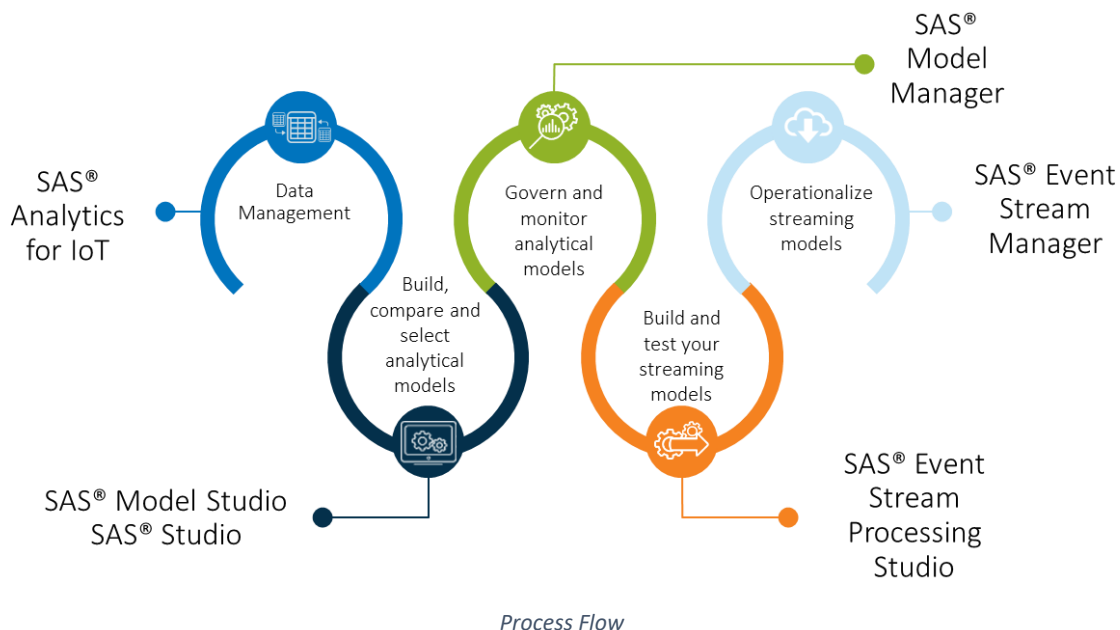


Deploy ASTORE from SAS Model Manager in SAS Event Stream Processing

Overview

The IoT analytical life cycle expands traditional analysis processing beyond investigation of stored data (i.e., at rest) to analytical investigation of streaming data (i.e., in motion) or at the edge. The combined capability of data management, streaming analytics execution, and intelligent decisioning enables fast and confident decision making from data, to discovery, to deployment.

SAS Analytics for IoT offers an optimized IoT solution ecosystem and addresses the entire analytical life cycle.



In this document we are focusing on the integrated process of model management and analytics deployment.

Key take-aways:

- Learn how to import ASTORE from models built in SAS Model Studio, SAS Studio, or Jupyter Notebook into SAS Model Manager
- Learn how to deploy ASTORE from SAS Model Manager in SAS Event Stream Processing Studio

SAS Model Manager (MM) is a modern, open, and flexible model management framework offering common services to inventory, governance, publish, monitor, and improve analytical models and operationalize analytics. It acts as a common model repository for models built in SAS Model Studio using VDMML (Visual Data Mining and Machine Learning) or VTA (Visual Text Analytics), or advanced analytical models in SAS Studio or python. It also supports changing nature of data, models, and alerts on it. You can publish and deploy models in a variety of ways, including : in SAS, databases, Hadoop,

streaming analytics, and devices; as well as through web services and APIs. You can monitor model performance overtime, retrain the models, and integrate within SAS Event Stream Processing Studio (ESP Studio).

