

SAS<sup>®</sup> Viya<sup>®</sup> Trial

# Deploy Insights Guide

MLOps Engineering Tasks



# Intro

# Data and AI Life Cycle: Deploy Insights

A recent study by The Futurum Group showed that SAS Viya increases data and AI team productivity by 4.6x.

The analysts compared SAS Viya to alternatives in an end-to-end customer churn prediction analysis, a common use case relevant to many industries.

The final step in the data and AI life cycle is **Deploy Insights**. This was performed by an **MLOps Engineer** persona, who was tasked with deploying the model and maintaining its performance over time.

This guide will walk you through the steps an MLOps Engineer took to complete the Deploy Insights portion of the life cycle in SAS Viya.

NOTE: The Develop Models steps must have been done before completing this portion of the life cycle. Visit [this link](#) to find the Develop Models Guide if needed.

# MLOps Engineer

Deploy & Maintain Models, Create  
Monitoring Reports

# Tasks

1. Central Model Repository
2. Automatic Generation of Deployment Files
3. Model Comparison Metrics and Graphs
4. Model Scoring – Testing for Deployment
5. One-Click Deployment – Multiple Deployment Destinations
6. Model Performance Reports
7. Retraining
8. Versioning
9. Alerting
10. Scheduling in Production

# Resources

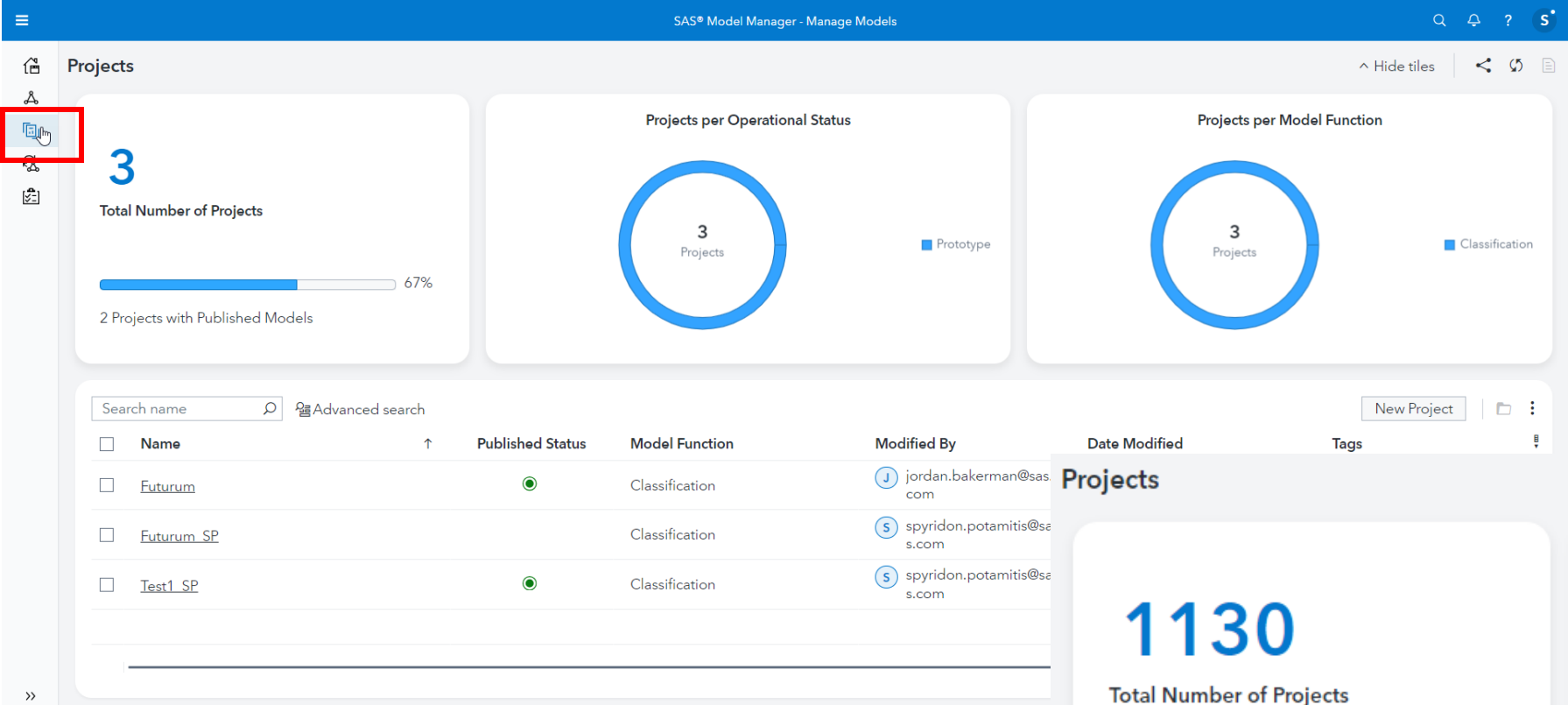
Watch before start

- [Quick Start - Data & AI Life Cycle](#)
- [Quick Start - SAS Drive](#)
- [Quick Start - Manage Data](#)
- [Quick Start - Model Manager](#)
- [Quick Start - Develop SAS Code in SAS Studio](#)
- [Quick Start - Develop Flows in SAS Studio](#)
- [Webinar - How Do I Use SAS Model Manager?](#)

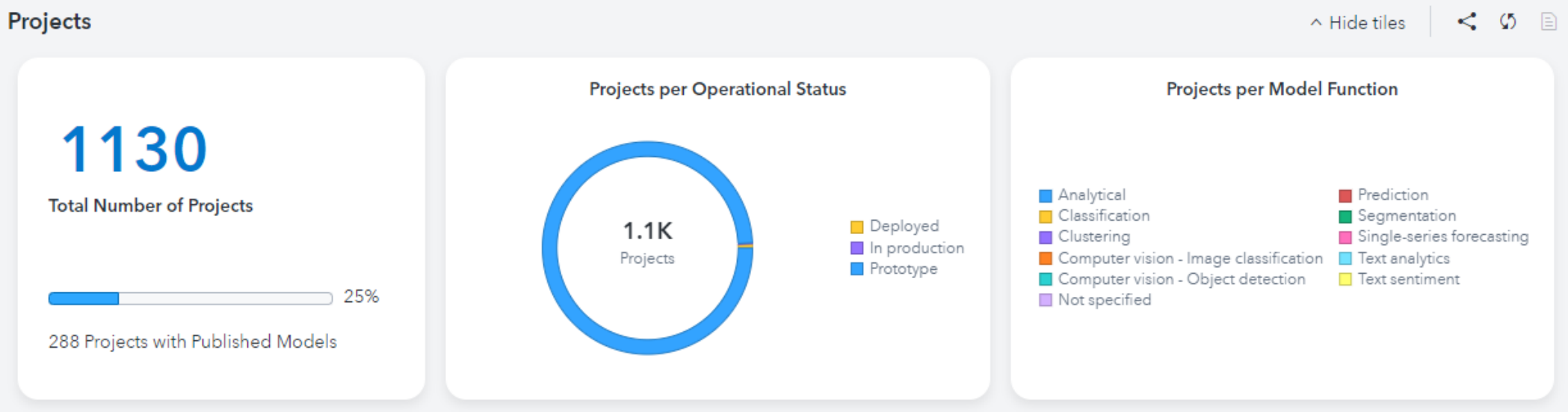
# Central Model Repository

# Central Model Repository – SAS Model Manager

- After the Data Scientist chose the champion model, it was registered to a central model repository called Model Manager. The Data Scientist passed the champion model to the repository by simply choosing to register it from the pipeline comparison page in Model Studio. This creates a project in SAS Model Manager.
- Navigate to the “Projects” tab by clicking the relevant icon on the left of the screen. This view lists all projects, project types, and deployments for the organization so the relevant teams and management can easily monitor these important metrics. In SAS Model Manager, you can save both SAS, Python and R models and manage them from a central place regardless of where you have developed them. In that way, models are treated as organizational assets, and proper governance and management are applied.



Example of the “Projects” view for large organizations. We can see below 1.1K projects and the associated project types.



# Central Model Repository – SAS Model Manager

The user can now search for the project created by the Data Scientist upon model registration. Registering a model will automatically instantiate a Model Manager project if one does not currently exist. Search for the project based on the name that the Data Scientist gave and click it to open it. Select 'Futurum\_MLOps' as a project.

