SAS® Cloud Analytic Services 3.1: Graphing Your Output

SAS® Documentation

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2016. *SAS® Cloud Analytic Services 3.1: Graphing Your Output*. Cary, NC: SAS Institute Inc.

SAS® Cloud Analytic Services 3.1: Graphing Your Output

SAS Institute Inc., SAS Campus Drive, Cary, NC 27513-2414

September 2016

3.1-P1:casdatagraph

Contents

Chapter 1

Graphing Your CAS Output

Graphing Data That Is Processed in CAS

Use Cases for Graphing Data That Is Processed in CAS

You can generate graphs using data that is processed in SAS Cloud Analytic Services (CAS).

There are two use cases for graphing the results of your analysis:

- An analytical procedure might provide an option for generating a graph when the procedure is run. To determine whether a graphing option is available for a particular procedure, see the documentation provided for the procedure.
- For procedures that do not provide the option to generate a graph, first run the analytical procedure as usual. Then, use an ODS Graphics procedure to create the graph from the analytical procedure's output. For more information, see "Create Graphs of CAS Data Using the ODS Graphics Procedures" on page 1.

In either case, you can use the ODS GRAPHICS statement to customize graph output. For more information, see "ODS GRAPHICS Statement" in *SAS Viya ODS Graphics: Procedures Guide*.

Create Graphs of CAS Data Using the ODS Graphics Procedures

You can use the ODS Graphics procedures to graph CAS data that is available to the SAS client. This topic describes availability in more detail later.

There are three ODS Graphics procedures that you can use to graph an analytical procedure's output:

- SGPLOT creates single-cell plots with a variety of plot and chart types and overlays.
- SGPANEL creates classification panels for one or more classification variables. Each graph cell in the panel can contain either a simple plot or multiple, overlaid plots.
- SGSCATTER creates scatter plot panels and scatter plot matrices with optional fits and ellipses.

Here are the main steps for using the ODS Graphics procedures with data that has been analyzed in the cloud:

1. Run the analytical procedure, making sure that the ODS Graphics procedures can access the output CAS data. The examples provided use the CAS engine LIBNAME statement for this purpose.

   The CAS LIBNAME engine is a bi-directional engine. You can read data from CAS and you can add data to CAS. Large volumes of data can be processed in CAS at in-memory speeds and then used by the ODS Graphics procedures to perform data visualization.

   Note: Some analytical procedures can summarize or reduce the amount of data. If you are working with large amounts of data, you should summarize or reduce the data before attempting to graph it. Doing this improves the performance of your program.

2. Run an ODS Graphics procedure and specify the data set you created in step one.

Examples

Example 1: Graph Clustering Analysis Results Obtained from the KCLUS Procedure

The KCLUS procedure places objects into groups, or clusters. Objects in a given cluster tend to be similar to each other in some sense, and objects in different clusters tend to be dissimilar. The KCLUS procedure supports k-means and k-modes clustering models.

Example Code 1.1 *Program*

```
libname mycas cas;                                        /* 1 */

proc casutil;                                             /* 2 */
  load data=sashelp.iris;
quit;

proc kclus data=mycas.iris outstat=mycas.centroids        /* 3 */
       seed=12345 maxclusters=3;
  input sepallength sepalwidth petallength petalwidth / level = interval;
  outputtables clustersum=clustersummary iterstats=iterationstatistics;
run;

ods graphics / width=4in;                                  /* 4 */
title "Decreasing SSE Values";
proc sgplot data=mycas.iterationstatistics;                /* 5 */
  step x=iterationnum y=iterationsse;
```

```
  xaxis label="Iterations";
  yaxis type=log label="SSE Values";
run;
title;
ods graphics / reset=all;
```

1 The CAS engine LIBNAME statement assigns the Mycas libref. The CAS engine is a bi-directional engine. You can read data from CAS and you can add data to CAS.

2 The CASUTIL procedure is used to load data into the CAS memory.

3 The KCLUS procedure clusters the data. The procedure uses the OUTPUTTABLES statement to output the cluster summary and the iteration statistics to CAS tables.

4 The ODS GRAPHICS statement is used to specify the width of the graph. The height is sized automatically so that the default aspect ratio of the graph is maintained.

