Quick Start

Orientation

This software supports analytical data preparation, variable transformations, exploratory analysis, analytical modeling, integrated model comparison, and scoring. Here are the main components:

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAS Studio</td>
<td>The integrated programming environment.</td>
</tr>
<tr>
<td>SAS Cloud Analytic Services (CAS)</td>
<td>The analytic engine. CAS uses high-performance, multi-threaded analytic code to rapidly process requests against data of any size.</td>
</tr>
</tbody>
</table>
| SAS Visual Data Mining and Machine Learning | CAS based procedures for:  
- analyzing complex data  
- building predictive models  
- advanced statistical operations |

CAS based procedures run against data that is in CAS. For example, before you can use CAS to work with a SAS data set, you must load that data set into CAS. The following instructions demonstrate basic mechanics.

Demonstration: Load Personal Data

1. Sign in to SAS Studio.
Demonstration: Provide Shared Data

1. Sign in to SAS Studio and start a CAS session, if you have not already done so.
2. Create a container for shared CAS data.
   a. In the navigation pane, open the Snippets section.
   b. Select Snippets ⇒ Cloud Analytic Services.
   c. Right-click New Caslib for Path and select Open.
   d. In the code editor, edit the snippet so that it looks like this:

```
CASLIB demoCas PATH="/FilePath/*" DATASOURCE=(SRCTYPE="path") GLOBAL;
```

**Note:** Enter a path that is relative to and accessible from your CAS server. You can reference an empty directory.
Click to run the code.

Note: If an error indicates that you do not have permission to create global caslibs, see “Adjust Caslib Management Privileges” in SAS Viya Administration: SAS Cloud Analytic Services. Initially, only administrators can add global caslibs. An administrator can enable non-administrators to add global caslibs.

3 Give all users Read access to the new caslib.
   a In the banner, click CAS Administration.
   b In the sign-in window, enter your operating system credentials.
   c In CAS Server Monitor, beneath the SAS Cloud Analytic Services banner, click
   d On the Configuration page, select Access Controls.
   e In the Caslibs list, select the caslib.
   f In the upper right, click Edit.
   g In the Edit Access Controls window, adjust settings as follows:
      ■ In the Read Info row for Authenticated Users, select the Grant radio button.
      ■ Click Add Row. In the new row at the end of the page, select Authenticated Users, the Grant radio button, and the Select activity.
   h Click OK to save your changes.
   i Under Access Controls, review the results of your changes.
   j At the right edge of the banner, click your user name, and select Sign Out.

4 Load data to the new caslib.
   a In the navigation pane in SAS Studio, right-click the snippet Load data to Caslib and select Open.
   b In the code editor, edit the SAS data set section so that it looks like this:

```
PROC CASUTIL;
   LOAD DATA=sashelp.cars OUTCASLIB="demoCas"
   CASOUT="demoTable" PROMOTE;
RUN;
```
   c Select the preceding code. In the toolbar, click to run only the four lines of selected code.

5 Verify that other users can see the data. For example, ask them to complete these steps:
   a Sign in to SAS Studio, start a CAS session, and run the Generate SAS librefs for Caslibs snippet.
   b In the navigation pane, open the Libraries section.
   c Select My Libraries → demoCas.
   d Double-click demoTable to open it.

**Tip: Automatically Connect and Generate Librefs**

For convenience, you can configure SAS Studio to perform the following tasks each time you sign in:
4

- Start a CAS session.
- Generate SAS librefs for existing caslibs that have names that are no more than eight characters long.

Complete the following steps:

1. In the SAS Studio banner, click , and select **Edit Autoexec File**.

2. On the **Autoexec.sas** tab of the Edit Autoexec File window, paste the following code:
   
   ```
   cas casauto;
   caslib _all_ assign;
   ```

3. Run the code.

4. Save the code.

### Documentation: References by Task

<table>
<thead>
<tr>
<th>Task</th>
<th>Refer to:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Do you need to migrate your data to UTF-8?</td>
<td><em>Migrating Data to UTF-8 for SAS Viya</em></td>
</tr>
</tbody>
</table>
| Access your data | “Accessing Data” in *SAS Cloud Analytic Services: Accessing and Manipulating Data*  
“Common Tasks for Accessing and Manipulating Data” in *SAS Cloud Analytic Services: Accessing and Manipulating Data* |
| Manipulate your data | “DATA Step Basics” in *SAS Cloud Analytic Services: Accessing and Manipulating Data*  
“Common Tasks for Accessing and Manipulating Data” in *SAS Cloud Analytic Services: Accessing and Manipulating Data* |
| Prepare, Model, Assess | “MDSUMMARY” in *SAS Cloud Analytic Services: Language Reference*  
*SAS Visual Data Mining and Machine Learning: Data Mining and Machine Learning Procedures*  
*SAS Visual Data Mining and Machine Learning: Statistical Procedures* |
| Graph your output data | “Graphing Your CAS Output” in *SAS Cloud Analytic Services: Graphing Your Output* |
| Program with the Lua, Python, or CASL languages | *SAS Cloud Analytic Services: Getting Started with Lua*  
*Getting Started with CASL*  
*SAS Cloud Analytic Services: Getting Started with Python* |