**Projects** Hide tiles Share Refresh Menu

**1130**  
Total Number of Projects  
25%  
288 Projects with Published Models

**Projects per Operational Status**  
1.1K Projects  
Legend: Deployed (Yellow), In production (Purple), Prototype (Blue)

**Projects per Model Function**  
1.1K Projects  
Legend: Analytical (Blue), Classification (Yellow), Clustering (Purple), Computer vision - Image classification (Orange), Computer vision - Object detection (Teal), Not specified (Light Purple), Prediction (Red), Segmentation (Green), Single-series forecasting (Pink), Text analytics (Cyan), Text sentiment (Light Yellow)

Search: futurum Advanced search New Project

<input type="checkbox"/>	Name	Published Status	Model Function	Modified By	Date Modified	Tags
<input type="checkbox"/>	<u>Futurum</u>		Classification		Apr 8, 2024, 03:38:46 PM	




# Central Model Repository – SAS Model Manager

This project only has a single model called “Forest” (Pipeline 1) because this is the only model that has been registered for the project. The role has been set to champion with the star insignia. The other fields have automatically been filled by the system (model function, project version, score code type, algorithm, date modified, modified by and tags).

☰ Futurum

Models Variables Properties Files Scoring Performance Workflow History

🔍 Search name    Version: Version 1 (1.0) ▼

<input type="checkbox"/>	Name ↑	Role	Model Function	Project Version	Score Code Type	Algorithm	Date Modified	Modified By	Tags
<input type="checkbox"/>	<a href="#">Forest (Pipeline 2)</a>		Classification	Version 1 (1.0)	DS2 multi-type	Forest	Apr 8, 2024, 03:38:49 PM	Jordan.Bakerman@sas.com	<span>DS2 multi-type</span>

# Central Model Repository – SAS Model Manager

The “Variables” tab displays all the inputs and outputs for the entire project based on the champion model. This is also created automatically.

Futurum								
Models Variables Properties Files Scoring Performance Workflow History								
<input type="checkbox"/>	Name	Data Type	Input	Output	Length	Measurement		
<input type="checkbox"/>	Location	Character	✓		15	Nominal		
<input type="checkbox"/>	Marital_status	Character	✓		15	Nominal		
<input type="checkbox"/>	Market_condition	Character	✓		10	Nominal		
<input type="checkbox"/>	Monetary	Decimal	✓		8	Nominal		
<input type="checkbox"/>	Mortgages_num	Decimal	✓		8	Nominal		
<input type="checkbox"/>	Overdrafts_num	Decimal	✓		8	Interval		
<input type="checkbox"/>	Product_segment	Character	✓		1	Nominal		
<input type="checkbox"/>	Recency	Decimal	✓		8	Nominal		
<input type="checkbox"/>	Referrals_num	Decimal	✓		8	Nominal		
<input type="checkbox"/>	Regularity	Decimal	✓		8	Nominal		
<input type="checkbox"/>	Repaymentdelays_num	Decimal	✓		8	Nominal		
<input type="checkbox"/>	Risk_tolerance	Character	✓		20	Binary		
<input type="checkbox"/>	Savings_behavior	Character	✓		10	Binary		
<input type="checkbox"/>	Savings_num	Decimal	✓		8	Interval		
<input type="checkbox"/>	Socialmedia_usage	Character	✓		6	Nominal		
<input type="checkbox"/>	Survey_response	Character	✓		3	Binary		
<input type="checkbox"/>	Transaction_anomalies_num	Decimal	✓		8	Nominal		
<input type="checkbox"/>	Transaction_pattern	Character	✓		10	Binary		
<input type="checkbox"/>	EM_EVENTPROBABILITY	Decimal		✓	8			
<input type="checkbox"/>	EM_PROBABILITY	Decimal		✓	8			
<input type="checkbox"/>	P_Churn0	Decimal		✓	8			
<input type="checkbox"/>	P_Churn1	Decimal		✓	8			

# Central Model Repository – SAS Model Manager

The “Properties” tab provides additional information for the project based on the champion model. The champion model is essentially the guide for creating the project in SAS Model Manager. Nothing needs to be altered because SAS fills out the necessary values based on registering a model previously.

General	
Name:	Futurum
Description:	<input type="text"/>
Model function: ⓘ	Classification ▼
Operational status: ⓘ	Prototype ▼
Created by:	Jordan.Bakerman@sas.com
Date created:	Apr 8, 2024, 03:37:35 PM
Date modified:	Apr 8, 2024, 03:38:46 PM
Location:	/Model Repositories/DMRepository
Champion version:	Version 1
Champion model name:	Forest (Pipeline 2)
UUID:	96169739-5456-48f1-90e1-c2e3f659ed8d
External URL:	<input type="text"/>
External project ID:	4a37c724-9d99-451c-8aa7-4e0a32c2cf16

Model Evaluation	
Default training table:	cas-shared-default/CASUSER(Jordan.Bak... <input type="button" value="🗑️"/> <input type="button" value="Clear"/>
Target variable:	Churn ▼
Target level:	Binary ▼
Target event value:	1
Target values:	<input type="text"/>
Output event probability variable:	P_Churn1 ▼



# Automatic Generation of Deployment Files

# Automatic Generation of Deployment Files

Go back to the “Models” tab for the project and select/click the champion model. Then click on the “Files” tab. This view shows all the deployment artifacts that are automatically created upon model registration. It includes the score code file and other metadata included in JSON files.

SAS Model Manager - Manage Models

Futurum\_MIOps

Models Variables Properties Files Scoring Performance Workflow History

Search name Version: Version 2 (2.0)

Name	Role	Model Fu...	Project Ve...	Score Cod...	Algorithm	Date
Forest (Pipeline 1)		Classificatio	Version 2 (2.0)	DS2 multi-type	Forest	May 2, 09:16:

Forest (Pipeline 2) (1.1) Project: Futurum, Version 1 (1.0)

Files Variables Properties Versions

dmcas\_epscorecode.sas

SCORE CODE

- dmcas\_epscorecode.sas
- dmcas\_packagescorecode.sas

PROPERTIES AND METADATA

- AstoreMetadata.json
- dmcas\_fitstat.json
- dmcas\_lift.json
- dmcas\_misc.json
- dmcas\_miscTable.json
- dmcas\_modelInfo.json
- dmcas\_packagemetadata.json
- dmcas\_properties.json
- dmcas\_relativeimportance.json
- dmcas\_scorecode.json

```
1 /*-----*/
2 /* Product: Visual Data Mining and Machine Learning
3 /* Release Version: V2024.01 */
4 /* Component Version: V2024.01 */
5 /* CAS Version: V.04.00M0P01162024 */
6 /* SAS Version: V.04.00M0P011624 */
7 /* Site Number: 70180938 */
8 /* Host: sas-cas-server-default-client */
9 /* Encoding: utf-8 */
10 /* Java Encoding: UTF8 */
11 /* Locale: en_US */
12 /* Project GUID: 4a37c724-9d99-451c-8aa7-4e0a32c2cf16
13 /* Node GUID: 4ba0cb53-2601-4c0c-9e4d-1e1f58adbd8a
14 /* Node Id: 4H6O0R5ZJJ12TFTK3K196MY */
15 /* Algorithm: Forest */
16 /* Generated by: Jordan.Bakerman@sas.com */
17 /* Date: 08APR2024:17:41:59 */
18 /*-----*/
19 data sasep.out;
20 dcl package score 4H6O0R5ZJJ12TFTK3K196MY(1);
```

# Automatic Generation of Deployment Files

The “Variables and Properties” tab within the champion model shows project and model metadata needed to score new observations that have been automatically created by SAS. The “Versions” tab shows the model versions. In this case, only a single version has been created. If more than one model or more than one version is added, the user can choose which model to govern.

Forest (Pipeline 2) (1.1) | Project: [Futurum, Version 1 \(1.0\)](#)

Files Variables Properties Versions

<input type="checkbox"/>	Name	Data Type	Input
<input type="checkbox"/>	<a href="#">Amount_avg</a>	⊕ Decimal	✓
<input type="checkbox"/>	<a href="#">Balance_avg</a>	⊕ Decimal	✓
<input type="checkbox"/>	<a href="#">Behavior_segment</a>	⊕ Character	✓
<input type="checkbox"/>	<a href="#">Channel</a>	⊕ Character	✓
<input type="checkbox"/>	<a href="#">Churn_num</a>	⊕ Decimal	✓
<input type="checkbox"/>	<a href="#">Complaints_num</a>	⊕ Decimal	✓

Forest (Pipeline 1) (2.0) | Project: [Futurum, Version 1 \(1.0\)](#)

Files Variables Properties Versions

New Version Set Version

Versions	Displayed Ver...	Modified By	Published Status
2.0	✓	jordan.bakerman@sas.com	
🔒 1.1		jordan.bakerman@sas.com	Published
🔒 1.0		jordan.bakerman@sas.com	

Forest (Pipeline 2) (1.1) | Project: [Futurum, Version 1 \(1.0\)](#)