5 The SGPLOT procedure accesses the ITERATIONSTATISTICS table in CAS by means of the CAS engine LIBNAME statement shown in line 1 of the program. The SGPLOT procedure then creates a step plot that shows the iteration statistics. For each iteration, the procedure plots the sum of squared errors (SSE) using a logarithmic scale.

The KCLUS procedure generates the following series of tables:

Output 1.1 *KCLUS Procedure Output*

The KCLUS Procedure

Number of Observations Read	150
Number of Observations Used	150

Model Information	
Maximum Iterations	10
Stop Criterion	Cluster Change
Stop Criterion Value	0
Clusters	3
Initialization	Forgy
Seed	12345
Distance	Euclidean
Standardization	None
Interval Imputation	None

Cluster Summary for Interval Variables

Cluster	Frequency	Distance from Cluster Centroid to Observation			SSE	Standard Deviation	Nearest Cluster	Distance to Nearest Cluster Centroid
		Minimum	Maximum	Average				
1	38	2.5958	15.2971	7.1984	2387.9	7.9272	3	17.9718
2	50	0.6618	12.4803	4.8171	1515.1	5.5047	3	33.5693
3	62	2.1994	16.6064	7.3815	3982.1	8.0142	1	17.9718

Iteration History			
Iteration Number	SSE	SSE Change	Stop Criterion
0	148310		
1	33280	-115030	51.333333
2	8998.263899	-24282	6.666667
3	7994.304172	-1003.959727	2.666667
4	7891.005674	-103.298498	0.666667
5	7885.144143	-5.861531	0

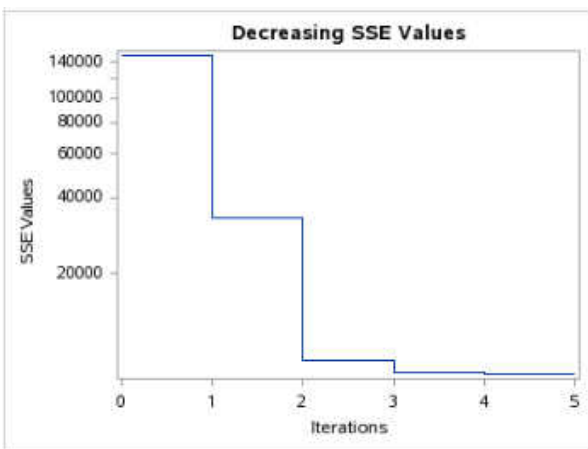
Descriptive Statistics		
Variable	Mean	Standard Deviation
SepalLength	58.433333	8.280661
SepalWidth	30.573333	4.358663
PetalLength	37.580000	17.652982
PetalWidth	11.993333	7.622377

Within Cluster Statistics			
Variable	Cluster	Mean	Standard Deviation
SepalLength	1	68.5000	4.9416
	2	50.0600	3.5249
	3	59.0161	4.6641
SepalWidth	1	30.7368	2.9009
	2	34.2800	3.7906
	3	27.4839	2.9628
PetalLength	1	57.4211	4.8859
	2	14.6200	1.7366
	3	43.9355	5.0889
PetalWidth	1	20.7105	2.7987
	2	2.4600	1.0539
	3	14.3387	2.9750

Output CAS Tables			
CAS Library	Name	Number of Rows	Number of Columns
CASUSERHDFS	CLUSTERSUMMARY	3	9
CASUSERHDFS	ITERATIONSTATISTICS	6	4

The SGPLOT procedure generates the following graph:

Output 1.2 *Graph Output*



For more information, see the following:

- "CASUTIL" in *SAS Cloud Analytic Services: Language Reference*
- KCLUS procedure in *SAS Visual Data Mining and Machine Learning: Statistical Procedures*
- "SGPLOT" in *SAS Viya ODS Graphics: Procedures Guide*

Example 2: Graph Summary Statistics Results Obtained from the MDSUMMARY Procedure

The MDSUMMARY procedure generates descriptive statistics of numeric variables such as the sample mean, sample variance, sample size, sum of squares, and so on.