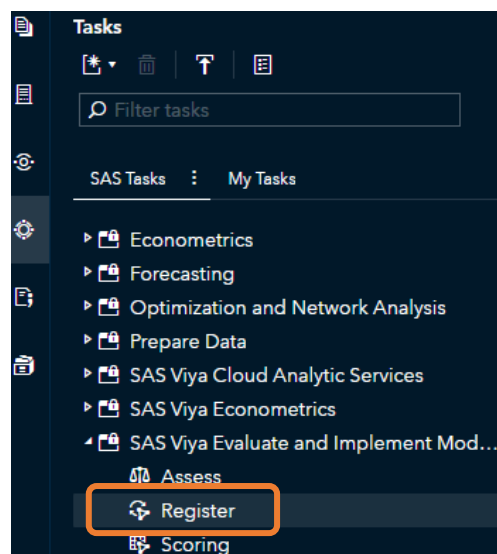
SAS Event Stream Processing Studio (ESP Studio) is a web-based client that enables you to create, edit, upload, and test event stream processing projects. The projects can be designed to perform tasks, such as calculating summary statistics on an interval of streaming data or filtering data before storage. It also enables the application of advanced analytics and machine learning techniques to streaming data. ESP also supports multiphase analytics. You can use offline models such as Support vector data description (SVDD) , Robust principal component analysis (RPCA) by pulling champion models from SAS Model Manager (MM), or you can build in-stream models like Subspace tracking (SST), Short-time Fourier transform (STFT) to design and test models on streaming data. It follows the concept of build once and deploy anywhere and publishing a project version from SAS Event Stream Processing Studio makes the project version available to SAS Event Stream Manager which can deploy the model to one or more devices as needed.

1. Importing model from SAS Studio to SAS Model Manager

The first example shows how to create and save an analytic store for a model using the SAS Studio interface. The SAVESTATE statement creates an analytic store for the model and saves it as a binary object in a data table. You can use an analytic store in the [ASTORE](#) procedure to score new data.

```
/* Phase 1: Train SVDD model using normal operations data from 18 engines */  
proc svdd data=mycas.turbofan_training ;  
  id engine_id Datetime;  
  input &varlist/level=interval;  
  kernel rbf / bw=trace(nrep=4);  
  code_file='/mnt/viya-share/homes/svdd_ds_scr.sas';  
  savestate rstore=mycas.turbofan_svdd;  
run;
```

Using Register task under SAS Viya Evaluate and Implement models, you can import a scoring model (ASTORE, an analytic object in a CAS table) from SAS Studio to SAS Model Manager (MM).



Below are steps to register an analytic store to MM. See [Register](#) for more details.

- Select the CAS table that contains the scoring model.
- Enter a name and description of your model.
- Under the Model Manager heading, enter the name of the project that you created in SAS Model Manager.
- Specify the location of the project in SAS Model Manager
- To run the task, click **Run**.

Import Model Information

▼ Model

Specify a CAS table that contains an analytic store.

MYCAS.TURBOFAN_SVDD

Model name: *

turbofan_svdd

Model description:

SVDD

▼ SAS Model Manager

▼ Project

Project name: *

TurboFan_SVDD

☐ New project version

Project location:

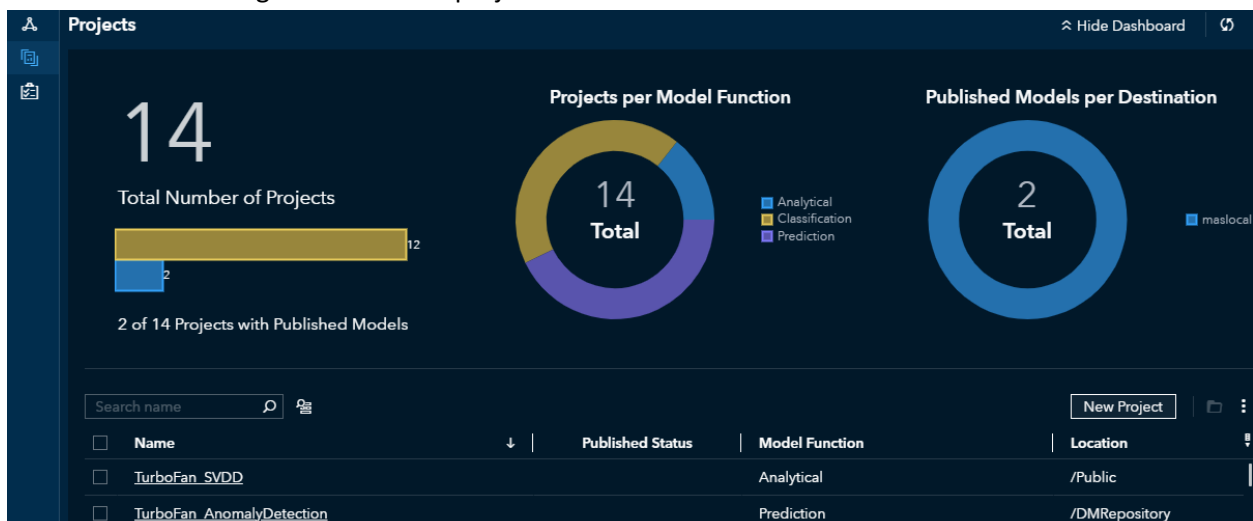
Public

☐ Add JSON files for model comparison

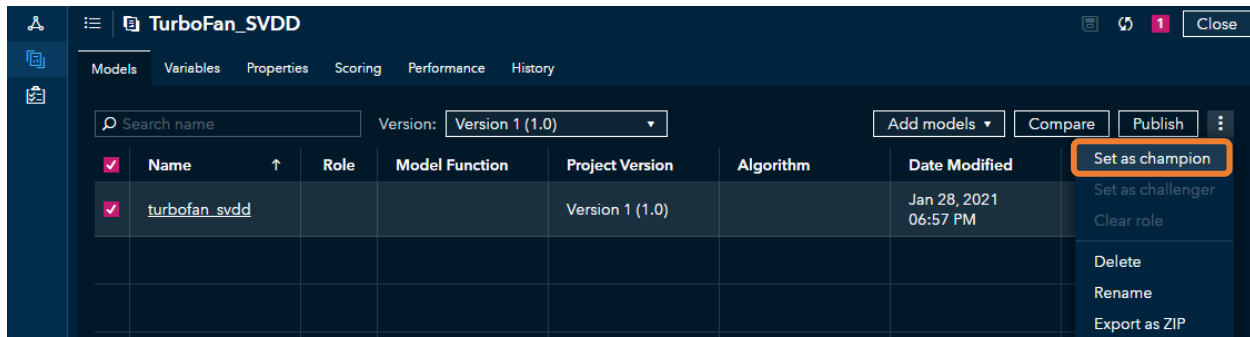
Note: A prerequisite for this task is an existing SAS Model Manager project. If no project exists, you must create one in SAS Model Manager.

*Tip: To avoid overwriting an existing SAS Model Manager project with the same name, select **New project version**. A new project is created with an incremental version number.*

- Open the Viya menu and select “Manage Models”. It opens SAS Model Manager.
- Model Manager shows list of projects.



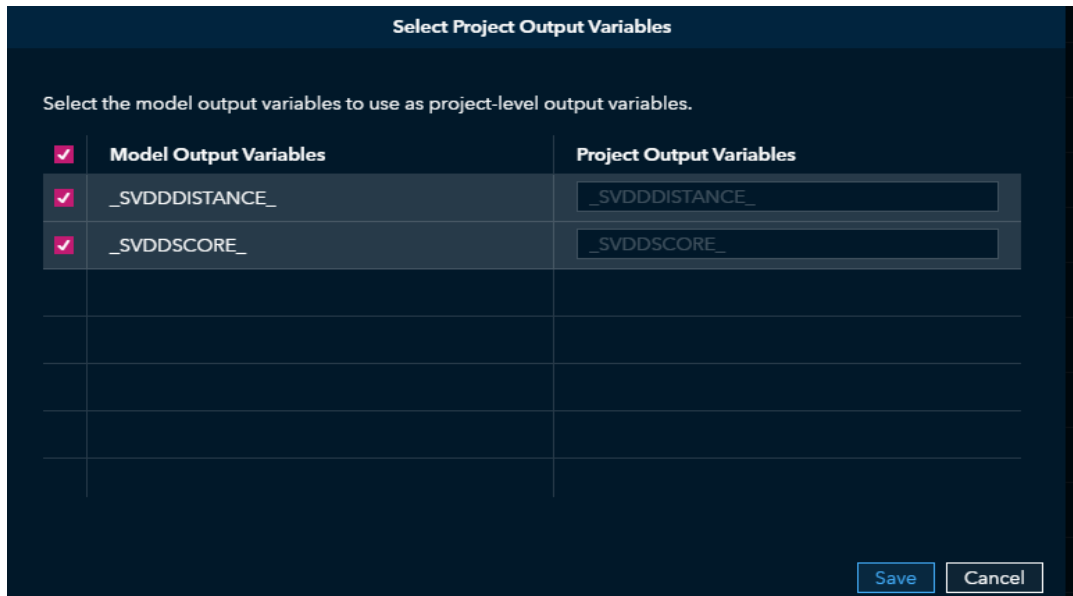
- Open the project used in Register task.



