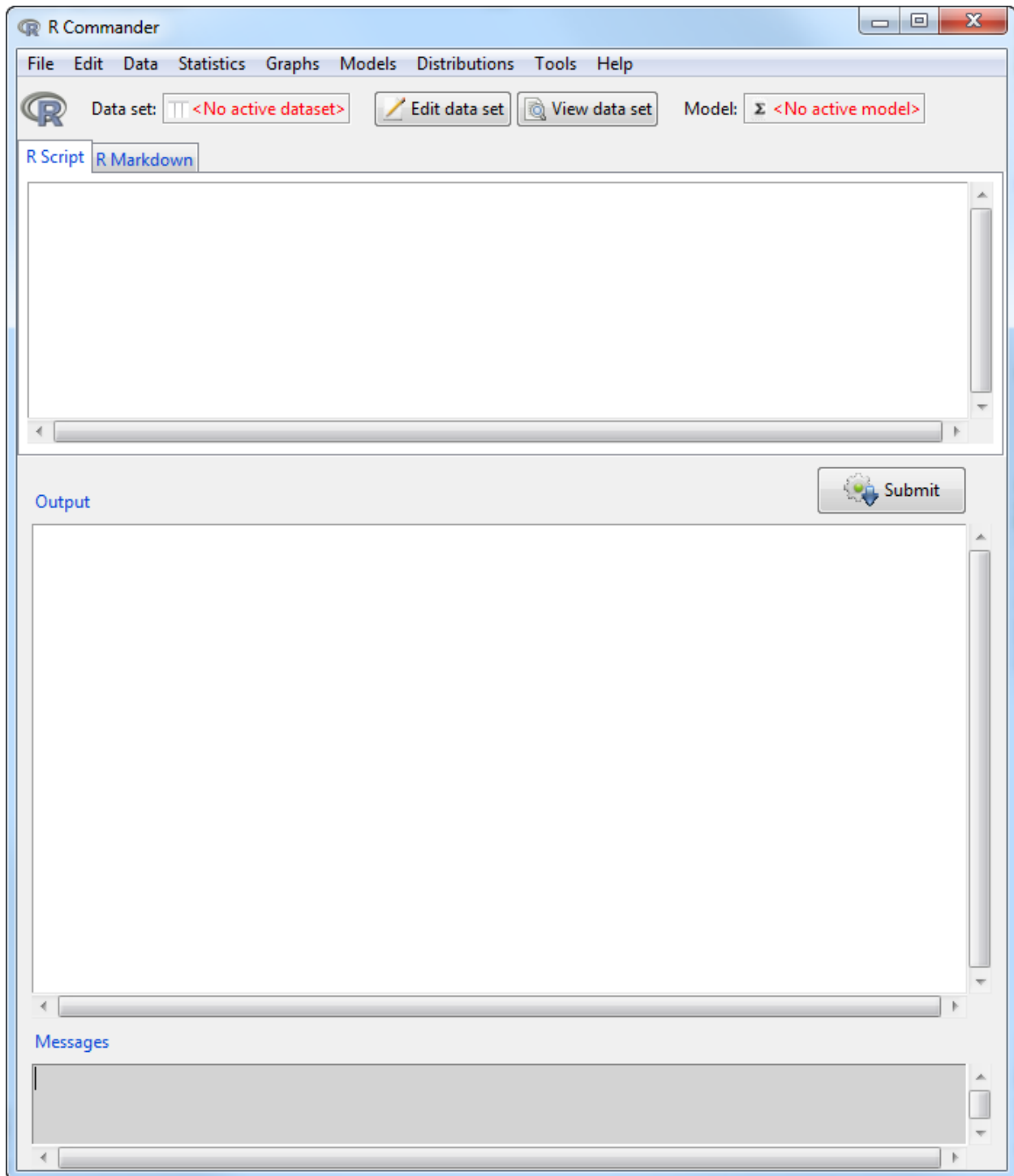
Your comments and questions are valued and encouraged. Contact the author at:

Name: Alan Mitchell, MS
Enterprise: Cancer Research and Biostatistics
Address: 1730 Minor Ave, Suite 1900
City, State ZIP: Seattle, WA 98101
Work Phone: (206) 839-1708
E-mail: alanm@crab.org
Web: <http://www.linkedin.com/in/alannmitchell/>

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.

