# Snowflake and SAS Viya Machine Learning Step-by-Step Tutorial: Understanding Medicare (CMS) Opioids and Social Determinants of Health Data for Risk Stratification

Last update: October 15, 2020

## CONTENTS



## MEDICARE OPIOIDS USE CASE

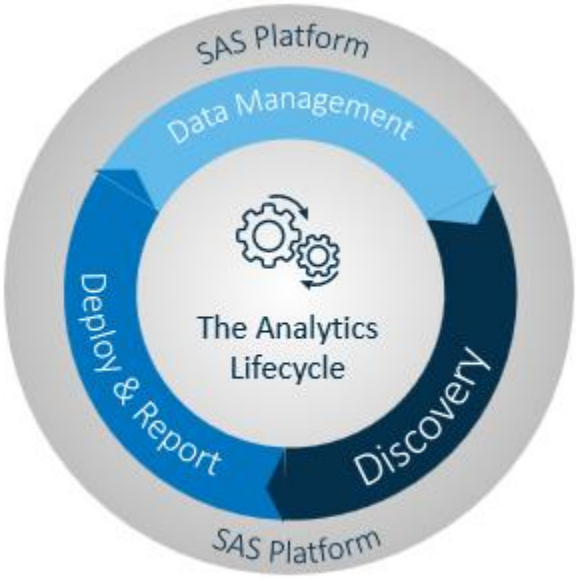
This tutorial is meant to jump start one's familiarity with Python and SAS Viya (especially machine learning using SAS Studio and SAS Visual Statistics). SAS Viya provides several visualizations to help users gain insights into their data. In this tutorial, we examine Opioids mortality & morbidity data, CMS Part D Opioid Prescribing Rate data in the United States along with social determinants of health (SDOH). High utilization of opioids in Medicare Part D may be an indication of OUD (Opioid Use Disorder) among beneficiaries and even fraud among providers. For a detailed analysis of this use case, please review the following:

https://www.sas.com/content/dam/SAS/support/en/sas-global-forum-proceedings/2018/2141-2018.pdf

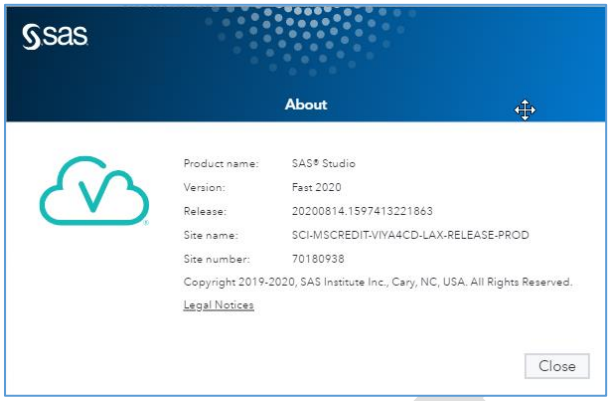
In addition to the experience with machine learning, this tutorial is meant to help users understand Medicare data. Although the use case is focused primarily on Medicare Part D and SDOH, the target variables (e.g., Part D Opioid Prescribing Rate, Drug Poisoning Deaths, etc) can be replaced with other variables related to quality measure, utilization, costs, etc. Likewise, the observations in the dataset are segmented or clustered using SDOH variables, but this can be substituted with other variables related to claims, clinical data, etc.

## ANALYTICS LIFECYCLE

Whatever data is used, an effective methodology to begin to transform data into actionable insights is the analytics lifecycle:




This tutorial will guide readers on this analytics lifecycle journey using Viya 2020:




Viya 2020 is the next generation platform for SAS with the following capabilities to better handle machine learning workloads:

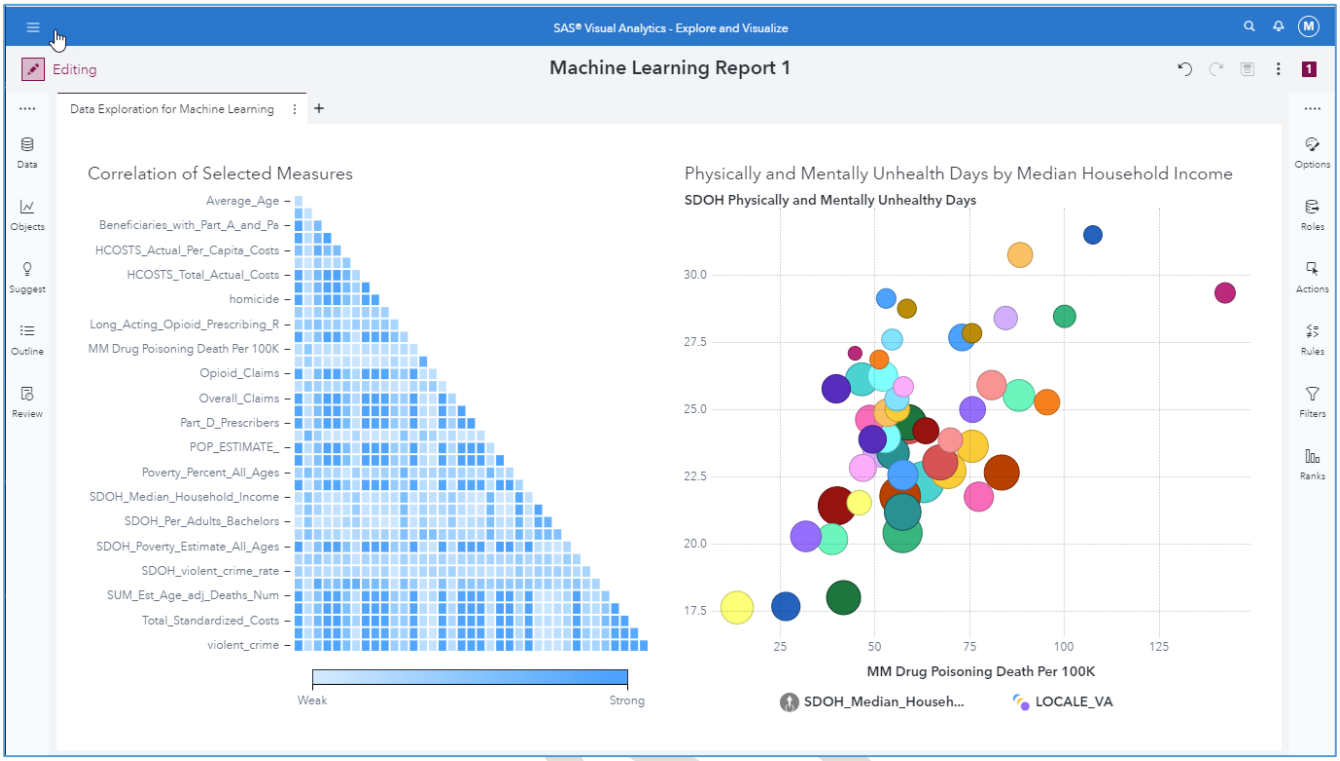
- Cloud-native and Microsoft Azure-ready.
- Containerization with Kubernetes
- CI/CD

The code and documentation is available at:

https://github.com/sasgovernment

Click on the folder link “Step-by-Step”. We will be accessing the data using APIs and code.

The ultimate goal is to produce an exploration as follows:



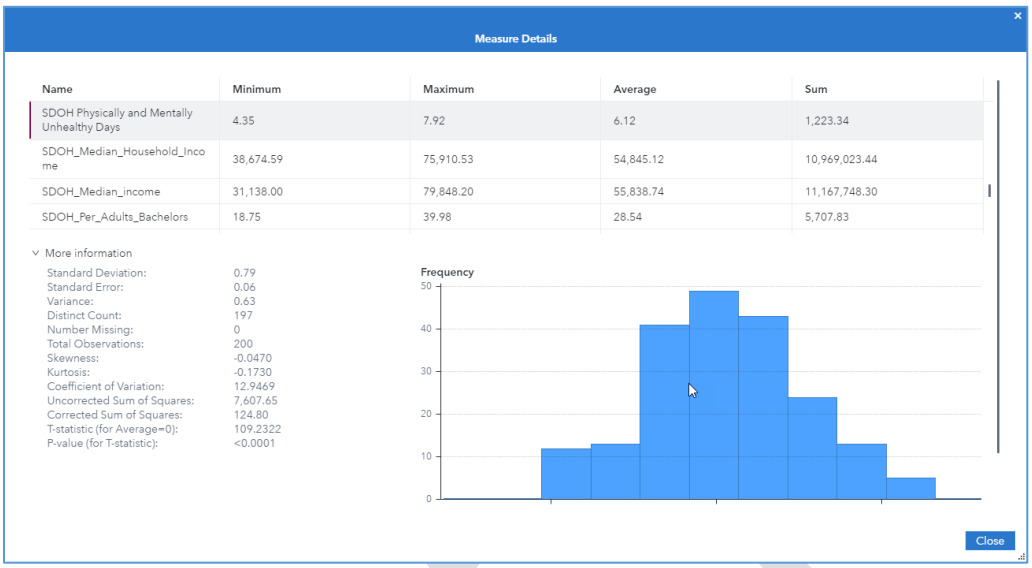

Screen 1. Data Exploration for Machine Learning.