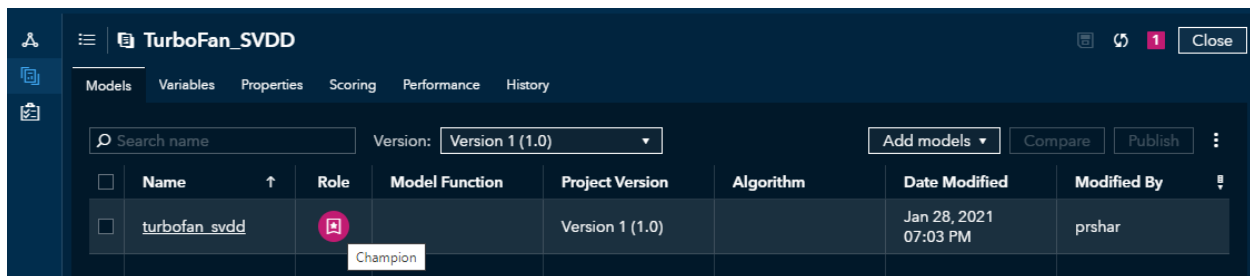
- Check box for the model name and click the ellipse → Select 'Set as champion'

Tip: Add model output variables to project output variables (if not added already present) to use them in ESP. Select Variables tab → Select Add Variables → Custom Variables.

- A window Select Project Output Variables pops up → Select model output variable(s) to map → Click 'Save'

