SAS® Cloud Analytic Services
3.1: Graphing Your 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**

```latex
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;
```

---
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.

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

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

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.

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**

![Table of cluster summary for interval variables](image)

The table shows the cluster summary for interval variables, including the distance from cluster centroid to observation, SSE, standard deviation, and nearest cluster centroid.
The SGPLOT procedure generates the following graph:

Output 1.2  Graph Output

For more information, see the following:
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**

```sas
libname mycas cas;

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

proc mdssummary 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**

![Graph Output](image)

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**

```sas
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";
```
The DATA step is used to generate random data.

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

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

The SG PLOT 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

```
proc sgplot data=SelectionSummary; /* */
    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 output 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 SG PLOT procedure creates a scatter plot of the SBC statistic for each step.

The REGSELECT procedure generates the following tables:
### Selection Details

#### Selection Summary

<table>
<thead>
<tr>
<th>Step</th>
<th>Effect Entered</th>
<th>Number Effects in</th>
<th>SDC</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Intercept</td>
<td>1</td>
<td>382197.969</td>
</tr>
<tr>
<td>1</td>
<td>x14</td>
<td>2</td>
<td>382162.046</td>
</tr>
<tr>
<td>2</td>
<td>x16</td>
<td>3</td>
<td>382136.240</td>
</tr>
<tr>
<td>3</td>
<td>x22</td>
<td>4</td>
<td>382117.445</td>
</tr>
<tr>
<td>4</td>
<td>x21</td>
<td>5</td>
<td>382102.368</td>
</tr>
<tr>
<td>5</td>
<td>x37</td>
<td>6</td>
<td>302068.022</td>
</tr>
<tr>
<td>6</td>
<td>x17</td>
<td>7</td>
<td>382077.032</td>
</tr>
<tr>
<td>7</td>
<td>x25</td>
<td>8</td>
<td>382067.399</td>
</tr>
<tr>
<td>8</td>
<td>x31</td>
<td>9</td>
<td>382056.485</td>
</tr>
<tr>
<td>9</td>
<td>x20</td>
<td>10</td>
<td>382005.110</td>
</tr>
<tr>
<td>10</td>
<td>x27</td>
<td>11</td>
<td>382043.910</td>
</tr>
<tr>
<td>11</td>
<td>x23</td>
<td>12</td>
<td>382037.854</td>
</tr>
<tr>
<td>12</td>
<td>x28</td>
<td>13</td>
<td>382031.841</td>
</tr>
<tr>
<td>13</td>
<td>x13</td>
<td>14</td>
<td>382026.593</td>
</tr>
<tr>
<td>14</td>
<td>x15</td>
<td>15</td>
<td>382022.118</td>
</tr>
<tr>
<td>15</td>
<td>x18</td>
<td>16</td>
<td>382016.665</td>
</tr>
<tr>
<td>16</td>
<td>x7</td>
<td>17</td>
<td>382013.495</td>
</tr>
<tr>
<td>17</td>
<td>x34</td>
<td>18</td>
<td>382011.521</td>
</tr>
</tbody>
</table>

### Analysis of Variance

<table>
<thead>
<tr>
<th>Source</th>
<th>DF</th>
<th>Sum of Squares</th>
<th>Mean Square</th>
<th>F Value</th>
<th>Pr &gt; F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>17</td>
<td>776086</td>
<td>45653</td>
<td>21.88</td>
<td>&lt;.0001</td>
</tr>
<tr>
<td>Error</td>
<td>49595</td>
<td>104283573</td>
<td>2088.47000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Corrected Total</td>
<td>49595</td>
<td>135299666</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Root MSE: 45.69973
- R-Square: 0.00739
- Adj R-Sq: 0.00703
- AIC: 431906
- AICC: 431506
- SBC: 382012
- ASE: 2357.71742

### Parameter Estimates

| Parameter | DF | Estimate | Standard Error | t Value | Pr > |t| |
|-----------|----|----------|----------------|---------|------| |   |
| Intercept | 1  | 15.875976 | 1.526462       | 10.27   | <.001| | |
| x7        | 1  | 2.653791  | 0.707450       | 3.77    | 0.002| | |
| x13       | 1  | 2.965012  | 0.735792       | 4.02    | <.001| | |
| x14       | 1  | 4.828069  | 0.715715       | 6.74    | <.001| | |
| x15       | 1  | 3.048105  | 0.715825       | 4.26    | <.001| | |
| x16       | 1  | 3.025908  | 0.715491       | 4.35    | <.001| | |
| x17       | 1  | 3.311469  | 0.712968       | 4.64    | <.001| | |
| x18       | 1  | 2.987045  | 0.714958       | 4.18    | <.001| | |
| x20       | 1  | 3.424873  | 0.714297       | 4.79    | <.001| | |
| x21       | 1  | 4.130678  | 0.732033       | 5.66    | <.001| | |
| x22       | 1  | 3.715440  | 0.721955       | 5.15    | <.001| | |
| x23       | 1  | 3.011115  | 0.709151       | 4.49    | <.001| | |
| x25       | 1  | 3.325882  | 0.696693       | 4.77    | <.001| | |
| x27       | 1  | 3.387023  | 0.711044       | 4.74    | <.001| | |
| x28       | 1  | 3.158438  | 0.700993       | 4.38    | <.001| | |
| x31       | 1  | 3.032301  | 0.706502       | 4.29    | <.001| | |
| x34       | 1  | 2.631488  | 0.735795       | 3.58    | 0.0033| | |
| x37       | 1  | 3.121256  | 0.713505       | 4.37    | <.001| | |
The SGPLOT procedure generates the following graph:

**Output 1.5  Graph Output**

![Graph Output](image)

**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.

```sas
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*