## DATA AND ANALYTICS QUESTION

The data is derived from Medicare Part D and includes primarily two sources:

- CCW - The CMS Chronic Conditions Data Warehouse (CCW) provides researchers with Medicare and Medicaid beneficiary, claims, and assessment data linked by beneficiary across the continuum of care. In the past, researchers analyzing data files were required to perform extensive analysis related to beneficiary matching, deduplication, and merging of the files in preparation for their study analysis. With the CCW data, this preliminary linkage work is already accomplished and delivered as part of the data files sent to researchers. The Chronic Conditions Data Warehouse (CCW) is a research database designed to make Medicare, Medicaid, Assessments, and Part D Prescription Drug Event data more readily available to support research designed to improve the quality of care and reduce costs and utilization.
- BRFSS - The Behavioral Risk Factor Surveillance System (BRFSS) is a state-based system of telephone health surveys that collects information on health risk behaviors, preventive health practices, and health care access primarily related to chronic disease and injury. The survey was established in 1984. Data are collected monthly in all 50 states, Puerto Rico, the U.S. Virgin islands, and Guam.

Key indicators include:




Screen 2. Measure Details for "Physically and Mentally Unhealthy Days".

Name of the dataset and data variables:

- myclouddata_wide.sas7bdat.
- It can be retrieved using the Data Management Flow or by selecting the dataset from the "data" folder in the GITHUB repository.

The Analytic question that we will pursue is:

> How can Jupyter Notebooks or SAS Viya be used to understand the risk of opioid deaths in Medicare using SDOH (Social Determinants of Health) data?

Prerequisites:

- Python 3.X (https://www.python.org/downloads/)
- Jupyter Notebook Version 6.0.1+ (https://jupyter.org/install)
- Python packages: SWAT; Numpy; etc
- GIT version 2.21.0.windows.1+ (https://git-scm.com/downloads)
- SAS Studio 3.8 (https://support.sas.com/downloads/package.htm?pid=1924) or SAS Studio Version Fast 2020
- Viya 3.5 or Viya 2020
- Snowflake ODBC Driver (Optional: https://docs.snowflake.com/en/user-guide/odbc-windows.html)

## DATA MANAGEMENT

### STEP 1: CLOUD DATA AND GETTING STARTED

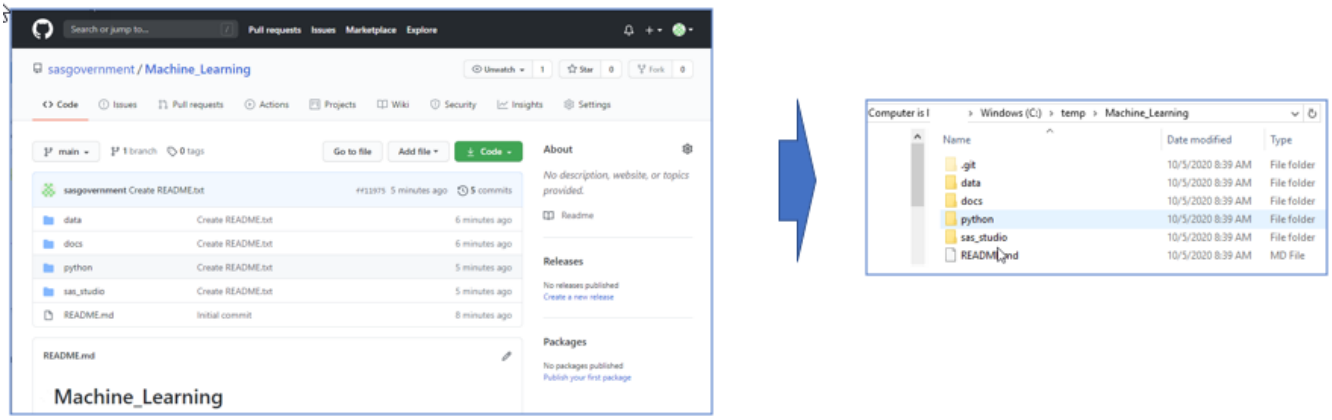
Data Management is the process of extracting, transforming, and loading the data. In our example, we are starting with a developed analytic dataset. We have already performed much of the initial analytic file creation including merging from the different data sources. We begin this process with the dataset as it exists in the link provided above.

## ACCESSING CODE AND DATA WITH GITHUB ~~(MICROSOFT)~~

You can use SAS to clone a Github repository, using the following command:

```
/*Use this SAS Code to clone a GITHUB Repo*/
data _null_;
 version = gitfn_version();
 put version=;
 rc = gitfn_clone("https://github.com/sasgovernment/Machine_Learning/",
   "C:\temp\Machine_Learning");
 put rc=;
run;
```

This will clone the github repo in your "C:\temp\" folder as shown in Screen 3.




Screen 3. Clone the Github repo.

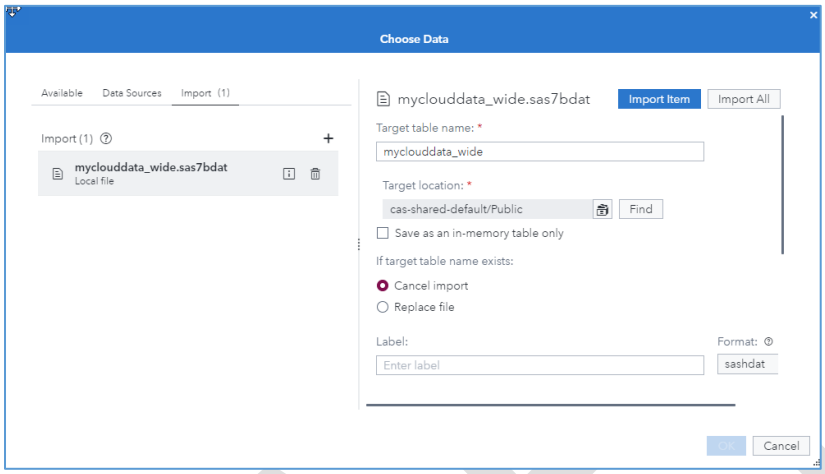
To use data in SAS Viya, you will need to load it to the in-memory engine or CAS (Cloud Analytical Services). There are several options to kick-start the project with data available in the Cloud and load data into CAS:

- Load the data from the "data" folder (it's been already downloaded from the Cloud)
- Use Python SWAT to run through an ETL process flow
- Use SAS Studio 3.8 and custom tasks in an ETL process flow
- Use SAS Studio 3.8 with Snowflake integration
- Use SAS Studio in Viya 2020

We will now explore each one.

## USING SAS VISUAL ANALYTICS TO LOAD DATA

In a new SAS Visual Analytics report, simply click on the Data icon on the far left pane. Select Import→Local Files and choose "myclouddata_wide.sas7bdat" from "C:\temp\Machine_Learning\data". Click on "Import Item" from Screen 4 to load the data into SAS Viya (CAS):

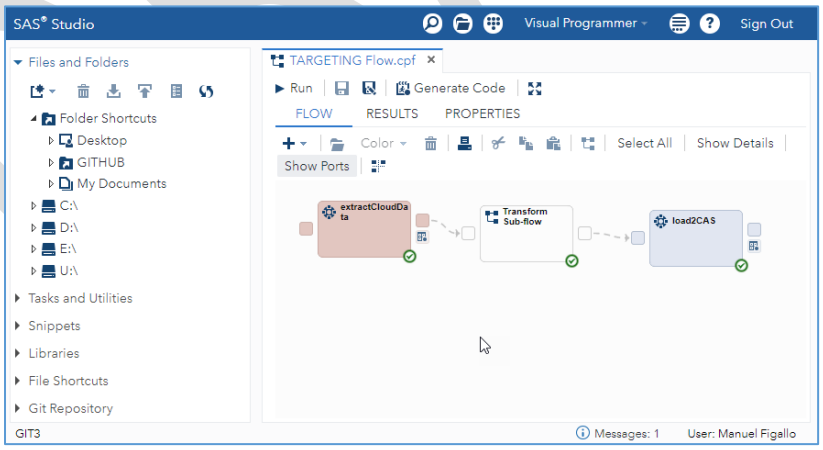



Screen 4. Loading CMS data into SAS Viya using SAS Visual Analytics.