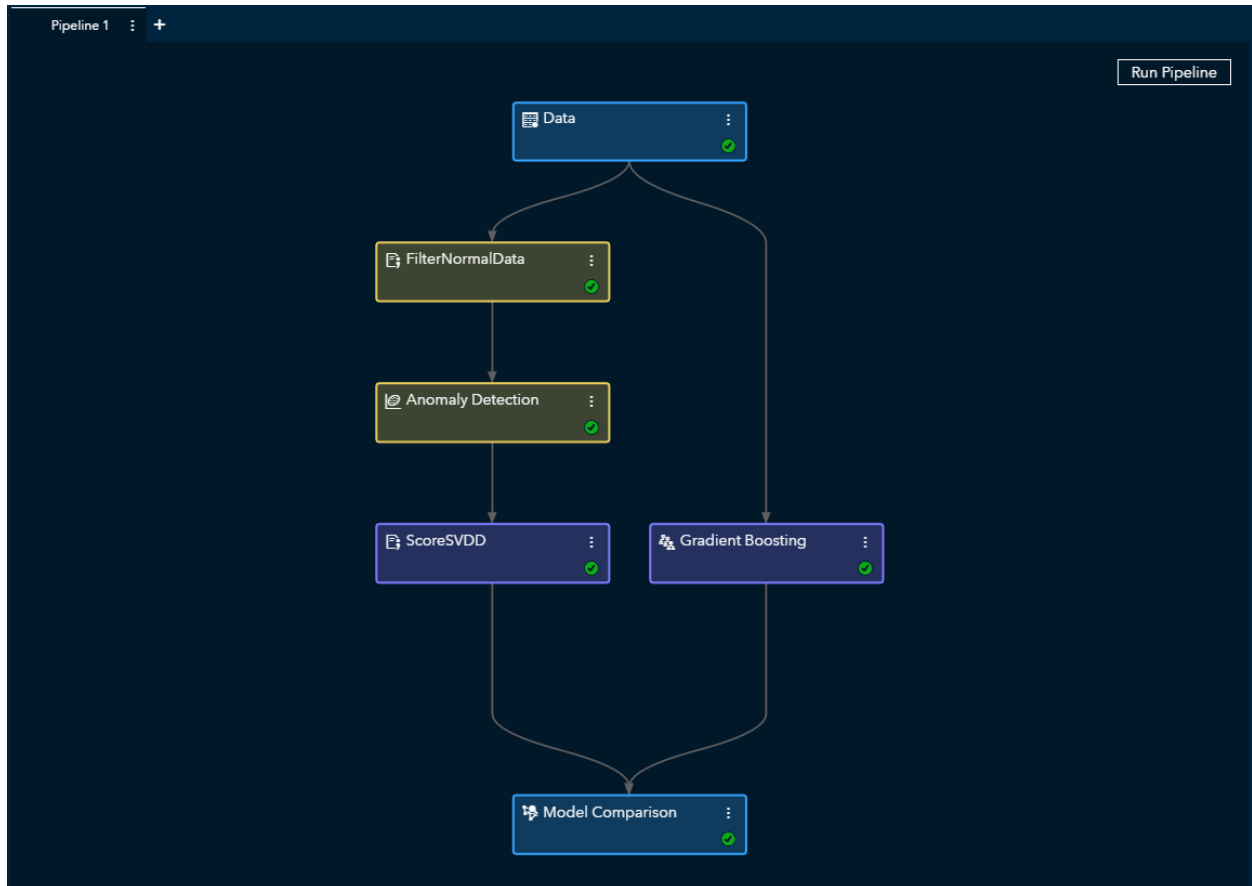



- Role for the model is set to Champion



2. Importing model from SAS Model Studio to SAS Model Manager

This example walks through the process of registering a model in MM from SAS Model Studio. The figure below shows a model pipeline using SAS Visual Data Mining and Machine Learning in SAS Model Studio.

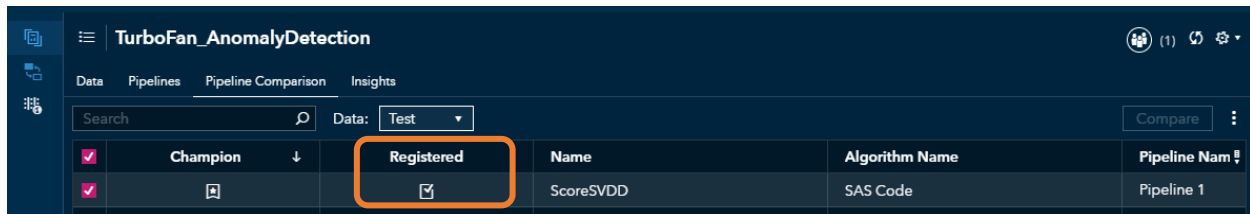


- Select Pipeline Comparison tab to view champion model
- To register the model, click the ellipse  in the upper right corner and select **Register models**.

Champion	Name	Algorithm Name	Pipeline Name	Average Squared ...
<input checked="" type="checkbox"/>	ScoreSVDD	SAS Code	Pipeline 1	24.688
<input type="checkbox"/>				
<input type="checkbox"/>				
<input type="checkbox"/>				
<input type="checkbox"/>				
<input type="checkbox"/>				
<input type="checkbox"/>				
<input type="checkbox"/>				

