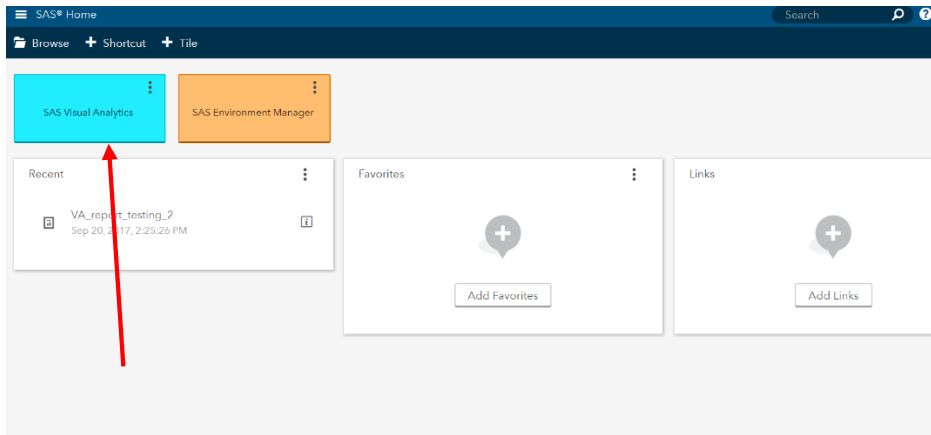


Visual Analytics

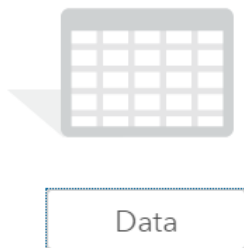
Logging In

1. Access the SAS Viya visual interface through the link provided to you.
2. Enter the credentials that have been provided to you to login.
3. Click on the 'Visual Analytics' shortcut shown on the home screen.

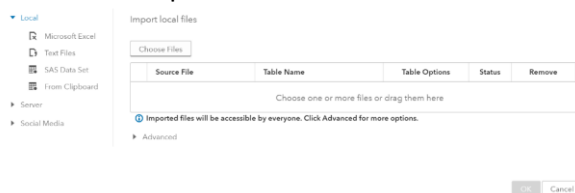


Loading/Unloading Your Tables

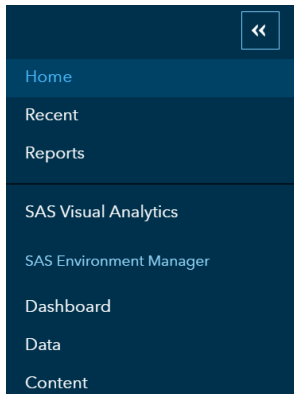
4. Choose the 'Data' option when prompted to open a data source (if you already have a report saved you can choose the open option).



5. You will see that data has already been loaded into Visual Analytics for your use, if you would like to use any of these tables, select the one(s) you wish to use or look at.
6. If you decide you would like to import your own data source click the 'Import' option and then select the type of data you will be importing. Click 'Choose Files' to browse specific the file explorer. You can upload multiple tables at the same time. Under the 'Status' column information will be given on whether the data is able to be uploaded, hovering your mouse gives details on the specific status



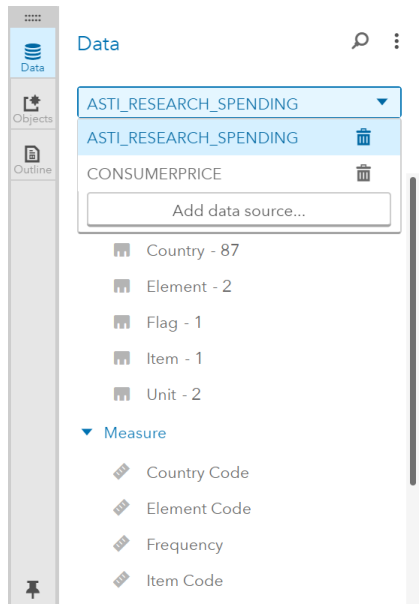
7. If there are any of your own tables you have loaded into SAS Viya and have realized they will no longer be needed, you can unload them from Visual Analytics, which will free up space in-memory. To do this, navigate to the left pane and click 'Manage Data'. You can unload any tables that will not be needed here by clicking the green lightning bolt so it whites out.