## USING SAS STUDIO 3.8

The github repository contains two types of files in the "sas_studio" folder to create a process flow in SAS Studio:

- CPF – the entire process flow to bring in data into SAS Studio from the cloud.
- CTM – the blocks of code in a GUI to build the process flow from scratch.




Screen 7. Process flow using SAS Studio Custom Tasks in a data process flow.

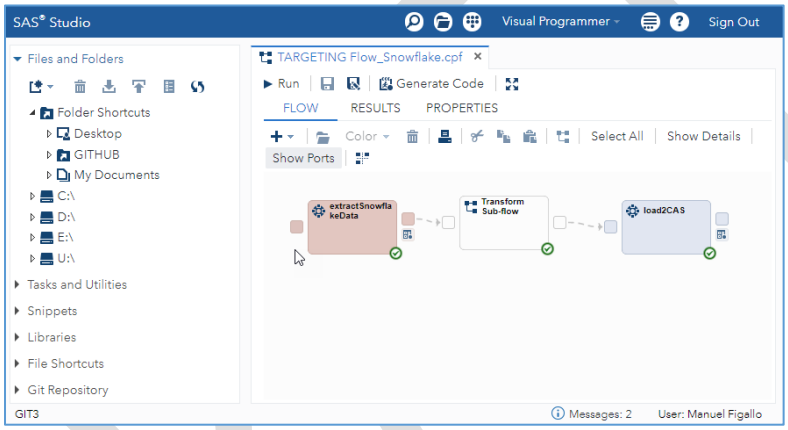
Using the files in "sas_studio" folder, you can create the process flow from scratch in SAS Studio or run a pre-made process flow. If you start from scratch, use the following code in a program node right after the transpose within a process sub-flow:

```
data work. MYCLOUDDATA_WIDE (drop=_NAME_);
        set work.MYCLOUDDATA_WIDE0;
        IF YEAR_NUM > 2016 THEN DELETE;
        YEAR_DT=mdy(1,1,year_num);
        format YEAR_DT mmddyy8.;
run;
```

Be sure to create a folder shortcut called “GITHUB” that points to “C:\temp\Machine_Learning\sas_studio”.

## USING SAS STUDIO 3.8 WITH SNOWFLAKE

A process flow can also be created utilizing the custom tasks (CTM files) for Snowflake integration as shown in Screen 8.

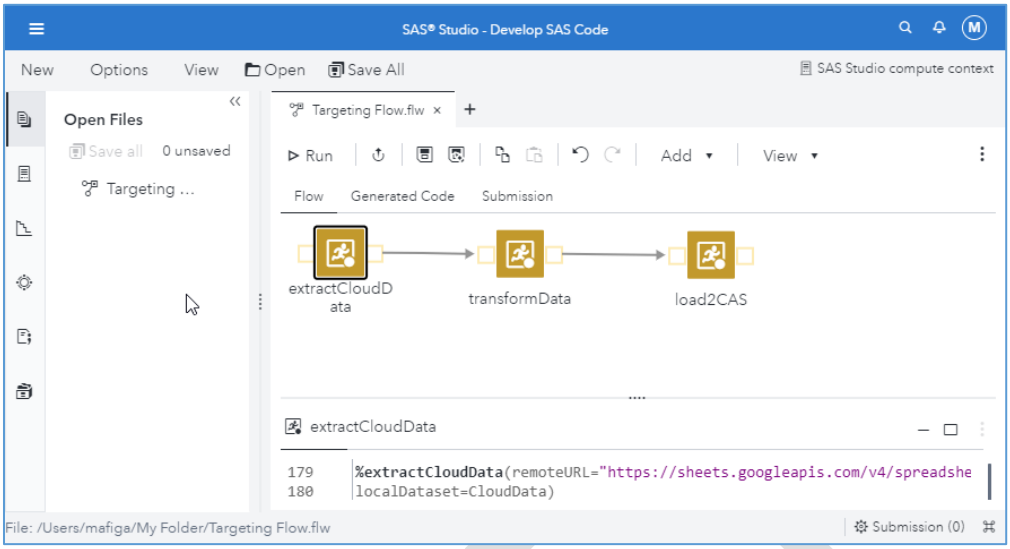



Screen 8. Process flow using SAS Studio Custom Tasks in a data process flow for Snowflake.

Note that in Screen 8, “extractCloudData” has been replace with “extractSnowflakeData”.

## USING SAS STUDIO ON VIYA 2020

SAS Studio on Viya 2020 allows you to also create “flows”. Simply go to the menu option “New→Flow”. From the “Flow” menu, select “Add→SAS Program”. Copy and paste the code from the “sas_studio/code” Github folder to produce the “Flow” as in Screen 9.




Screen 9. Process flow using SAS Studio Custom

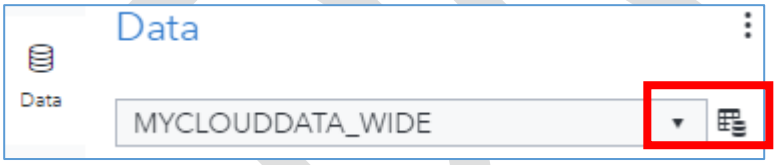
Click "Run" to produce output.

## SUMMARY OF DATA MANAGEMENT PHASE

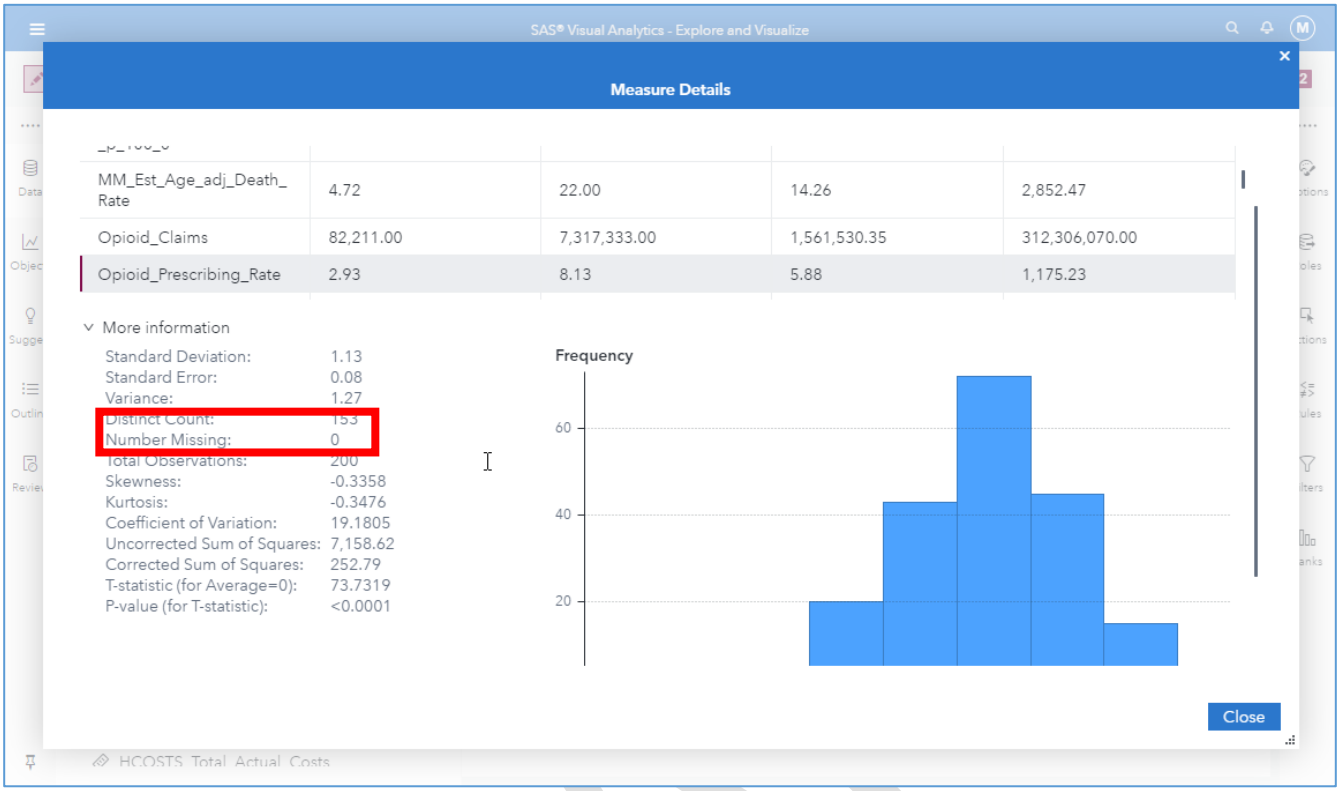
SAS Viya provides many alternatives to access data and load it into the CAS engine for fast access to analytics and dashboards.