Files Variables Properties Versions

General

User-Defined

Tags

General	
Name:	Forest (Pipeline 2)
Description:	
Created by:	Jordan.Bakerman@sas.com
Date created:	Apr 8, 2024, 03:37:57 PM
Date modified:	Apr 8, 2024, 03:38:49 PM
Location:	/Model Repositories/DMRepository/Futurum/Version 1
Project name:	<a href="#">Futurum</a>

# Model Comparison Metrics and Graphs

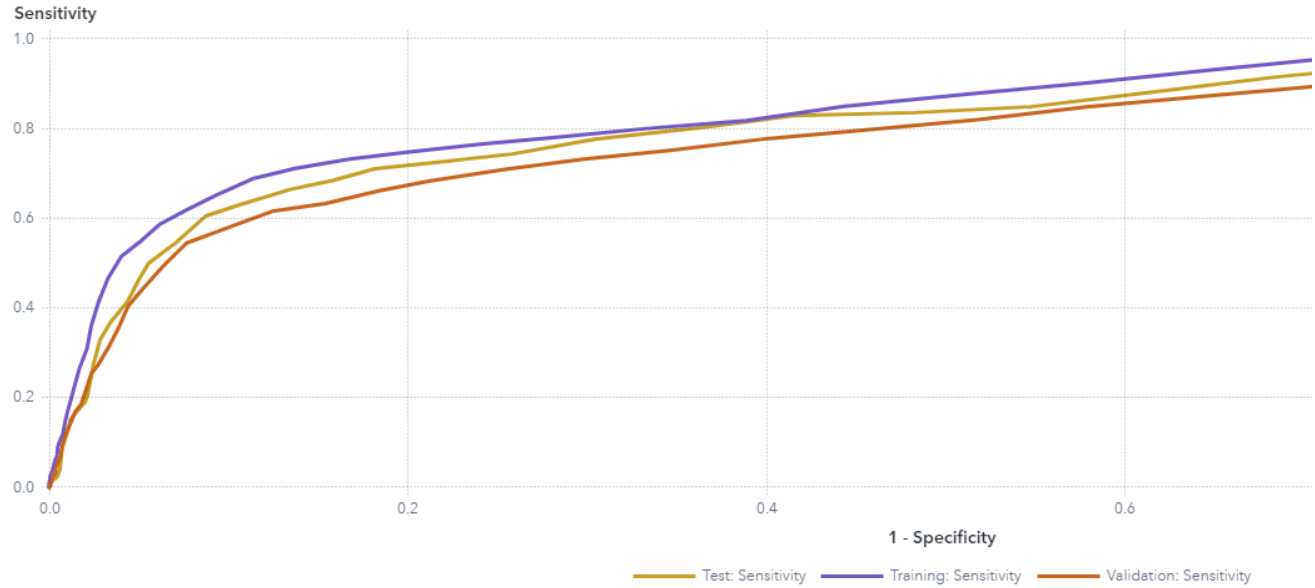
# Model Comparison Metrics and Graphs

Go back to the main “Project” page by clicking on the project name at the top. Select the checkbox next to the champion model and then select compare at the top right of the project. This view will compare any models we have registered in the project with fit statistics and graphics like lift and ROC. These views are automatically created.

The screenshot shows the SAS Model Studio interface. At the top, the project name is 'Forest (Pipeline 1) (1.0)' and the project path is 'Project: Futurum\_MLOps\_Version 2(2.0)'. Below this, there are tabs for 'Files', 'Variables', 'Properties', and 'Versions'. The 'Files' tab is active, showing a file named 'dmcas\_epscorecode.sas'. The file's properties and metadata are displayed below, including 'SCORE CODE' and 'PROPERTIES AND METADATA'.

Models	Forest (Pipeline 2) (1.1)
<b>Model Properties</b>	
Name	Forest (Pipeline 2)
Model function	Classification
Score code type	DS2 multi-type
Project version	Version 1 (1.0)
Location	/Model Repositories/DMRepository/Futurum/Version 1
Date modified	Apr 8, 2024, 03:38:49 PM
Modified by	Jordan.Bakerman@sas.com
Date published	
Description	

Test	
_C_	0.8110026688
_DIV_	1000
_KSCut_	0.21
_NObs_	1000
_TAU_	0.1605085085
_RASE_	0.3300123177
_miscEvent_	0.152
_ASE_	0.1089081298
_GAMMA_	0.6369891311
_GINI_	0.6220053376
_KS_	0.5300397219
_MCE_	0.152
_KSPostCutoff_	0
_MCLL_	0.360155016
_PartInd_	2
_TargetName_	Churn
_formattedPartition_	2
_miscKs_	0.165



# Model Scoring – Testing for Deployment

# Model Scoring Before Deployment

Then, navigate to the “Scoring” tab. Click “New Test” from the top right. The scoring test runs the model against new data to ensure the model will work in deployment. If you don’t see the data set you want to score, click the refresh button as shown below. Select “BANKING\_NEW” as the “Input Table” and click “Run.” The status will be a green checkmark when completed. Selecting the table icon under “Results” will allow the user to see the scoring test results and ensure there are predicted values. You see new variables created and the associated probabilities (P\_Churn1, EM\_EventProbability, EM\_Classification) that were given from the model test.

The screenshot shows a list of data sources under the 'Available' tab. A red box highlights the 'Refresh' button (a circular arrow icon) next to the 'BANKING\_ACCOUNT' entry.

Available	Data Sources	Import
Filter		
	<b>BANKING_ACCOUNT</b> 4/26/24, 10:28 PM • grant.wilson@sas.com	
	<b>BANKING_CUSTOMER</b> 4/26/24, 10:28 PM • grant.wilson@sas.com	
	<b>BANKING_FOR_SCORING</b> 4/29/24, 03:29 PM • spyridon.potamitis@sas.com	
	<b>BANKING_NEW</b> 4/29/24, 11:50 AM • spyridon.potamitis@sas.com	
	<b>BANKING_NEW</b> 4/29/24, 03:09 PM • spyridon.potamitis@sas.com	
	<b>SASVIYATYPES</b> 4/26/24, 08:11 PM • sas.admin-content-loader	

The 'New Test' dialog shows the following configuration:

- Name: Test 1
- Description: Enter a description
- Model: Forest (Pipeline 1) [Choose Model]
- Version: 2.0 (champion)
- Input table: BANKING\_NEW [Variables]
- Output data library: CASUSER(spyridon.potamitis@sas...)

Buttons: Save, Run, Cancel

The 'Scoring' tab shows a table with the following data:

Name	Results	Status	Model Name (Version)
Test 1			Forest (Pipeline 2) (1.1 - latest)

**Test Results**

Churn	P_Churn1
1	0.1477686378
1	0.2666602709
0	0.1342149011
0	0.177621548
0	0.100215183
0	0.2206893598
0	0.2268064135
0	0.1029484225
0	0.1000000000

# One-Click Deployment – Multiple Deployment Destinations

# One-Click Deployment

Go back to the main project page from the left of the screen and select the “Models” tab. Click the checkbox next to the champion model and then select “publish” on the top right of the project and choose “CAS (caslocal)” as the desired destination. (In our case we have configured CAS for batch scoring, but the deployment destination could be anything, including AWS, Azure, MAS, Git, Docker/SCR container, etc.). Specify the deployment model name, click the “Replace item with the same name” button and click publish. For this example, we chose CAS to deploy to for batch scoring purposes that use the SAS Viya engine. All the deployment artifacts move to the desired location, and we are ready to schedule scoring in production.

The screenshot shows the SAS Viya interface for managing models. At the top, there's a navigation bar with tabs for Models, Variables, Properties, Files, Scoring, Performance, Workflow, and History. Below the navigation bar, there's a search bar and a version selector set to 'Version 1 (1.0)'. On the right, there are buttons for 'Add models', 'Compare', and 'Publish', with the 'Publish' button highlighted by a red box. Below this is a table of models:

Name	Role	Model Fun...	Project Ver...	Score Code...	Algorithm	Date Modi...	Modified By	Tags
Forest (Pipeline 1)		Classification	Version 1 (1.0)	DS2 multi-type	Forest	Apr 24, 2024, 05:10:21 PM	jordan.bakerman@sas.com	Published, DS2 multi-type

The 'Publish Models' dialog box is shown with the following details:

- Destination:** CAS (caslocal) (highlighted with a red box)
- Tags:** (empty text field)
- ITEMS TO PUBLISH:**
  - Forest (Pipeline 1) (2.0)**
  - Model version to publish: 2.0 (champion)
  - Published name: Forest\_Pipeline\_1\_
  - Replace item with the same name (highlighted with a red box)
- Buttons: Publish, Cancel

- Amazon Web Services (Amazon-Web-Services)
- Azure (scrAzure)
- Azure Machine Learning (Microsoft-Azure-ML)
- CAS (GTP)
- CAS (SAS-Cloud-Analytic-Service)
- CAS (UKI)
- Git (GitRepoMAS)
- Git (GMP\_AndreyKhmyrov)
- SAS Micro Analytic Service (maslocal)