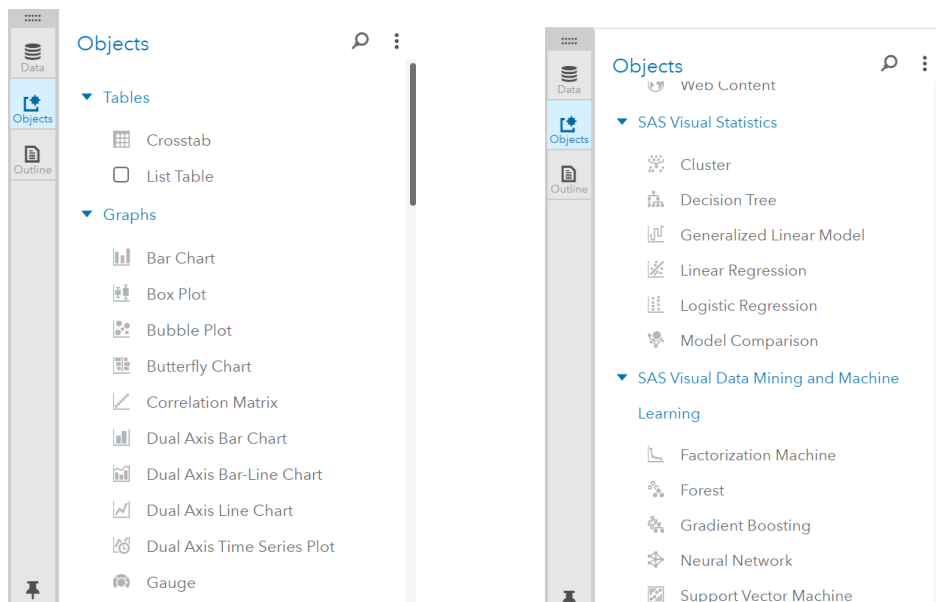
Navigating the Interface

8. Now that data has been imported, clicking the 'Data' tab within the pane on the left side of the screen (when in the exploration/visualization window) will allow you to see the columns/variables within your table(s). If multiple tables are uploaded, you will be able to scroll between them as well as remove specific tables at will. Dragging and dropping multiple or a

single data variable onto the center canvas will enable SAS' auto chart feature which will pick an appropriate visualization for whatever type of data is selected.



- The 'Object' tab on the left pane allows you to select various controls, graphs, machine learning techniques, classical statistical techniques as well as other objects and map them onto the center canvas. Once an object is placed in the center canvas, data from the 'Data' tab can populate most objects by dragging and dropping.



- On the right pane, there are several tabs that can be clicked on. All tabs are related to altering various properties of objects that have been placed in the centre canvas. To interact with any of these tabs, you first must click on one of your objects.

The 'Options' tab allows you to go and change the properties for the specified object.

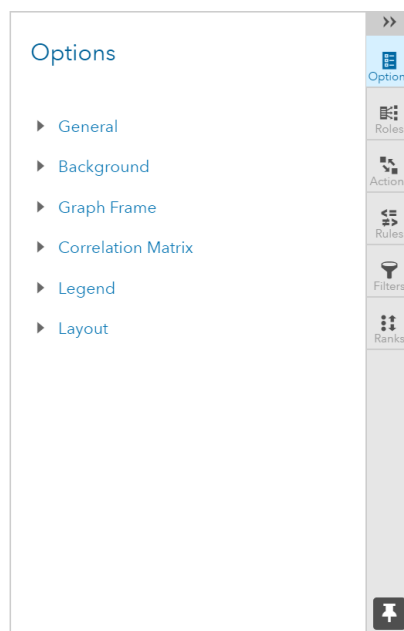
The 'Roles' tab allows you to manipulate what variables are to be included in a specific object.

The 'Actions Tab' allows you to do various actions, such as linking a webpage related to the report or linking/highlighting the same data in two different objects.