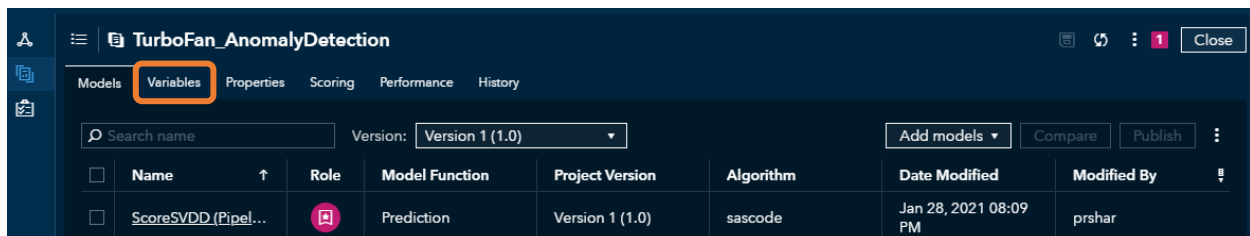
- Set as champion
- Remove challenger models
- Register models**
- Publish models
- Score holdout data
- Download score API
- Download score code
- Manage Models

- The registered status is added.



TurboFan_AnomalyDetection						
Data Pipelines Pipeline Comparison Insights						
Search Data: Test Compare						
✓	Champion ↓	Registered	Name	Algorithm Name	Pipeline Name	
✓		✓	ScoreSVDD	SAS Code	Pipeline 1	

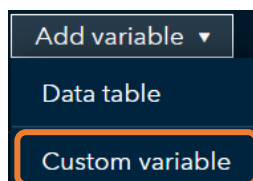
- Open SAS Model Manager to view the project and champion model. The Role is set to Champion and is ready to be used in ESP.



TurboFan_AnomalyDetection							
Models Variables Properties Scoring Performance History							
Search name Version: Version 1 (1.0) Add models Compare Publish							
□	Name ↑	Role	Model Function	Project Version	Algorithm	Date Modified	Modified By
□	ScoreSVDD (Pipe...		Prediction	Version 1 (1.0)	sascode	Jan 28, 2021 08:09 PM	prshar

Tip: Add model output variables to project output variables (if not already present) to use them in ESP.

a.) Select Variables tab → Select Add Variable → Custom Variable. See [link](#) for details.



b.) Select the model → Click ellipse → Select 'Set as Champion'.

3. Importing model from Jupyter notebook to SAS Model Manager

The next example shows how to create an ASTORE from a SVDD model built in Jupyter Notebook.

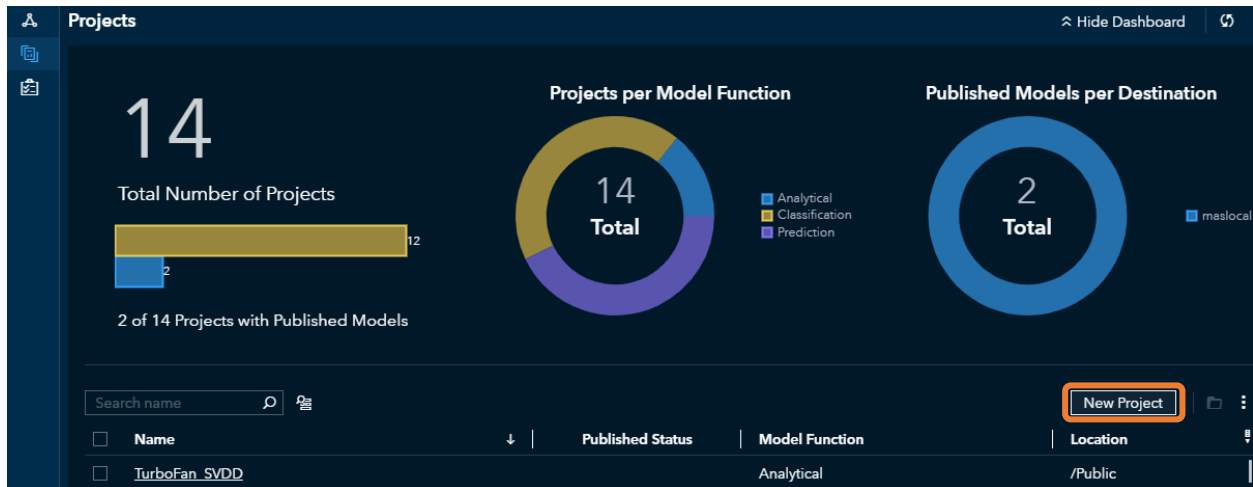
- Save model ASTORE. See this [link](#) for details.

```
# Phase 1: Model Training
# Run svDataDescription.svddTrain action set on Training data
sess.svDataDescription.svddTrain(bw=94,
                                  solver='actset',
                                  inputs=[{"name": "SUPPL_FAN_SP"}, {"name": "DIS_AIR_TEMP"}, {"name": "DUCT_PRESS_ACTV"}, {"name": "MIXED_AIR_TEMP"}, {"name": "RTRN_AIR_TEMP"}, {"name": "MAX_CO2_VAL"}, {"name": "CHW_VALVE_POSIT"}],
                                  id=["AHU"],
                                  savestate = {"name": "svdd_ahu", "replace": True},
                                  output={"casout": {"name": "sv", "replace": True},
                                  table={"caslib": "casuser", "name": "ahu_train"}
                                  )
```