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

APPENDIX – R COMMANDER



APPENDIX – ADDITIONAL CODE

CHOLERA DATA

```

data pumps;
input lat long best12.;
datalines;
51.51343153      -0.136462137
51.51396652      -0.13938074
51.51499622      -0.139465561
51.51244425      -0.131423584
51.51222911      -0.133388372
51.51163252      -0.135713397
51.51010934      -0.133755825
51.51138543      -0.137993287
;
run;

data deaths;
input lat long deaths @@;
psize = .8 + log(deaths);
datalines;
51.51350816  -0.137724505  3  51.51253721  -0.137450286  1  51.51234042  -0.13518804  4
51.51345184  -0.137677478  2  51.51258142  -0.137378651  4  51.51225201  -0.135203362  1
51.51340748  -0.137647307  1  51.51246385  -0.13744472  1  51.51230252  -0.135266279  1
51.51335256  -0.137606348  1  51.51242978  -0.137244076  1  51.51266311  -0.135559664  2
51.5132944  -0.137561218  4  51.51245446  -0.137170342  1  51.51266563  -0.135665499  1
51.51327397  -0.137331307  2  51.51240897  -0.137121561  1  51.51276211  -0.135769876  2
51.51344963  -0.137994913  2  51.51263075  -0.136774224  1  51.51281686  -0.135827116  3
51.51341847  -0.137839419  2  51.51273895  -0.136974819  2  51.5128841  -0.135909378  1
51.5134134  -0.138070595  3  51.51278276  -0.136846588  1  51.51293629  -0.135974475  4
51.51351713  -0.138017921  2  51.51304694  -0.137489677  1  51.51296983  -0.135877221  1
51.51347164  -0.138131486  2  51.51285504  -0.137327351  1  51.51302915  -0.135933668  1
51.5135528  -0.138357184  1  51.5128705  -0.137213397  2  51.51285559  -0.135122845  7
51.51330636  -0.138220061  3  51.51281654  -0.137162051  1  51.51293428  -0.1349162  3
51.51325987  -0.138172908  1  51.51277157  -0.137119263  1  51.5126223  -0.134438771  8
51.51320689  -0.138131611  4  51.51300428  -0.137325097  2  51.51228791  -0.134315607  1
51.51333005  -0.138439943  1  51.51313682  -0.137356084  2  51.51230499  -0.134761006  1
51.51325485  -0.138492406  1  51.51316402  -0.137260326  1  51.51324393  -0.134892223  5
51.5132686  -0.137718691  1  51.51317757  -0.137180422  2  51.51314635  -0.134187589  8
51.51320161  -0.137659318  4  51.51321213  -0.137100248  3  51.51325579  -0.134502721  2
51.51314584  -0.137605886  3  51.51327744  -0.136883237  1  51.51318818  -0.134230817  1
51.5135318  -0.138556418  2  51.51330487  -0.136790811  4  51.51332845  -0.134387631  1
51.5136829  -0.138593334  1  51.51333978  -0.136653253  15  51.51338349  -0.134434508  2
51.5134928  -0.138839214  2  51.51336164  -0.136571908  3  51.51346962  -0.134502721  1
51.51347003  -0.13876429  2  51.51339042  -0.13649948  4  51.513521  -0.134550233  2
51.51350124  -0.138657931  2  51.5132499  -0.13628779  5  51.51356506  -0.135038478  2
51.51373194  -0.138546982  1  51.51310628  -0.136123845  2  51.51351199  -0.134691353  3
51.51378392  -0.138602649  1  51.51301163  -0.136217804  1  51.51361851  -0.134952009  1
51.51383538  -0.13865052  3  51.51298049  -0.136317617  2  51.51357179  -0.135138269  2
51.51376659  -0.138681365  1  51.51294986  -0.136393737  1  51.51368439  -0.134857449  2
51.51368014  -0.139033902  1  51.5129208  -0.136493189  1  51.5133173  -0.135595117  3
51.51375357  -0.139115823  1  51.51287207  -0.13661291  1  51.51327018  -0.135555774  1
51.51359279  -0.139110552  1  51.51281935  -0.136767041  1  51.51322241  -0.135534359  2
51.51367306  -0.139410544  2  51.51295885  -0.136152767  1  51.51313817  -0.135439095  1
51.51363181  -0.139513911  2  51.51281305  -0.13642419  1  51.51309634  -0.135395865  1
51.51338801  -0.139868462  1  51.51274475  -0.136378701  1  51.51297285  -0.135295527  1
51.51338098  -0.138888557  1  51.51280327  -0.136216821  1  51.51336016  -0.13562629  1
51.51310376  -0.139491812  1  51.51270546  -0.136139355  1  51.51354894  -0.135842997  3
51.51305518  -0.139121662  1  51.51258124  -0.136231693  1  51.51352172  -0.135933826  3
51.51298366  -0.139111969  2  51.51253931  -0.13617152  1  51.51349247  -0.136022037  3
51.51305457  -0.138981382  8  51.51255498  -0.135991129  1  51.51268329  -0.134793423  3
51.51311526  -0.138830484  2  51.51250344  -0.135935804  1  51.51267492  -0.134587023  1
51.5131171  -0.139003675  1  51.51244832  -0.135895928  2  51.51264483  -0.134690226  2
51.51292135  -0.138221492  1  51.51236177  -0.13582443  1  51.51261108  -0.134794063  1
51.51297496  -0.138418592  1  51.5124454  -0.136104032  1  51.51322754  -0.133276776  1
51.51261681  -0.137890195  1  51.51208122  -0.135734652  4  51.51331846  -0.133059099  1
51.51255557  -0.137829019  1  51.5121728  -0.135652054  2  51.51324273  -0.133090044  2
51.51251853  -0.137778006  4  51.5121211  -0.135594287  1  51.51334794  -0.132727359  1
51.51250498  -0.137859319  1  51.51205996  -0.13551097  4  51.51363451  -0.13379201  1
51.51259991  -0.137988241  1  51.51197183  -0.134913344  4  51.51371652  -0.13383586  1