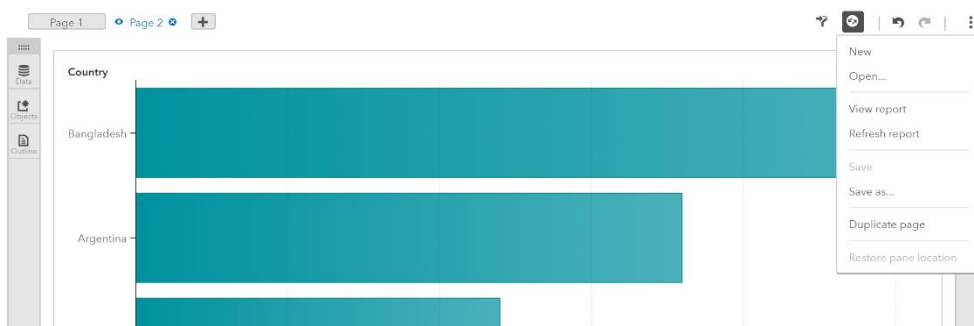
The 'Rules' tab enables you to populate intervals, add intervals, or add color-mapped values for the report object that is currently selected in the canvas. You can use this pane to specify both report-level or object-level display rules, depending on what you have selected in the canvas.

The 'Filters' tab allows you to create filters so that you can subset your data. Can subset the data for each individual object or report wide.

The 'Ranks' tab allows you to rank your data in a report object to show the greatest count or percent or the bottom count or percent for a category that's based on a measure.



11. When you are ready to save your report, look to navigate to the 3 vertical buttons (snowman) to locate both 'Save as', and once a file location has been specified, 'Save'. You can also open existing reports or create new ones from here.



SAS Studio (Coding Interface)

Logging In

1. Access SAS Studio through <https://ic.cac.queensu.ca/SASStudio>

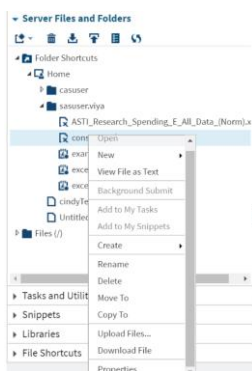
2. Enter the credentials that have been provided to you to login.

Navigating the Interface

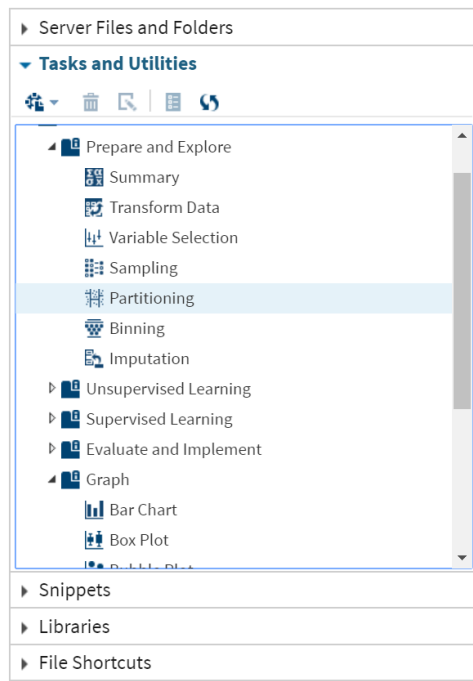
3. You will notice a row of bars with several tabs that resembles the picture below. The running person allows you to run either all or just the selected code. You can also save your program from here. There is also an undo/redo feature. The far-right button enables you to go to minimize the pane on the left side of the interface. Hovering over any of these buttons with your mouse allow you to see what each one does. The 'Log' tab displays any errors, warnings, and notes that may appear once code has been run, pay close attention to these (even the notes). The 'Results' tab is where your output will be displayed once you have run your code.



- Looking at the left portion of the interface, you will notice several tabs. The 'Server Files and Folders' tab allows you to navigate any files/folders available on the server. You can also upload or download any files that are located locally into the server from here.

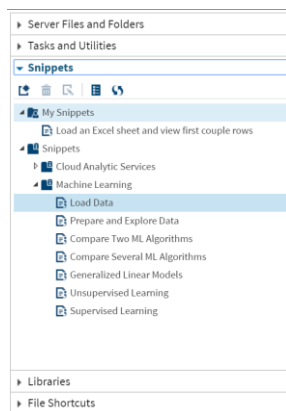


5. The 'Tasks and Utilities' tab provides a drop down list of various data preparation/exploration/analysis tasks. Double-clicking on any one of these will auto-generate the appropriate code for you, assuming you have a valid data source selected.