*Example with multiple publishing destinations*

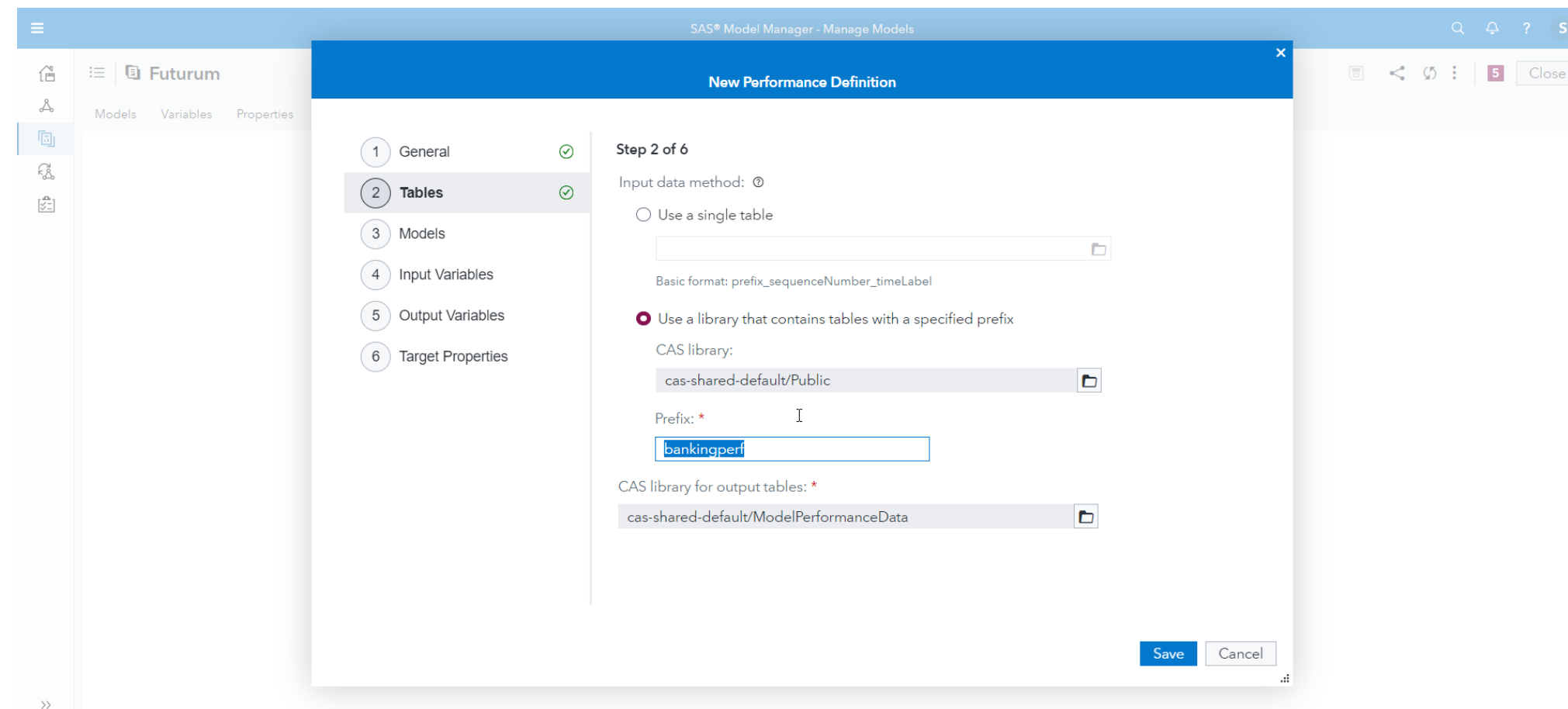
The 'Publishing Results' table shows the following data:

Name	Published Name	Status	Output
Forest (Pipeline 1)	Forest__Pipeline_1_	✔ Published successfully	<a href="#">Log</a>

# Model Performance Reports

# Model Performance Reports

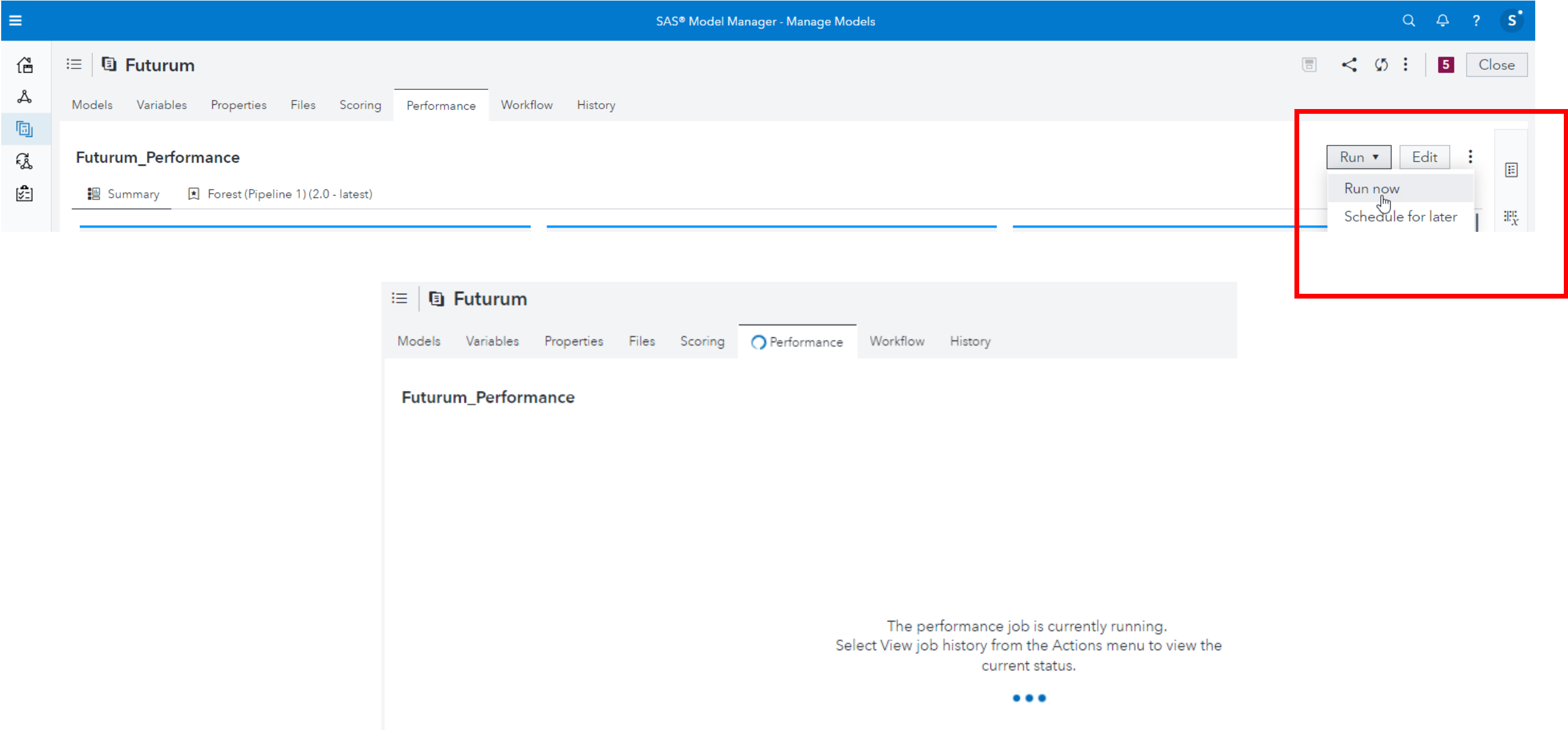
- Now that the model has been deployed, we must govern it and check for model degradation. Here, we assume new data has been collected over the following four quarters from deployment and has been placed in four new data sets (BANKINGPERF\_1\_Q1, BANKINGPERF\_2\_Q2, BANKINGPERF\_3\_Q3, BANKINGPERF\_4\_Q4). These data sets are already loaded in the “Public” library. You can locate and view them using the “Manage Data” application.
- Now go back to the Model Manager project, select the “Performance” tab, and click “New Definition.” (If the project is already created for you, then select “Edit” from the top right of the “Performance” tab.)
- Move to Step 2, on the “Tables tab” of the definition and choose “Use a library that contains tables with a specified prefix.” Choose the Public library where the data sets have been loaded. Set the prefix to the prefix for the four data sets, ‘bankingperf’ and click “Save.”
- The rest of the inputs in the definition should be set automatically for you.