Example Code 1.2 *Program*

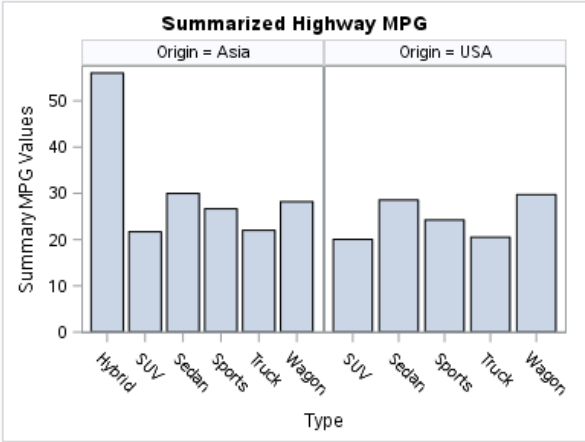
```
libname mycas cas;

proc casutil;
  load data=sashelp.cars;
quit;

proc mdsummary data=mycas.cars;                    /* 1 */
  var mpg_highway;
  groupby origin type / out=mycas.mpghw_sum;
run;

ods graphics / width=4in;
title "Summarized Highway MPG";
proc sgpanel data=mycas.mpghw_sum;                 /* 2 */
  where origin in ("Asia" "USA");
  panelby origin / uniscale=row;
  format _mean_ 2.;
  vbarparm category=type response=_mean_;
  rowaxis label="Summary MPG Values";
run;
title;
ods graphics / reset=all;
```

1 The MDSUMMARY procedure produces summary statistics for highway miles-per-gallon. The OUT= option in the GROUPBY statement creates a table in CAS. The table includes a row of summary statistics for each unique combination of origin and type.

2 The SGPANEL procedure plots the summarized results from the MDSUMMARY procedure. The procedure creates a parameterized vertical bar chart that shows the mean statistic for highway miles-per-gallon. The procedure subsets the data, comparing only the cars made in Asia and the U.S.A. The graph is paneled by country of origin.

The SGPANEL procedure generates the following graph:

Output 1.3 *Graph Output*



For more information, see the following:

- "CASUTIL" in *SAS Cloud Analytic Services: Language Reference*
- "MDSUMMARY" in *SAS Cloud Analytic Services: Language Reference*
- "SGPANEL" in *SAS Viya ODS Graphics: Procedures Guide*

Example 3: Graph Linear Regression Results Obtained from the REGSELECT Procedure

This example uses the REGSELECT procedure to perform model selection for ordinary linear least squares models. The procedure supports backward, forward, stepwise, LAR, and lasso selection.

Example Code 1.3 *Program*

```
libname mycas cas;

data mycas.analysisData;                                  /* 1*/
array x{50} x1-x50;
  do i=1 to 1000;
    y=rannor(1);
    do j=1 to dim(x);
      x{j}=ranuni(1);
      if ranuni(1) < .2 then y=y+j*x{j};
    output;
    end;
  end;
run;

proc regselect data=mycas.analysisData;                   /* 2*/
  ods output SelectionSummary=SelectionSummary;
  model y = x1-x50;
  selection method = stepwise  hierarchy=single;
run;

ods graphics / width=4in;
title "SBC Statistics";
```

```
proc sgplot data=SelectionSummary;                    /* 3 */
  format sbc comma.;
  scatter x=step y=sbc;
run;
title;
ods graphics / reset=all;
```

1 The DATA step is used to generate random data.

2 The REGSELECT procedure performs stepwise selection using the default SBC selection criterion. The procedure then outputs the selection summary statistics.

This example does not out save the output table in CAS. For information about saving the output table in CAS, see the Note at the end of this example.

3 The SGPLOT procedure creates a scatter plot of the SBC statistic for each step.

The REGSELECT procedure generates the following tables:

Output 1.4 *REGSELECT Procedure Output*

The REGSELECT Procedure

Model Information	
Data Source	ANALYSISDATA
Response Variable	y

Selection Information	
Selection Method	Stepwise
Select Criterion	SBC
Stop Criterion	SBC
Effect Hierarchy Enforced	Single
Stop Horizon	3