Once the file has been loaded, we begin Data Management tasks including assessing variables to see if the data is complete and if variables to be used for analysis are of a normal distribution.

These features are available by clicking on the down arrow in the left pane:




We also know that some of the statistical tests we will be running assume a normal distribution of the variables. SAS Studio and Viya makes this process easy with the "Measure Details" after selecting the down-arrow as shown above. You will get something like this for each variable to assess the completeness of the variables:




Screen 10. Measure Details for "Opioid Prescribing Rate"

Notice the "Number Missing" for Opioid Prescribing Rate.

Rename "SDOH PMU" to "SDOH Physically and Mentally Unhealthy Days"

Rename "MM_Drug_poison_deaths_p_100_0" to "MM Drug Poisoning Death Per 100K".

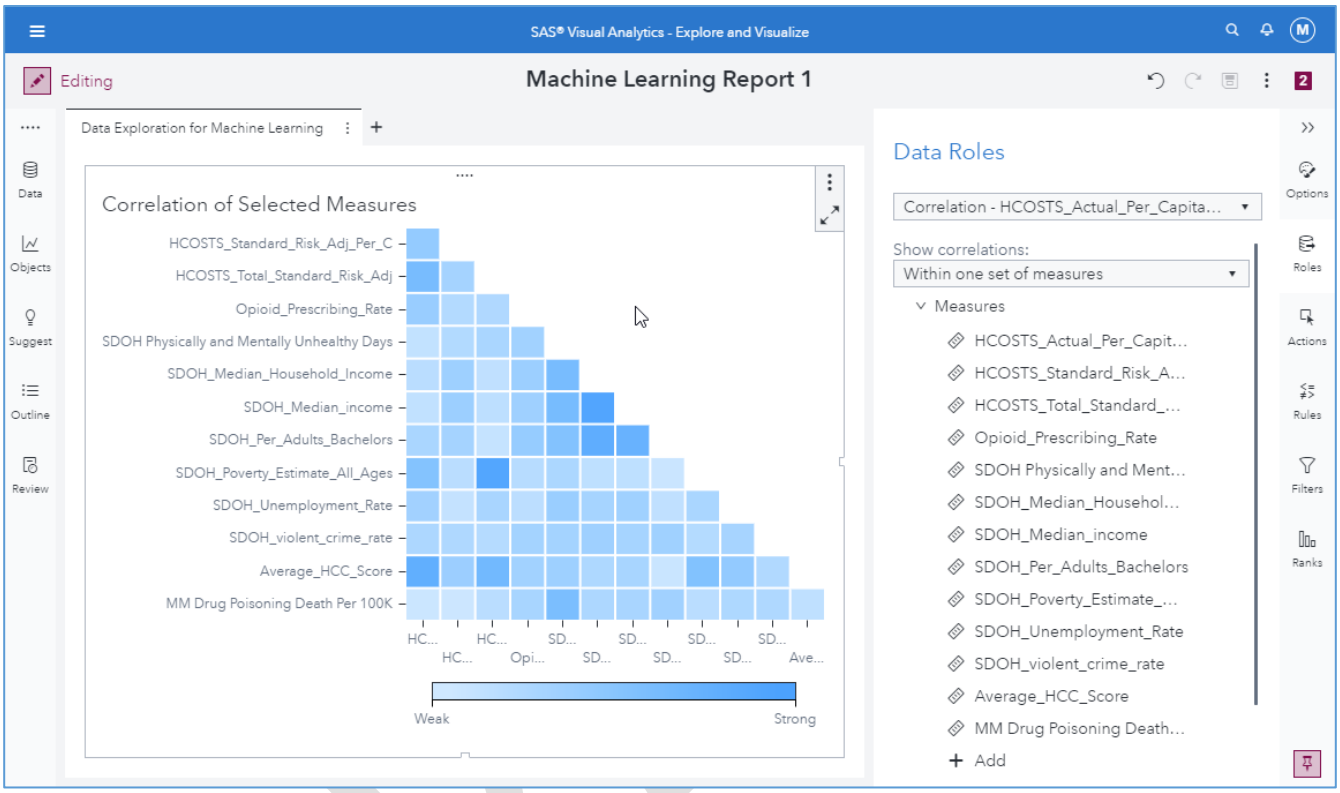
## DISCOVERY

### STEP 1: EXPLORATION WITH A CORRELATION MATRIX AND BUBBLE PLOT

You can explore your data in SAS Visual Analytics using a variety of visualizations.

These visualizations are available by clicking on the left page "Objects":


Drag and drop the correlation matrix to the canvas, selecting the varibles in Screen 11.

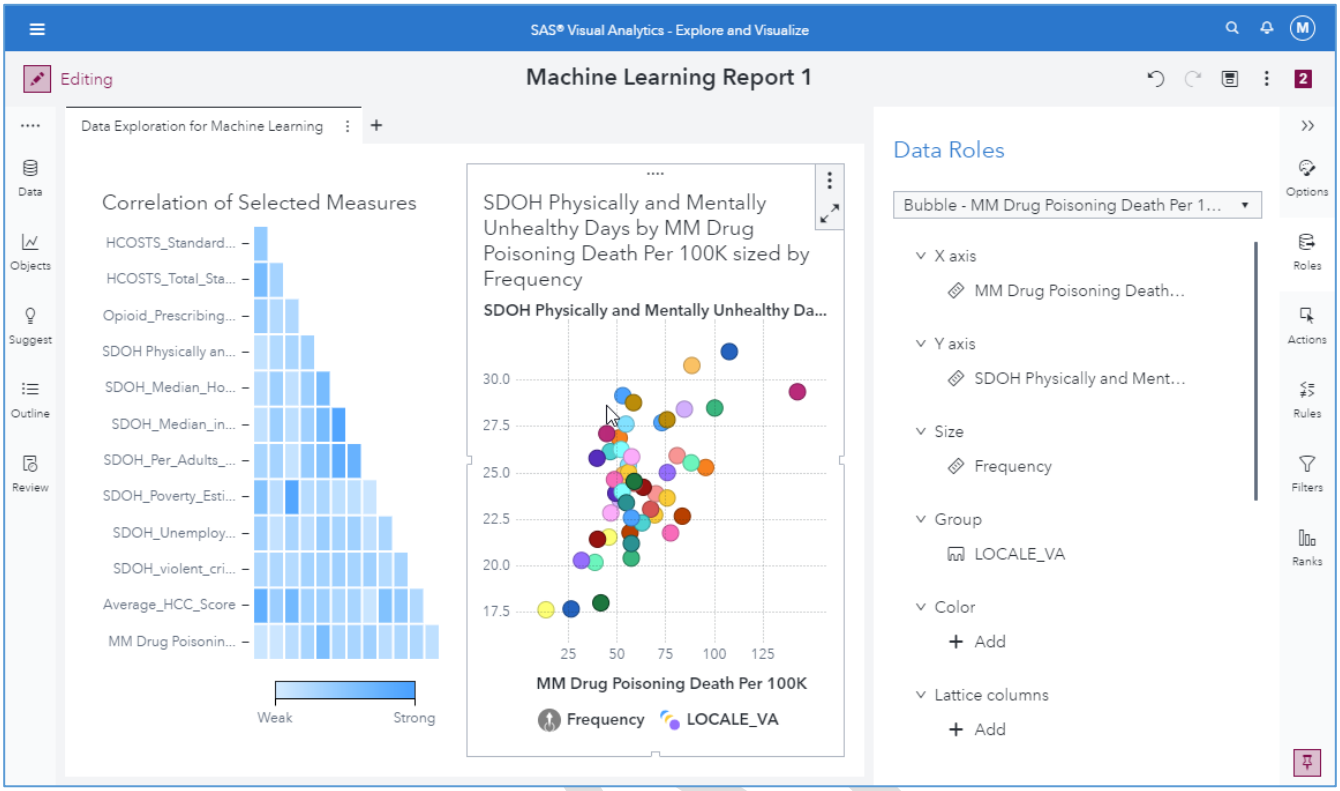



Screen 11. Correlation Matrix with Selected Variables. Notice the dark blue squares.

A correlation matrix displays the degree of correlation between multiple intersections of measures as a matrix of rectangular cells. Each cell in the matrix represents the intersection of two measures, and the color of the cell indicates the degree of correlation between those two measures.