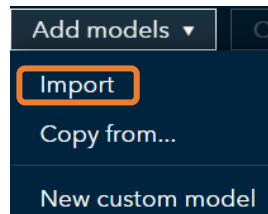
- The ASTORE is saved to the path specified below.

```
store=sess.download(rstore='svdd_ahu')
with open('/user/my_code/ahu_svdd.astore', 'wb') as file:
    file.write(store['blob'])
```

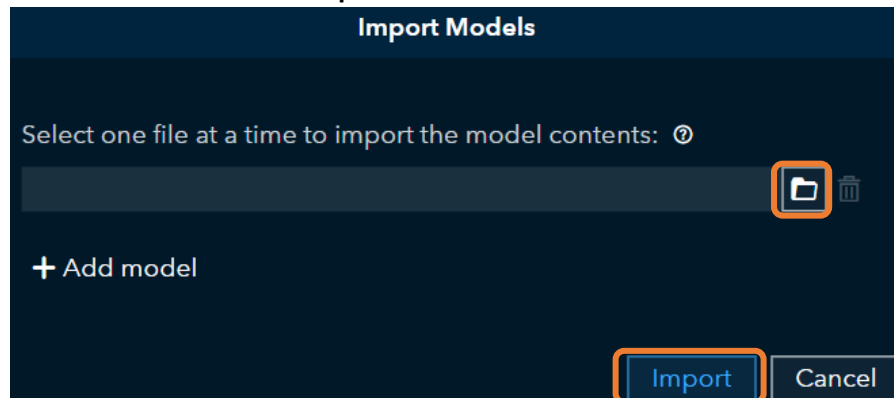
- Open SAS Model Manager
- Click **New Project** in the right. See [steps](#) to create a new project.



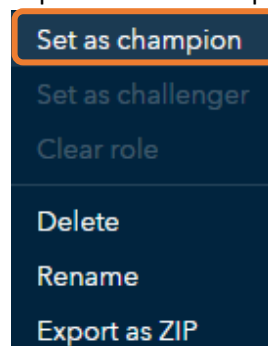
- Select Add Models → Import



- Browse to ASTORE location. Click **Import**.



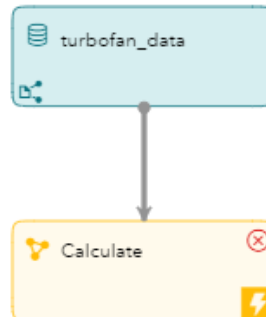
- Check Model and Project variables. Add new variables as needed.
- Select the model and click 'Set as Champion' from the ellipse on right upper corner.