Number of Observations Read	50000
Number of Observations Used	49951

Dimensions	
Number of Effects	51
Number of Parameters	51

Selection Details

Selection Summary			
Step	Effect Entered	Number Effects In	SBC
0	Intercept	1	382197.969
1	x14	2	382162.648
2	x16	3	382139.240
3	x22	4	382117.445
4	x21	5	382102.368
5	x37	6	382089.022
6	x17	7	382077.932
7	x25	8	382067.399
8	x31	9	382058.485
9	x20	10	382051.110
10	x27	11	382043.910
11	x23	12	382037.854
12	x28	13	382031.841
13	x13	14	382026.593
14	x15	15	382022.116
15	x18	16	382016.465
16	x7	17	382013.495
17	x34	18	382011.521*

Analysis of Variance					
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	17	776096	45653	21.86	<.0001
Error	49933	104283573	2088.47000		
Corrected Total	49950	105059668			

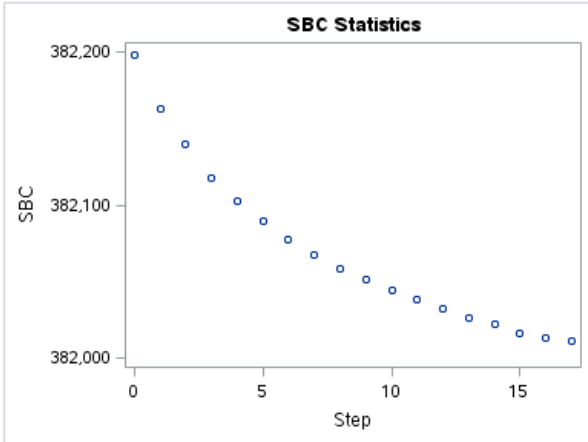
Root MSE	45.69978
R-Square	0.00739
Adj R-Sq	0.00705
AIC	431806
AICC	431806
SBC	382012
ASE	2087.71742

Parameter Estimates					
Parameter	DF	Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	15.675976	1.526462	10.27	<.0001
x7	1	2.663791	0.707450	3.77	0.0002
x13	1	2.958612	0.735792	4.02	<.0001
x14	1	4.826960	0.715715	6.74	<.0001
x15	1	3.048105	0.715825	4.26	<.0001
x16	1	3.825909	0.715491	5.35	<.0001
x17	1	3.311469	0.712968	4.64	<.0001
x18	1	2.987045	0.714958	4.18	<.0001
x20	1	3.424873	0.714287	4.79	<.0001
x21	1	4.139878	0.732033	5.66	<.0001
x22	1	3.715440	0.721956	5.15	<.0001
x23	1	3.181115	0.709151	4.49	<.0001
x25	1	3.325882	0.696630	4.77	<.0001
x27	1	3.367023	0.711014	4.74	<.0001
x28	1	3.104349	0.708038	4.38	<.0001
x31	1	3.032301	0.706502	4.29	<.0001
x34	1	2.631468	0.735796	3.58	0.0003
x37	1	3.121256	0.713808	4.37	<.0001

Task Timing		
Task	Seconds	Percent
Setup and Parsing	0.04	43.17%
Levelization	0.02	19.67%
Model Initialization	0.00	0.29%
SSCP Computation	0.02	24.40%
Model Selection	0.00	3.01%
Cleanup	0.00	4.53%
Total	0.09	100.00%

The SGPLOT procedure generates the following graph:

Output 1.5 *Graph Output*



Note: The procedure uses the ODS OUTPUT statement to output the summary table rather than save the table in CAS. This method is useful for moderate amounts of data that you want to use only for local plotting. You might not want to save such data in CAS. However, if you do want to save the data in CAS, you can instead use the OUTPUTTABLES statement that is described in Example One. The current example might be altered as follows. Highlighted text indicates the parts that are changed.

```
proc regselect data= mycas.analysisData;
  outputtables selectionSummary=selectSumm;
  model y = x1-x50;
  selection method = stepwise  hierarchy=single;
run;

ods graphics / width=4in;
title "SBC Statistics";
proc sgplot data=mycas.selectSumm;
  scatter y=sbc x=step;
run;
title;
ods graphics / reset=all;
```

For more information, see the following:

- "CASUTIL" in *SAS Cloud Analytic Services: Language Reference*
- REGSELECT procedure in *SAS Visual Data Mining and Machine Learning: Statistical Procedures*
- "SGPLOT" in *SAS Viya ODS Graphics: Procedures Guide*