Notice that the data variables are grouped into HCOSTS (Health Costs), SDOH (Social Determinants of Health), and MM (Mortaility and Morbidity).

Next to the correlation, add an the object “Bubble Plot” as in Screen X.



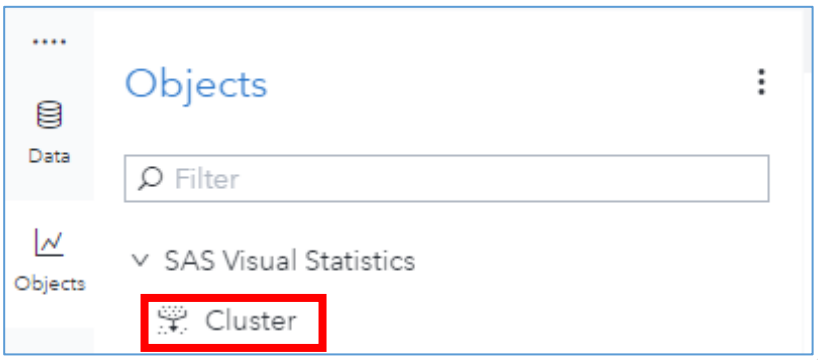

Screen 12. Bubble Plot.

A bubble plot is a variation of a scatter plot in which the markers are replaced with bubbles. A bubble plot displays the relationships among at least three measures. Two measures are represented by the plot axes, and the third measure is represented by the size of the bubbles. A bubble plot is useful for data sets with dozens to hundreds of values. You can add categories to the Grouping and Lattice roles

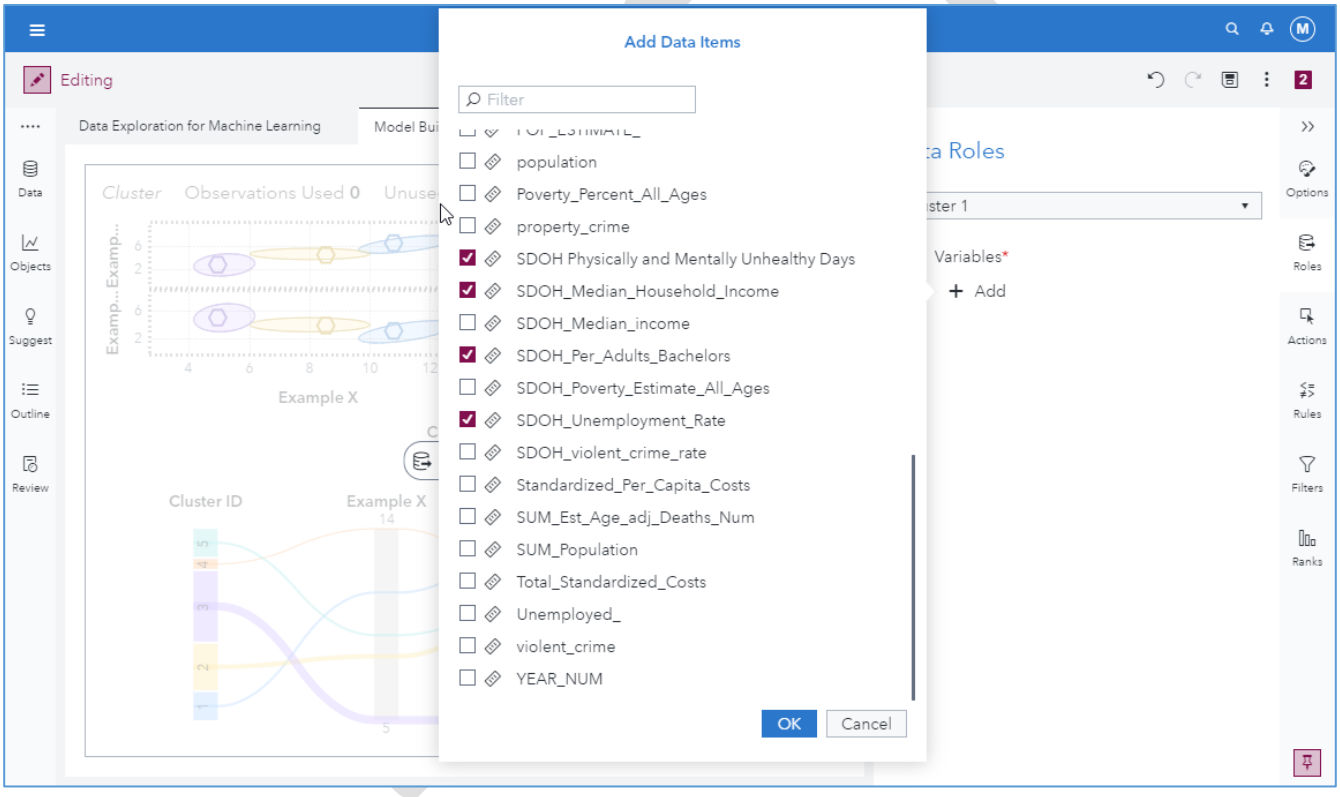
## STEP 2: MODEL BUILDING WITH AN UNSUPERVISED MACHINE LEARNING CLUSTER

Now that the data has been loaded, we can begin by exploring it. Let's begin with a K-Means cluster which is an unsupervised maching learning algorithm. Clustering is a method of data segmentation that puts observations into groups that are suggested by the data. The observations in each cluster tend to be similar in some measurable way, and observations in different clusters tend to be dissimilar. Observations are assigned to exactly one cluster. From the clustering analysis, you can generate a cluster ID variable to use in other explorations.

The cluster visualization is available from the left pane:




At this point, the graphical user interface should look like this:

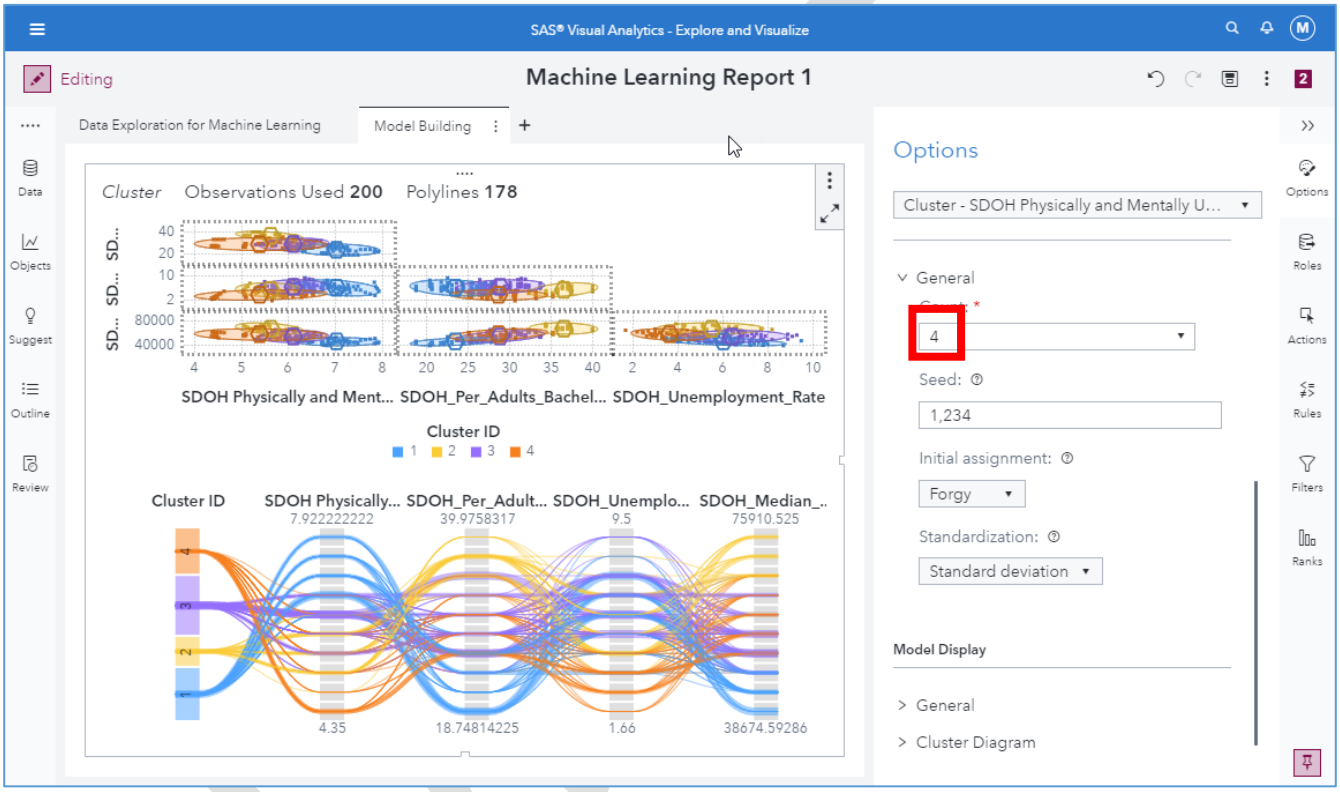



Screen 13. Feature selection for a  machine learning cluster.