# Model Performance Reports

## Run Now or Schedule Job

On the “Run” tab of the Performance page, you can “Run now” or schedule a performance report to be run in the future. Click “Run now” from the top right.



# Model Performance Reports

The initial report from running the performance definition provides details on the report. It scored 4,000 observations (1,000 for each quarter) and the overall misclassification rate. Now, select the model's name to view the performance reports in detail.

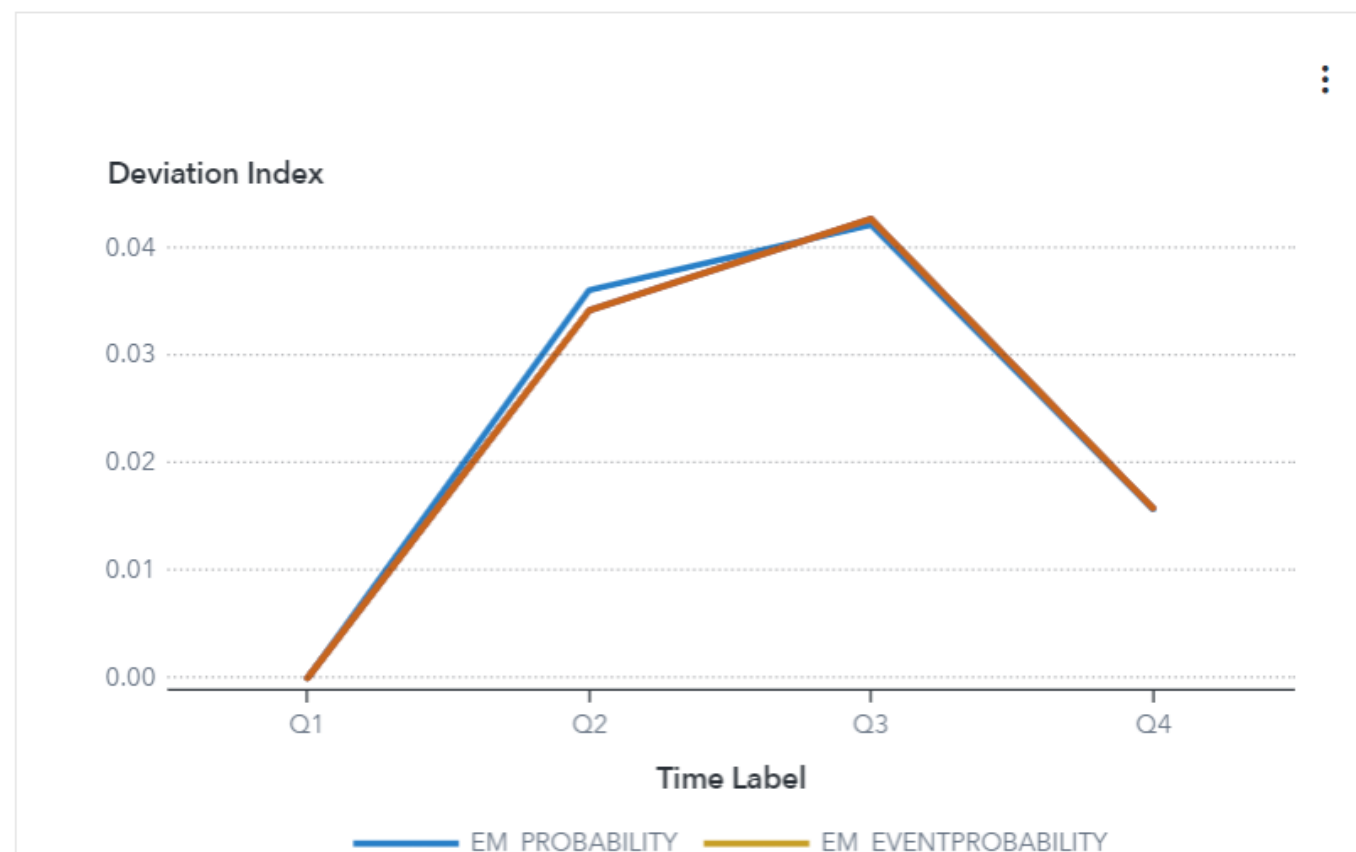
The screenshot displays the SAS Futurum Performance report for the model 'Forest (Pipeline 2) (1.1 - latest)'. The report is organized into several sections:

- Summary:** Shows the model name 'Forest (Pipeline 2) (1.1 - latest)' highlighted in a red box.
- Scored Observations:** Displays a large blue number '4,000'.
- Definition Details:** Lists the creator (Jordan.Bakerman@sas.com), last updated date (Apr 11, 2024, 02:59:23 PM), and the model name.
- Model Performance Rank:** Shows a rank of 1 for 'Forest (Pipeline 2)'.
- Performance Metrics:** A detailed view of the model's performance, including:
  - Misclassification: 21.00%
  - False positive rate: 9.49%
  - Recall: 57.14%
- Metadata:** Includes the last updated date (Apr 8, 2024, 03:38:49 PM), modified by (Jordan.Bakerman@sas.com), created by (Jordan.Bakerman@sas.com), and score code type (DS2 multi-type).

# Model Performance Reports

- Select the model's name next to the "Overview" tab on the page. This brings you to the performance report, which is one of the most important capabilities of SAS Model Manager as it streamlines model performance reporting in an automatic and comprehensive way. It shows the metrics and graphics for each quarter scored against the champion model and also describes each graphic in natural language.
- The report automatically generates the following graphics for each quarter (Input variable distribution, input variable binning, input variable characteristic, PSI out-of-bounds indicators, output variable stability, lift, ROC, Gini, KS trend, KS, feature contribution, standard KPI trend, PSI out-of-bounds indicators, and FCI out-of-bounds indicators).
- For some of the graphics, you can switch between variables to look for variable distribution drift. For some others, like below, you also get guidance regarding the metrics, how to interpret them and when action needs to be taken. See the example below.

## OUTPUT VARIABLE TRACKING



### Output Variable Stability

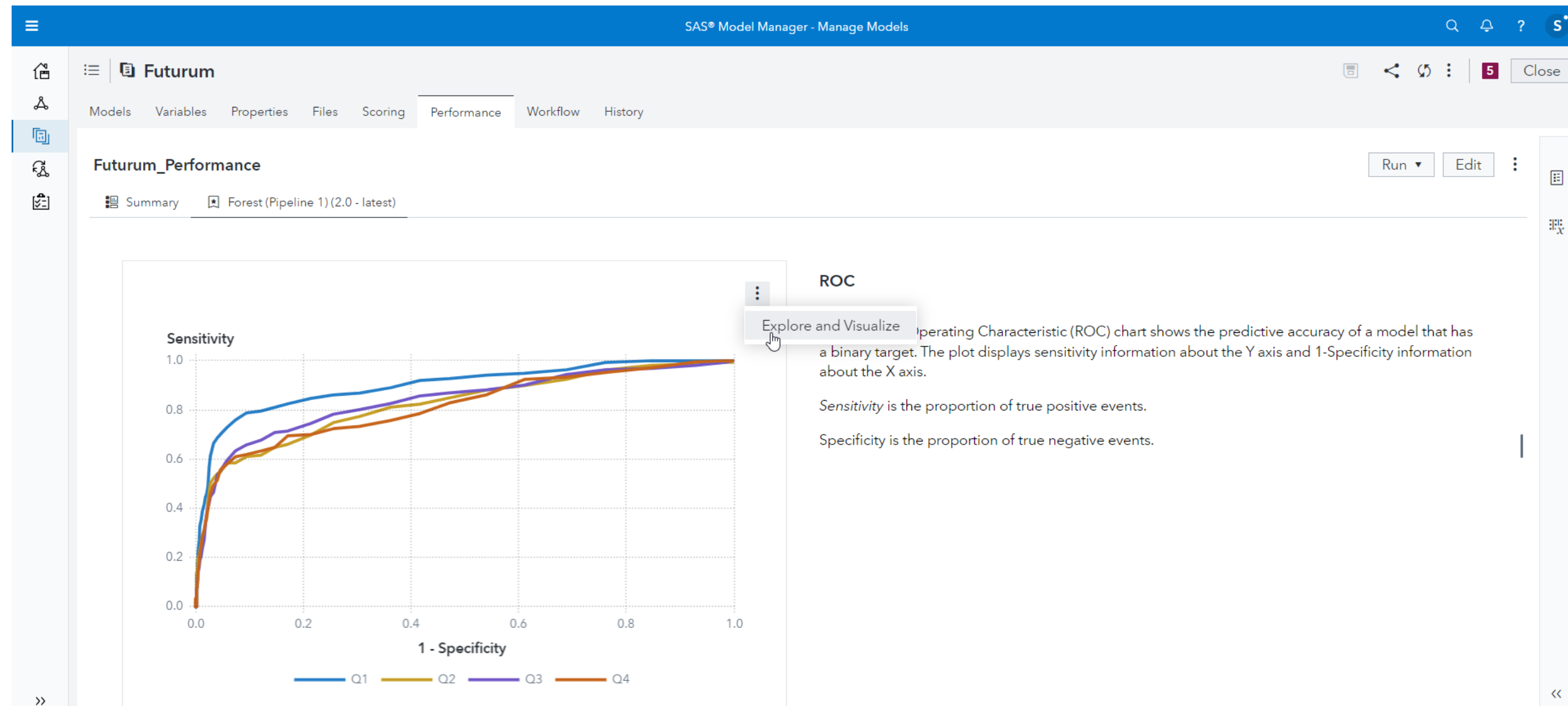
This chart evaluates changes in the distribution of scored output variable values as models score data over time. It also detects and quantifies shifts in the distribution of output variable values in the data that is produced by the models.