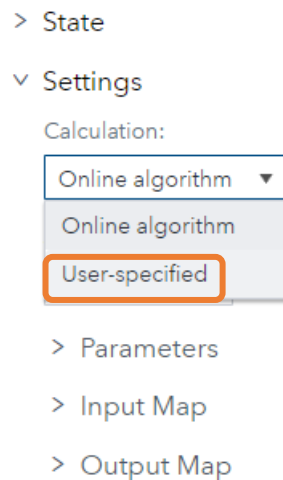
4. Deploying ASTORE from SAS Model Manager in SAS Event Stream Processing

In this section, an ASTORE is imported from MM and is used to build and test streaming model for real time analytics in SAS Event Stream Processing Studio (ESP). A *project* specifies how an engine analyzes and transforms input event streams into meaningful results.

- Open SAS Event Stream Processing Studio
- Add a Source window and configure the input stream
- Add a Calculate window under Analytics to workspace and connect it to Source window




- Go to Settings and select User-Specified under Calculation.



- Expand the Handlers→ Click Edit input handler

▼ Handlers

		
Source	Edge Role	Handler
turbofan_data	Data	(undefined)

- Select 'Import a module from SAS Model Manager' under Handler type

Input Handler ✕

Handler type: *

SAS Micro Analytic Service ▼

SAS Micro Analytic Service

Import a module from SAS Model Manager


Import a module from a SAS Model Manager ZIP file

Function:

score ▼

OK

Cancel

- Click the search icon  . It shows the models in MM repository and all available champion models.

Import from SAS Model Manager

🔍

☒ Search champion models only

Name	Type	Scor...	Cha...	V...	Last Updat...	Tags
Repository	Repository		✕			
Test	Project	ds2M...	✓		2021-01-29T0...	
TurboFan_AnomalyDetection	Project	ds2M...	✓		2021-01-29T0...	
TurboFan_AnomalyDetection	Project version	ds2M...	✓	1.0	2021-01-29T0...	
ScoreSVDD (Pipeline 1)	Model	ds2M...	✓	2	2021-01-29T0...	

- Select the Champion model for the project → Click OK
- Check the box for 'Add the output schema to the window' → Click OK.

Input Handler ✕

Handler type: *

Import a module from SAS Model Manager ▼

[View model properties](#)

[Model input schema](#)

[Model output schema](#)

☒ Add the output schema to the window

OK

Cancel

- Click Output schema on the right pane and edit rows

Calculate

Calculate window

At least one key field is required.

Key	Type	Field	Mapping
	⊕	_SVDDDISTANCE_	(unmapped)
	⊕	_SVDDSCORE_	(unmapped)

- Click Import Schema and add Key field and other fields as needed → Click OK

Output Schema

Key	Field Name	Type
<input checked="" type="checkbox"/>	Event_ID	string
<input type="checkbox"/>	Engine_ID	string
<input type="checkbox"/>	DATETIME	string
<input type="checkbox"/>	_SVDDDISTANCE_	double
<input type="checkbox"/>	_SVDDSCORE_	double

- Save the project and select **Enter Test Mode** → Click **Run Test**

Enter Test Mode

ESP server:

- Click OK when prompted to load and start project in Cluster
- Review the results

turbofan_data
Score_SVDD
Filter_Alerts

Continuous query: cq1
Currently retained events: 39933
☒ Show formatted fields

Drag a column heading here to group by that column

Opcode	Event_ID	Engine_ID	DATETIME	_SVDDDISTANCE_	_SVDDSCORE_
Insert	c64207cb-f5a2-42cf-a...	engine9	09JAN2016:04:30:00...	0.887765	-1.000000
Insert	d53bd5eb-cfb8-4242...	engine9	08JAN2016:22:00:00...	0.892006	-1.000000
Insert	1cbe0621-dd95-45a0...	engine9	08JAN2016:15:30:00...	0.890414	-1.000000
Insert	77463cbf-6fc4-4a52-...	engine9	08JAN2016:09:00:00...	0.893284	-1.000000

Summary

As this document illustrates, SAS provides toolkits to harness value from IoT data and fill the gaps we see in enterprise wide deployment management. SAS Analytics for IoT provides a streamlined framework for the entire IoT analytical lifecycle and enables business users to easily select sensor data, build models leveraging multiphase analytics approach, govern and monitor models, and design and deploy models to devices that scales from edge to cloud. All of this allows you to harvest new opportunities as you progress on your digital transformation journey.