The left pane contains the measure in the dataset.  The measures should be in this order:

SDOH Physically and Mental...
SDOH_Per_Adults_Bachelors
SDOH_Unemployment_Rate
SDOH_Median_Household_...

Then, drag and drop them into the middle pange or canvas. On the far right pane change the "Properties" to 4. This is shown here (as is the output from the cluster):




Screen 14. Modifying the number of clusters.

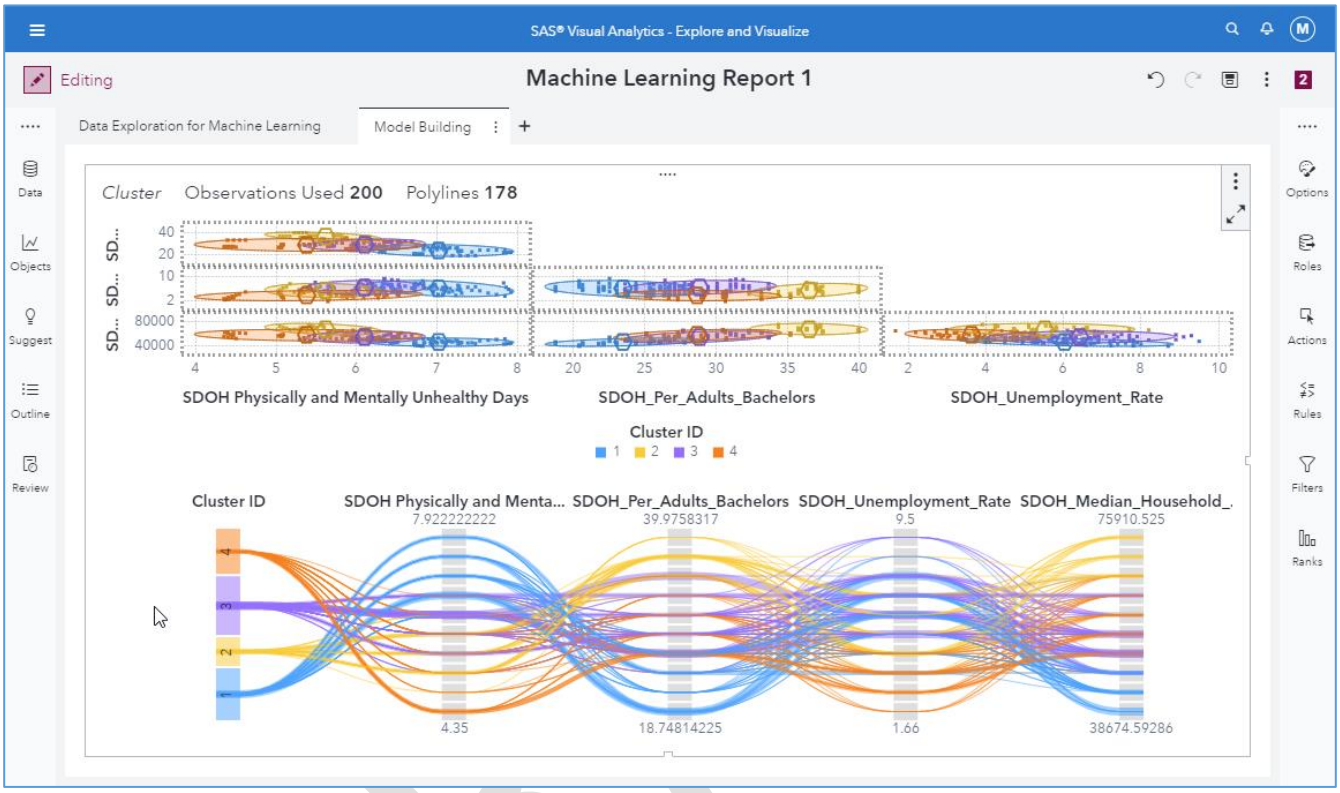
Notice that there are two section to the cluster: the cluster matrix, and the parallel coordinates plot. In the upper-right hand corner of the the parallel coordinates plot, click on the following icon:

The Cluster Matrix displays a two-dimensional projection of each cluster onto a specified number of effect pairs. These projections are useful for spotting cluster similarities and differences within the plotted effect pairs.

We will focus our attention on the parallel coordinates plot, however. The Parallel Coordinates plot shows patterns in the data and clusters. In this plot, the cluster ID is on the far left, and each variable is a column with its binned range of values

displayed vertically. Color-coded polylines are drawn from each cluster and show which range of values the cluster contains for every variable.

You should now see this:




Screen 15. K-means cluster for unsupervised machine learning.

Click on “Cluster ID” 1 in the far left (blue) to see this:


Screen 16. Risk segmentation focused on a"high-risk" cluster.

This represents counties with the following attributes:

- SDOH Physical_or_mental_unhealthy_LOG (HIGH)
- SDOH_Per_Adults_Bachelors (LOW)
- SDOH_Unemployment_Rate (HIGH)
- SDOH_Median_Household_Income… (LOW)

Next we will export the score code for the cluster and update our dataset with it.

### SUMMARY OF DISCOVERY PHASE

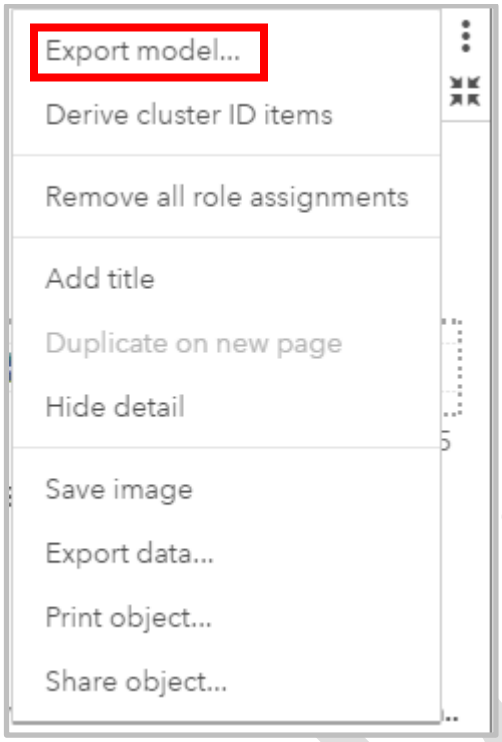
We can use SAS Viya to easily and quickly explore the data and crate machine learning models.