```



```

51.51246829 -0.137612112 1 51.51214007 -0.134938554 1 51.51372753 -0.133950088 2
51.51361445 -0.133885228 2 51.51467661 -0.135824463 1 51.51346847 -0.137024817 1
51.51390994 -0.134066275 1 51.51470256 -0.136060676 1 51.51394499 -0.136444785 2
51.5138142 -0.134014246 1 51.51466545 -0.136215545 2 51.51396505 -0.136296885 1
51.51379415 -0.1344981 1 51.51459745 -0.136728809 3 51.51365538 -0.137160922 1
51.51392131 -0.134576358 1 51.5143644 -0.136725064 3 51.51370667 -0.137216223 2
51.51400587 -0.134804529 1 51.51438306 -0.136593454 1 51.51383272 -0.137266592 3
51.51368707 -0.134717258 1 51.51414894 -0.136573848 1 51.51400872 -0.138094539 1
51.51412255 -0.134678727 1 51.51423842 -0.136490143 3 51.51386288 -0.137156987 1
51.51398165 -0.134006334 5 51.51405178 -0.136506439 1 51.51359265 -0.137789525 4
51.51384799 -0.133929206 1 51.51411717 -0.13591741 1 51.51380263 -0.137933672 2
51.51415524 -0.134158153 1 51.51416612 -0.135752468 1 51.51373429 -0.138033771 2
51.51423661 -0.134240736 2 51.51418604 -0.135676802 1 51.51380156 -0.138066245 1
51.51429141 -0.134273384 2 51.51422469 -0.135581844 2 51.51415176 -0.13787719 5
51.51432046 -0.134451809 1 51.51412372 -0.135643548 2 51.51483859 -0.137706028 3
51.51440914 -0.134161244 1 51.51408688 -0.135802381 4 51.51488487 -0.137501429 2
51.51446756 -0.133972611 1 51.51405043 -0.135892944 5 51.51461664 -0.136902003 3
51.51444773 -0.133953736 1 51.51403519 -0.135963787 2 51.51479646 -0.136858806 2
51.51447252 -0.133862666 2 51.5139114 -0.13527917 5 51.51240124 -0.138268051 1
51.51449219 -0.133879089 2 51.51408914 -0.135168128 5 51.51208875 -0.137917641 1
51.51461206 -0.133615343 5 51.51388539 -0.135375949 3 51.51194598 -0.137556594 1
51.51458793 -0.133716216 1 51.5138569 -0.135473157 3 51.51431469 -0.135209268 1
51.51456293 -0.133643695 1 51.51381598 -0.135608626 1 51.51431055 -0.135370458 1
51.51459475 -0.133519176 1 51.51378201 -0.13569876 5 51.51423565 -0.135150756 2
51.51463632 -0.13353924 1 51.51376289 -0.135786321 4 51.51419861 -0.135269507 3
51.51465173 -0.133469629 4 51.51369295 -0.136011065 4 51.51444952 -0.136020282 2
51.51468437 -0.133357362 1 51.51357218 -0.136372992 1 51.51441635 -0.136122314 1
51.51467164 -0.1332604 2 51.51354864 -0.136469341 4 51.51463449 -0.136016308 1
51.51469606 -0.133186747 1 51.51351962 -0.136558616 1 51.51465996 -0.135911597 1
51.51592516 -0.134268069 1 51.51349468 -0.136671255 3 51.51493524 -0.134612261 1
51.51528539 -0.13505305 1 51.51344975 -0.136747241 2 51.51447934 -0.135497664 1
51.51523952 -0.135189381 1 51.51458606 -0.135447072 1 51.51448977 -0.135355443 1
51.51490885 -0.13581638 1 51.51483382 -0.135372034 2 51.51442557 -0.135442794 2
51.51493408 -0.136598332 3 51.51455755 -0.13465366 1 51.51454357 -0.134483575 1
51.51500429 -0.136377547 1
;
run;

```