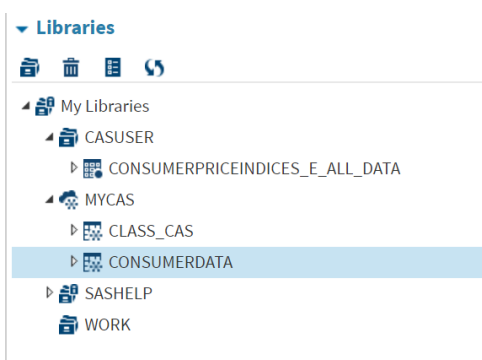


- The 'Snippets' tab allows you to utilize prewritten pieces of code. You can use the ones that have been provided to you for both CAS actions and machine learning. You can also write and save your own snippets that will be available for use.

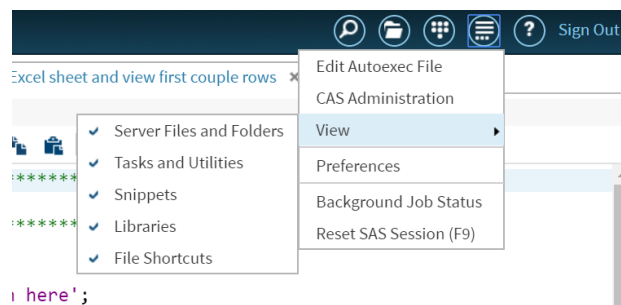
A snippet has been provided to you that will allow you to load an Excel sheet (xlsx) into SAS, start a CAS session, and view the first ten rows of said table. All you have to do is double click a snippet, specify your specific file path/table names and hit run. This snippet can also be used to import other file types with a couple quick modifications.



- The 'Libraries' tab allows you to view all tables that have been loaded into a library. Double clicking on a table will open it for you in SAS Studio and allow you to navigate and filter it.



8. The top panel has several buttons, the file folder icon allows you to open an existing SAS program. The icon selected in the picture below provides several options, including one to reset your SAS Session, which can be useful when running into issues.



Bringing in Your Data

9. When you are ready to load your external data into SAS Studio, navigate to the 'Server Files and Folders' tab on the left pane. Click the 'Upload' button, and then specify which folder you want your file to be uploaded to. Click 'Choose Files' to bring up a file explorer that enables you to navigate and choose the file(s) you would like to upload.
10. You can use the Snippet that has been provided below which will import your file to a library, start a CAS (the engine that SAS runs on behind the scenes) session, load the file to a specified caslib and then create a table that examines the first ten rows. You're also free to code this yourself, or to use a compilation of other snippets that can achieve the same result. The snippet provided is for a.xlsx file, but with a simple modification and a correctly specified path, it can easily be used to load other filetypes such as.csv.
11. Now that your tables are loaded into SAS, you are free to either code or use snippets/tasks to explore and analyze your data!

Snippet to Load Excel File

```
/* **** */
/* Import your XLSX (Excel) Files */
/* **** */
```

```
LIBNAME CASuser XLSX 'Your file path here';
```

```
/******  
/* Start a session named mySession using the existing CAS server connection */  
/* while allowing override of caslib, timeout (in seconds), and locale */  
/* defaults. */  
/******  
cas mySession sessopts=(caslib=casuser);
```

```
libname mycas cas caslib=casuser;
```

```
/******  
/* Load file from a client location ("pathToClientFile") into the specified */  
/* caslib ("myCaslib") and save it as "tableNameForLoadedFile". The PROMOTE */  
/* option makes loaded data available to all active sessions. */  
/******
```

```
proc casutil;
```

```
    load file="(Your file path here-identical to above filepath)"
```

```
    outcaslib="casuser" casout="Name your output here";
```

```
run;
```

```
/******  
/* Look at first 10 rows */  
/******
```

```
proc print data = mycas.outputnamefromabove (obs = 10);  
run;
```