## DEPLOYMENT

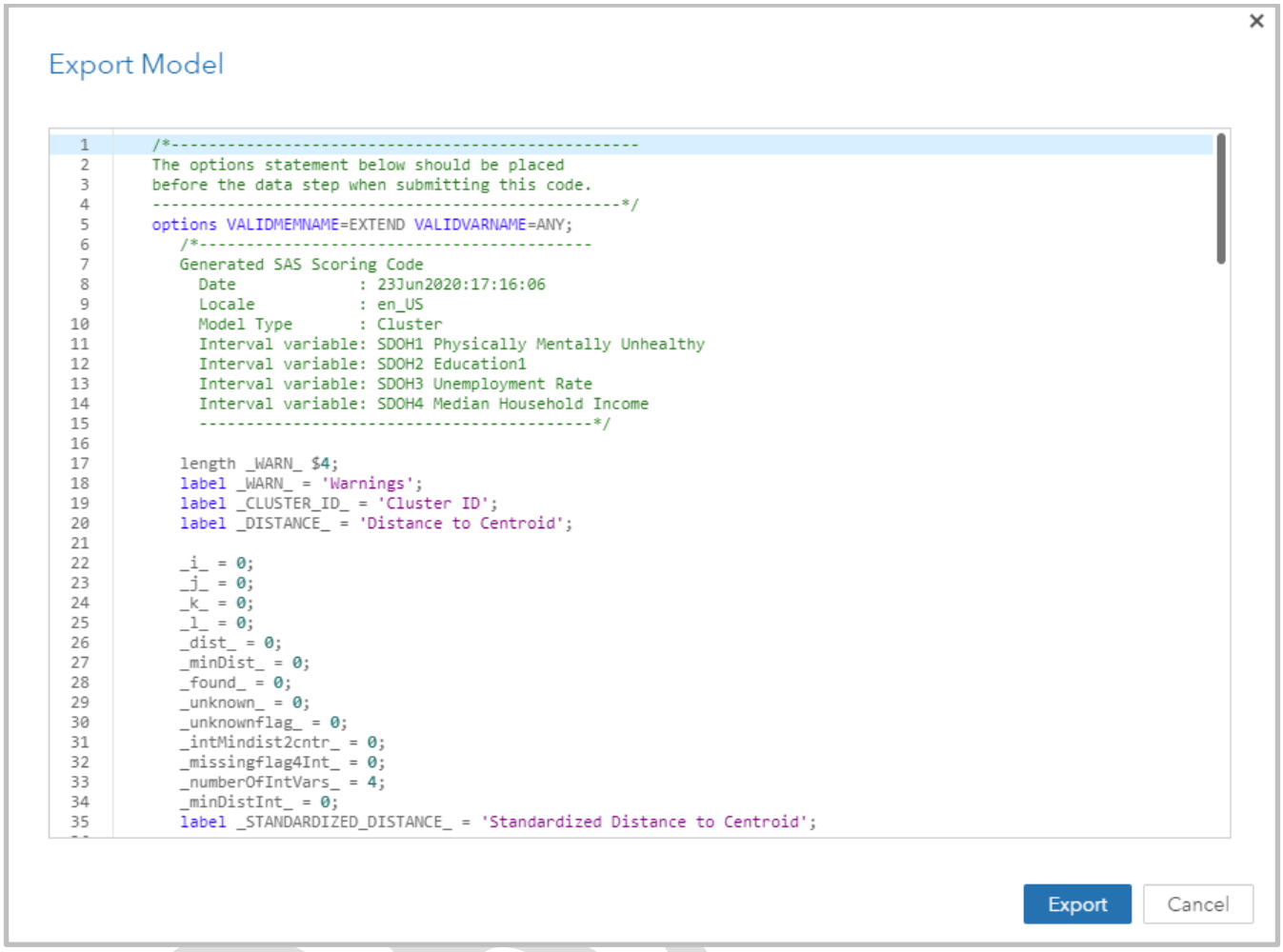
### STEP 1: DEPLOY AND SCORE BY EXPORTING CODE

To score a dataset you will first need to export the score code. From the top of the cluster visualization, select the down arrow.

You will see a list of possible actions:




Select “Export Score code..” and place the score code on the server.

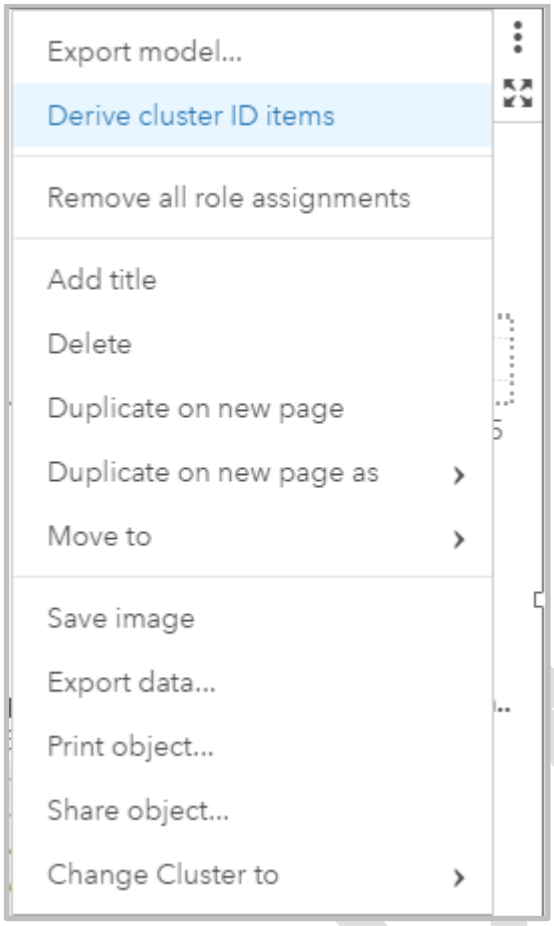



Next, you can use the “deployClusterModel” macro available in the “macros” folder in the GITHUB location:

https://github.com/sasgovernment

## STEP 2: DEPLOY AND SCORE BY DERIVING ID ITEMS

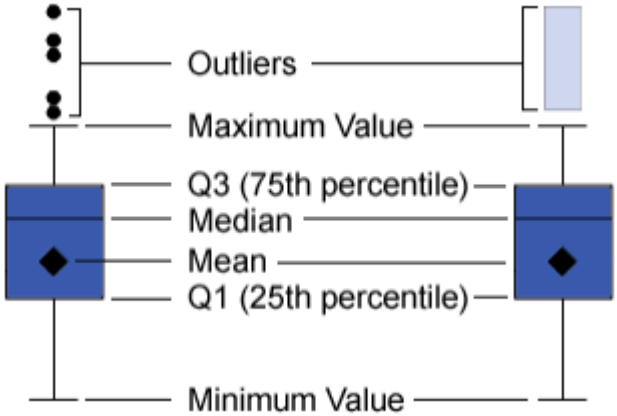
Another approach is to simply “Derive cluster ID items”:




## STEP 3: DEPLOY TO A VISUALIZATION OBJECT BY CREATING A BOX PLOT

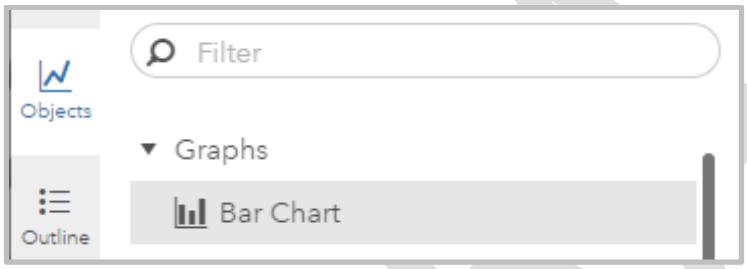
To understand how opioid_prescribing_rates may be distributed among our newly created clusters, we use a box plot.

A box plot displays the distribution of data values by using a rectangular box and lines called “whiskers.”

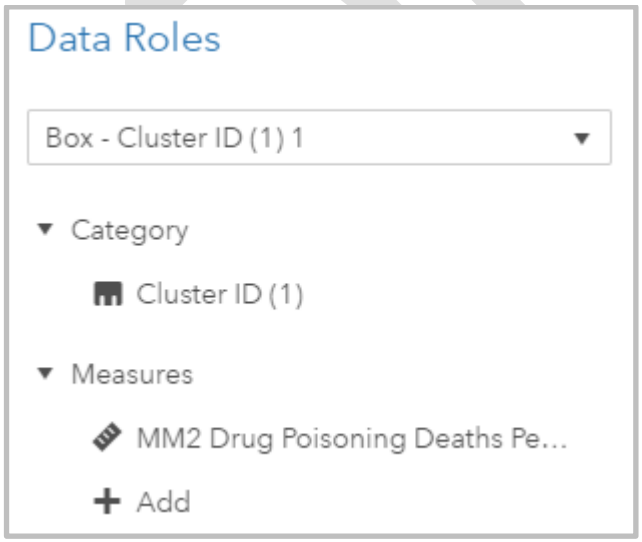



The bottom and top edges of the box indicate the interquartile range (IQR). That is, the range of values that are between the first and third quartiles (the 25th and 75$^{th}$ percentiles). The marker inside the box indicates the mean value. The line inside the box indicates the median value.