If an output variable from the training data and the output variable from the input data have identical distributions, then that output variable's deviation index is equal to 0. An output variable with a deviation index value that is greater than 0.10 and less than 0.25 is classified as having a mild deviation.

A variable that has a deviation index value that is greater than 0.30 is classified as having a significant deviation. Too much deviation in predictive variable output can indicate that model tuning, retraining, or replacement might be necessary.

# Model Performance Reports

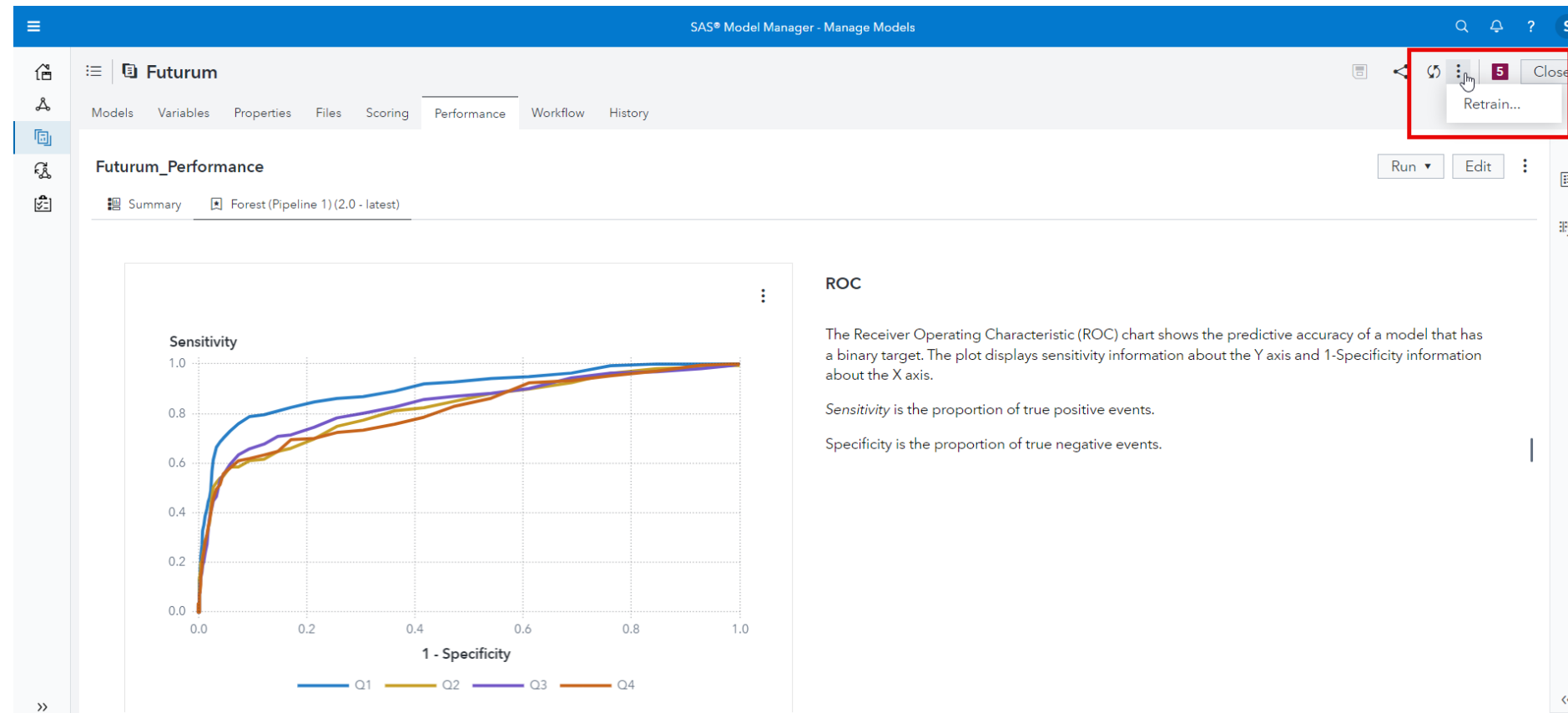
- Graphics like Lift and ROC look for model degradation. For example, the accuracy of the champion model according to the ROC chart is larger for Q1 than the following three quarters. Therefore, this may be a signal to retrain the model if the degradation is significant enough.
- Another useful capability for the MLOps Engineer is taking this graph and embedding it into custom reports. This can be done by clicking “Explore and Visualize,” where the user can take the underlined data used in creating this graph and create custom graphs. We won’t do that in our example, and we’ll move to retraining the models instead.



# Retraining

# Retraining

The actions button (three vertical dots) from the main project view allows the user to retrain models if degradation is significant. In this case, choose “Retrain now” with a new data table, which is already available and is called ‘BANKING\_RETRAIN’ in the “Public” Folder. Click send. The entire pipeline with the retrained model is rerun on the new data set.



The 'Retrain Project' dialog box is shown. It contains the following text and options:

- Select an option and send a request to the Model Studio application for retraining.
- Set the project retrain state to needed
- Retrain now with a new data table

Below the selected option, there is a text input field containing the path: `cas-shared-default/Public/BANKING_RETRAIN`. At the bottom right of the dialog, there are two buttons: 'Send' and 'Cancel'.

# Retraining

The run may take some time, and you can track its progress in the “Build Models” application. When the models are rerun, go to the “Models” tab and refresh the view. In the “Version” tab, select “All versions.” You should be able to see two models now. Click the boxes next to the models, and from the top right, select “Compare.”

The screenshot shows the SAS Build Models application interface. At the top, there is a navigation bar with tabs for Models, Variables, Properties, Files, Scoring, Performance, Workflow, and History. Below the navigation bar is a search bar and a dropdown menu labeled "Version:" with "All versions" selected. A red box highlights this dropdown menu. Below the search bar is a table with columns: Name, Role, Model Fun..., Project Ver..., Score Code..., Algorithm, Date Modi..., Modified By, and Tags. The table contains two rows: "Forest (Autotunir)" and "Logistic Regressi".