First, select Visualization→New. Then, select the following icon from the top:



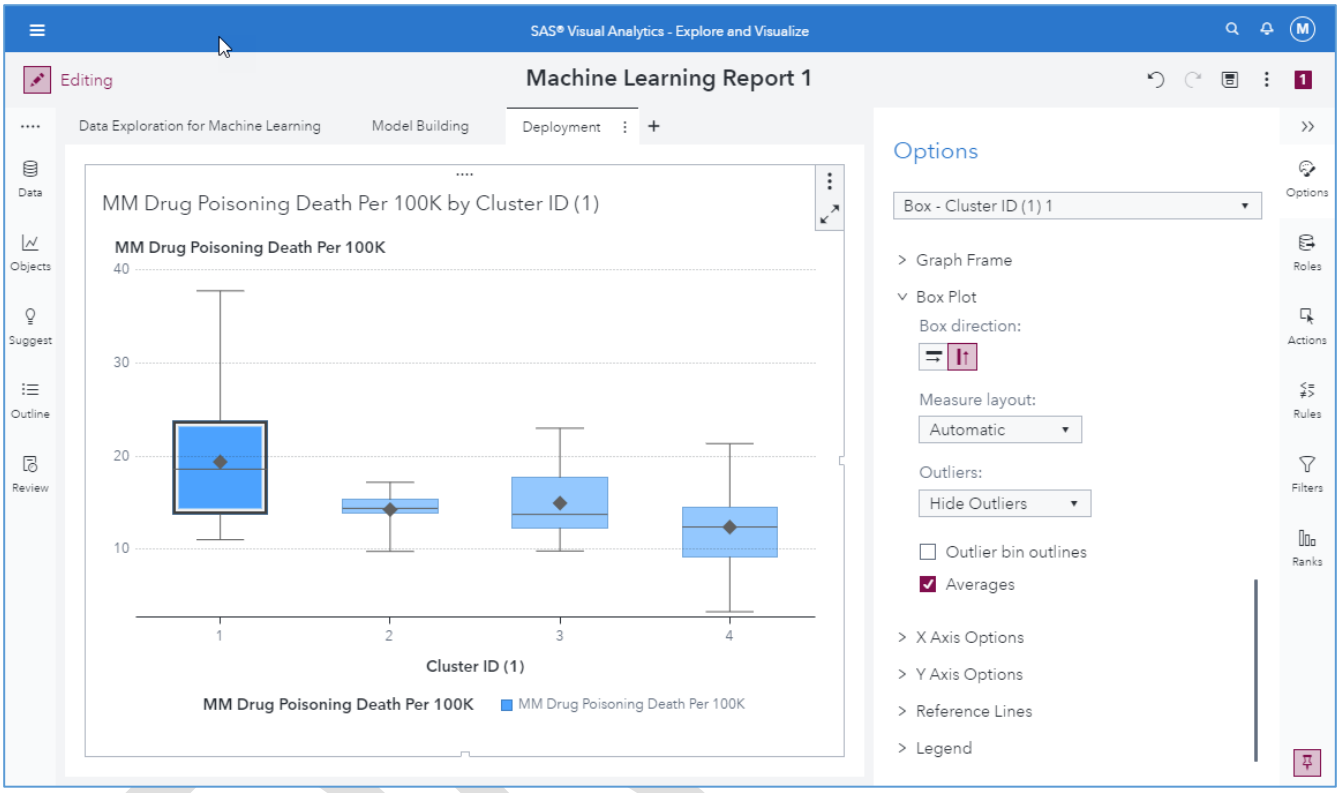

You can then produce the box plot by selecting the following variables in “Roles”:




(Note that you have to convert ClusterID to a “Category” type to use it in the box plot.)

Change the Properties to accommodate and show outliers and averages:

You will now have this visualization:

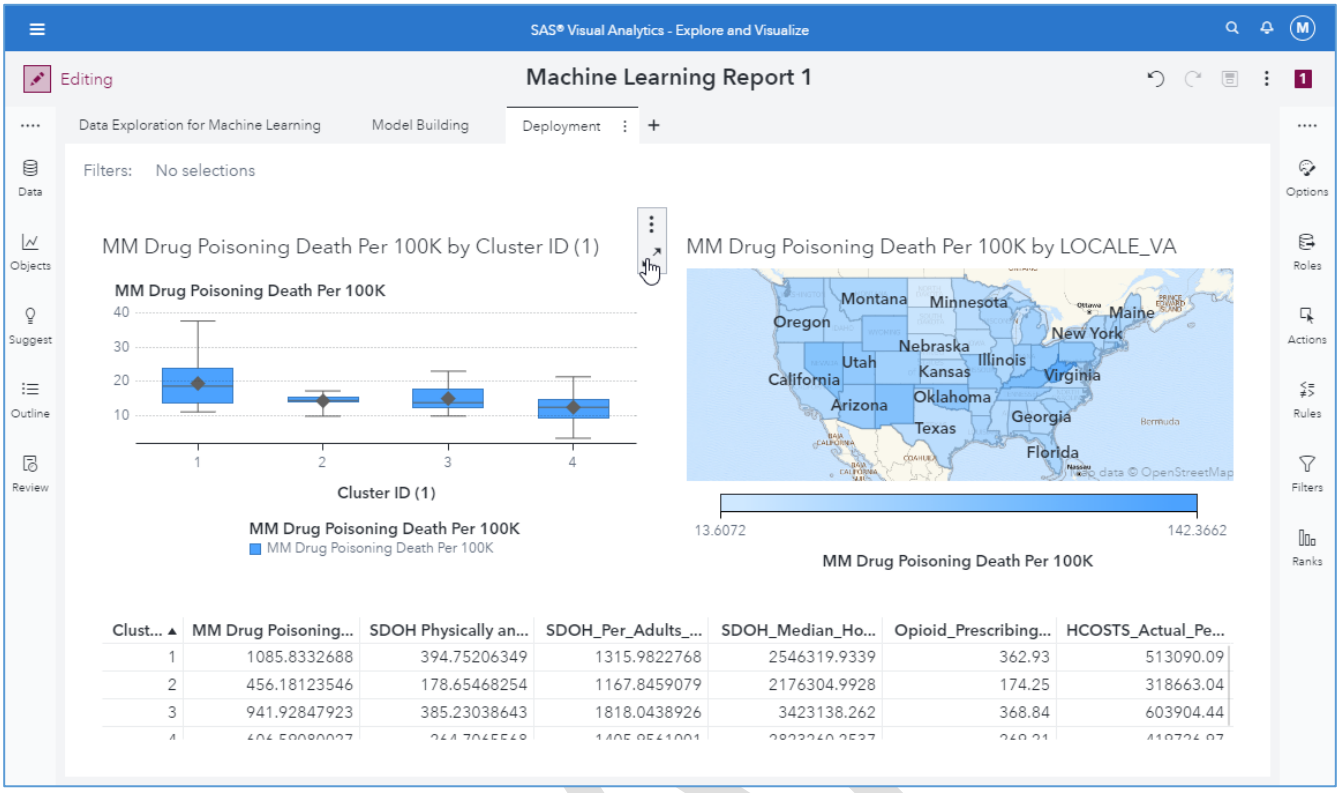



Screen 17. Box Plot showing the distribution of our "high-risk" cluster.

Interpretation of this finding could be that the Appalachian area does not have enough variance in socio-economic status or that the opioid epidemic in the Part D program does not differ between socio-economic groupings.

## STEP 4: DEPLOYING THE CLUSTER MODEL IN A REPORT OR DASHBOARD

A dataset scored with our machine learning model can also be deployed in a dashboard as shown in screen 18.




Screen 18. Box Dashboard with machine learning output.

The dashboard can also made interactive so that selecting one visual component will impact the rest. To complete this tutorial, try to complete the dashboard in Screen 18 on your own.

### SUMMARY OF DEPLOYMENT PHASE

After a machine learning model has been created, it can be deployed in a variety of ways, including as a dashboard to make the machine learning output more consumable.

## CONCLUSIONS

Using Jupyter Notebooks (Python) and Viya, users can enhance the analytic power of their data, explore new data sources, investigate them, and create visualizations to uncover relevant patterns. Users can then easily share those visualizations in reports. In traditional reporting, the resulting output is well-defined up-front. That is, you know what you are looking at and what you need to convey. However, data discovery invites you to explore and understand the data, its characteristics, and its relationships. Then, when useful visualizations are created, you can incorporate those visualizations into reports that are available on a mobile device or in the web browser.

## DEFINITIONS

**Correlation** - Correlation is a measure of association between two variables. The strength of the relationship is described as a value between -1 and 1. The closer the value is to -1 or 1, the stronger the relationship. The closer the value is to 0, the weaker the relationship. The colors in the correlation matrix shows the relationship in absolute terms, either weak (0)

or strong (1, -1). The actual value of the correlation appears in the tooltip and the results table. Double click or exploring a cell in the matrix will allow you to see a plot of the regression line.

**Linear Regression** - A linear fit line is the straight line that best represents the relationship between two variables. If the points on the scatter plot are tightly clustered around the line, then it likely provides a good approximation for the relationship. If not, another fit line should be considered to represent the relationship. If outliers (points which are distant from the rest of data) are present, they can have a strong influence on the slope of the line, and those points should be examined more closely.

**Bubble Plots** - A bubble plot displays the values of at least three measures by using differently sized plot markers (bubbles) in a scatter plot. The values of two measures are represented by the position on the plot axies, and the value of the third measure is represented by the marker size. You can create animated bubble plots to display changing data over time.

**Box Plots** - A box plot displays the distribution of data values by using a rectangular box and lines called “whiskers.”