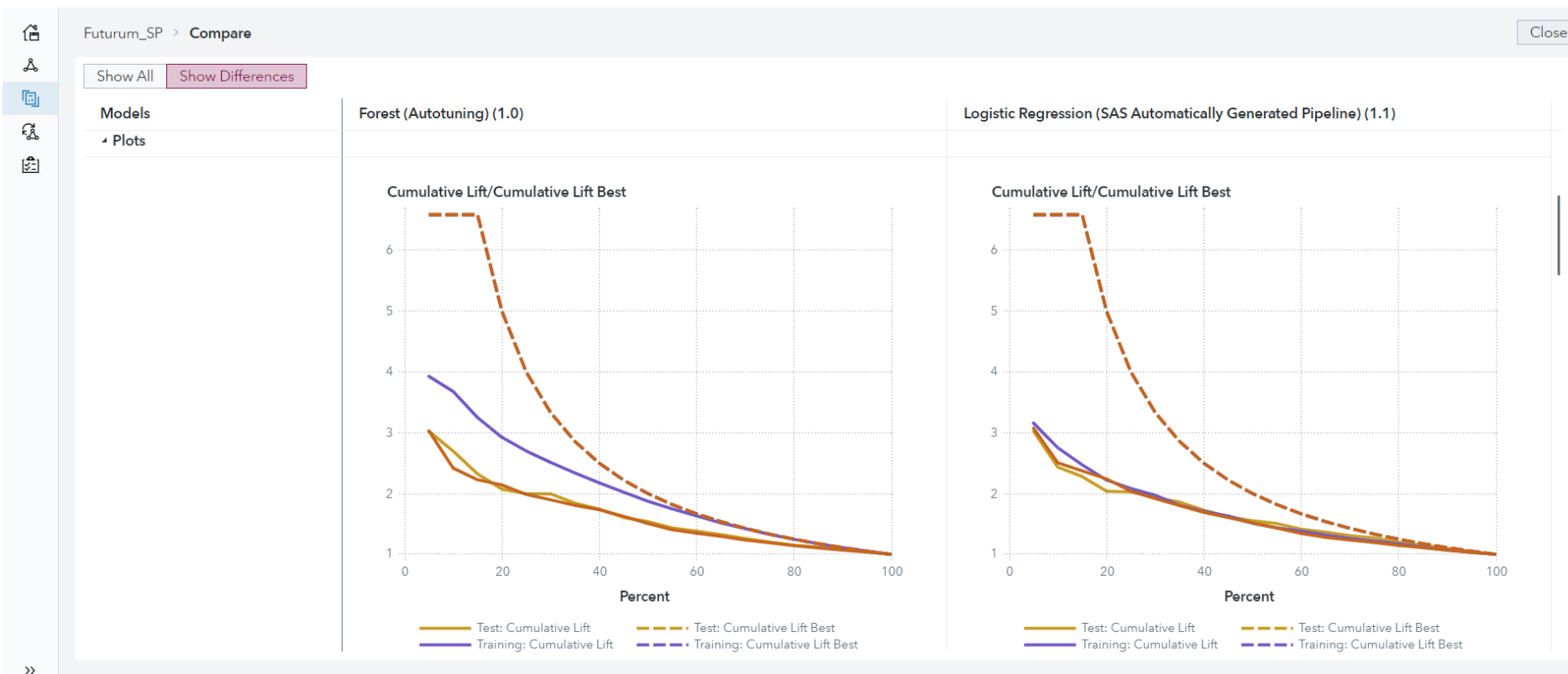
<input type="checkbox"/>	Name	Role	Model Fun...	Project Ver...	Score Code...	Algorithm	Date Modi...	Modified By	Tags
<input type="checkbox"/>	Forest (Autotunir		Classification	Version 2 (2.0)	DS2 multi-type	Forest	Apr 29, 2024, 05:15:57 PM	spyridon.potamitis@sas.com	DS2 multi-type
<input type="checkbox"/>	Logistic Regressi		Classification	Version 1 (1.0)	DATA step	Logistic regression	Apr 29, 2024, 02:05:25 PM	spyridon.potamitis@sas.com	DATA step

The screenshot shows the same SAS Build Models application interface as above. In this view, the checkboxes in the first column of the table are checked. A red box highlights the "Compare" button in the top right corner of the table area.

<input checked="" type="checkbox"/>	Name	Role	Model Fun...	Project Ver...	Score Code...	Algorithm	Date Modi...	Modified By	Tags
<input checked="" type="checkbox"/>	Forest (Autotunir		Classification	Version 2 (2.0)	DS2 multi-type	Forest	Apr 29, 2024, 05:15:57 PM	spyridon.potamitis@sas.com	DS2 multi-type
<input checked="" type="checkbox"/>	Logistic Regressi		Classification	Version 1 (1.0)	DATA step	Logistic regression	Apr 29, 2024, 02:05:25 PM	spyridon.potamitis@sas.com	DATA step

# Retraining

- Automatically, you will get a comparison of the two models. The older one and the new one. Click “Show differences” and investigate the results and the plots generated.
- After you compare the models, you can go back to the “Models” tab. Click on the new version of the model you created, and then click on the three-bullet icon as you see below. Now, you can set the new model as a champion in production or as a challenger to run in the background, and you can keep measuring the results as new data comes in.
- When you are happy with the performance of the new model, you can set it as a champion and move it to production.



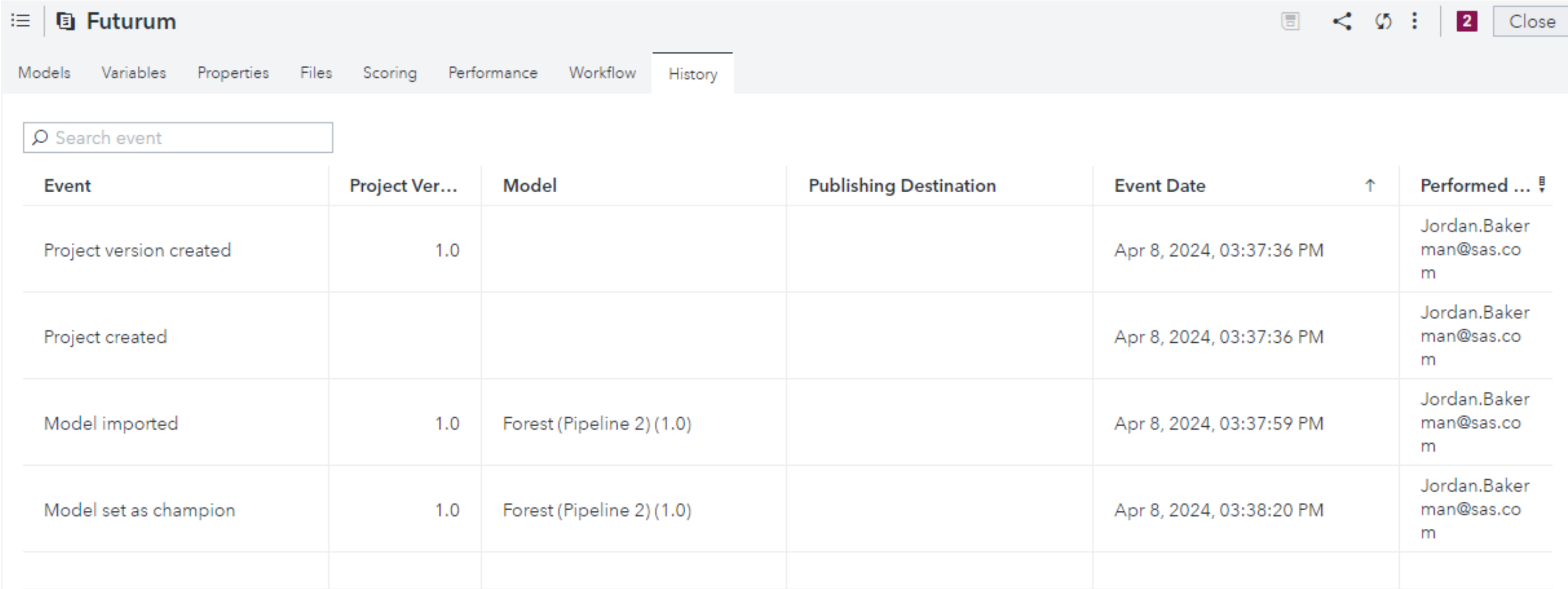
The screenshot shows the 'Manage Models' window in SAS Model Manager. It displays a table of models with columns: Name, Role, Model Fun..., Project Ver..., Score Code..., Algorithm, Date Modi..., Modified By, and Tags. The 'Forest (Autotuning)' model is selected. A context menu is open over the 'Forest (Autotuning)' model, showing options: Set as champion, Set as challenger, Clear role, Delete, Rename, and Export as ZIP.

Name	Role	Model Fun...	Project Ver...	Score Code...	Algorithm	Date Modi...	Modified By	Tags
Forest (Autotuning)	Classification	Version 2 (2.0)	DS2 multi-type	Forest	Apr 29, 2024, 05:15:57 PM	spyridon.potamitis@sas.com	DS2 multi-type	
Logistic Regression	Classification	Version 1 (1.0)	DATA step	Logistic regression	Apr 29, 2024, 02:05:25 PM	spyridon.potamitis@sas.com	DATA step	

# Versioning

# Versioning

Now, navigate to the “History” tab in the project view, which displays all the events for the project to ensure that the project is well governed and all the activities are tracked. To ensure regulatory purposes are met, you cannot modify this table. For example, the table shows when and who imported models, when champions are set, and model versions if they have been retrained.



Event	Project Ver...	Model	Publishing Destination	Event Date	Performed ...
Project version created	1.0			Apr 8, 2024, 03:37:36 PM	Jordan.Bakerman@sas.com
Project created				Apr 8, 2024, 03:37:36 PM	Jordan.Bakerman@sas.com
Model imported	1.0	Forest (Pipeline 2) (1.0)		Apr 8, 2024, 03:37:59 PM	Jordan.Bakerman@sas.com
Model set as champion	1.0	Forest (Pipeline 2) (1.0)		Apr 8, 2024, 03:38:20 PM	Jordan.Bakerman@sas.com

# Alerting

# Alerting

- Select the “Properties” tab in the Model Manager Project and then select “Model Evaluation.” Under “Model Evaluation,” SAS has generated the metadata for the alerting evaluation. The user can change the table if desired.
- Under “Model Assessment Criteria,” we can set an alert. Use the drop-down to set the assessment indicator to “Misclassification,” set the alert condition to “Greater than,” and type the Alert Threshold as “0.2.” SAS will now send alerts if the misclassification is greater than 0.2. You can trigger this rule via the Workflow Manager or call it an API from other applications.

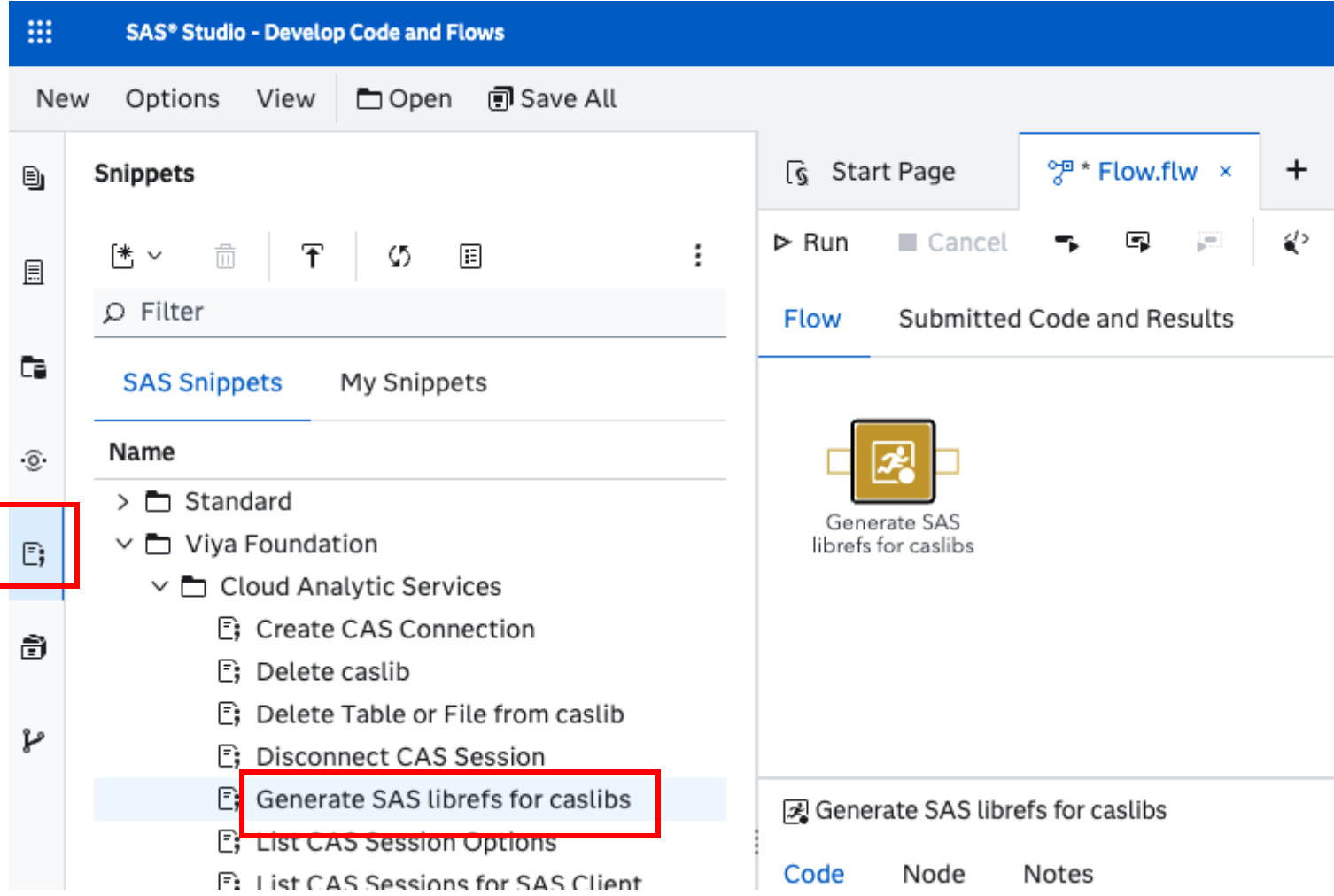
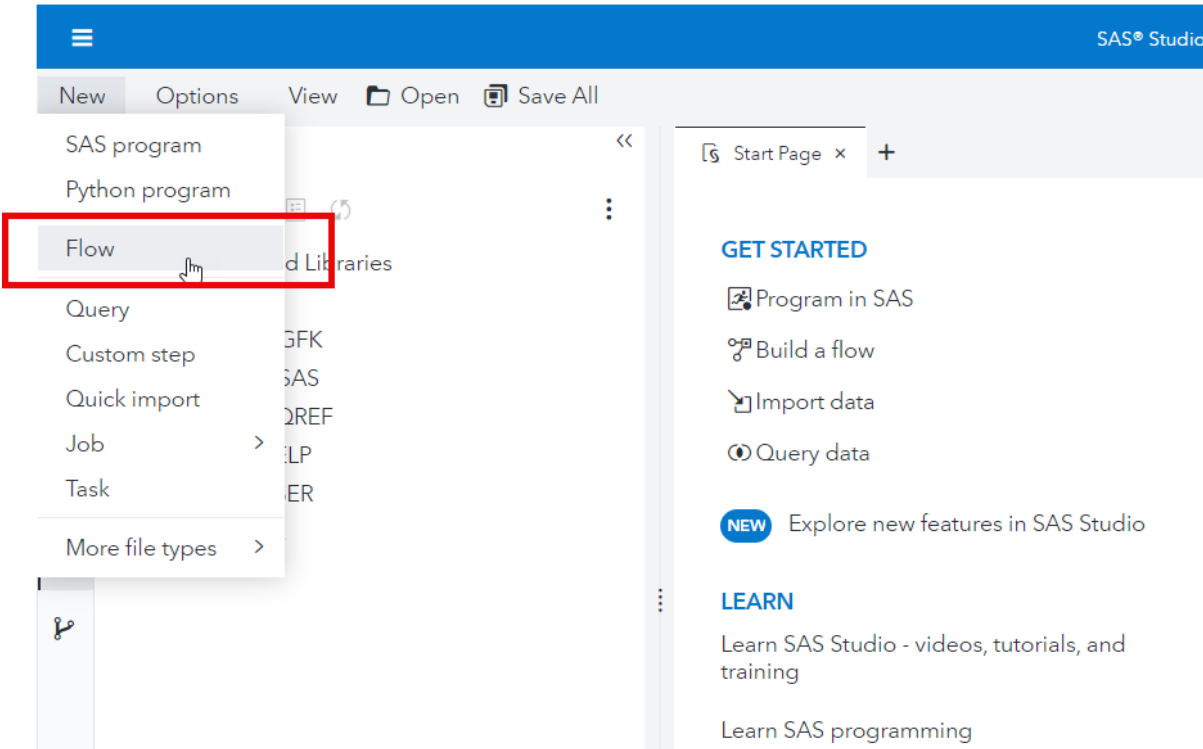
The screenshot shows the SAS Model Manager interface. The 'Properties' tab is selected in the top navigation bar. In the left sidebar, the 'Model Evaluation' sub-tab is selected. The main content area displays the 'Model Evaluation' configuration, including fields for 'Default training table', 'Target variable', 'Target level', 'Target event value', 'Target values', and 'Output event probability variable'. Below this, the 'MODEL ASSESSMENT CRITERIA' table is visible, with the following data:

Assessment Indicator	Priority	Occurrences	Alert Condition	Alert Threshold
Misclassification	High	1	Greater than	0.2

# Scheduling in Production

# Scheduling

- Since we have published a model in the previous step, we are ready to use this model in deployment. Move to the Applications Menu and select “Develop Code & Flows.” This will take us from SAS Model Manager to SAS Studio.
- Start by clicking New > Flow in the top right-hand corner. Then, make a connection to the CAS server. To do that we’ll use the Snippets option on the left of the screen. Snippets are lines of commonly used code or text that you get out of the box (you can also create your own). Navigate to Viya Foundation -> Cloud Analytics Services -> Generate SAS librefs for caslibs. Then drag and drop the snippet into your Flow.



# Scheduling

- Now, navigate to “SAS Steps” on the left of the screen to create our flow. To be able to view CAS tables, it’s time to run the “Generate SAS librefs for caslibs” step. Right-click on this and select “Run Node.” You’ll see a green checkmark that indicates that the step was run successfully.
- Now select “Data [Input and Output]” from the Steps menu and double-click on the “Table” option. Do this again, as we’ll need two tables for scoring: one that includes the data we want to score and one that includes the Model that we’ll score the data with. Use your mouse to connect the “Generate SAS librefs for caslibs” step with the first table icon. Now, click on the first table you have joined with the first step as below to see the options of the “Table.” Then click the little icon next to the “Select a library.” box.

The image consists of two side-by-side screenshots of the SAS Studio interface, illustrating the steps described in the text.

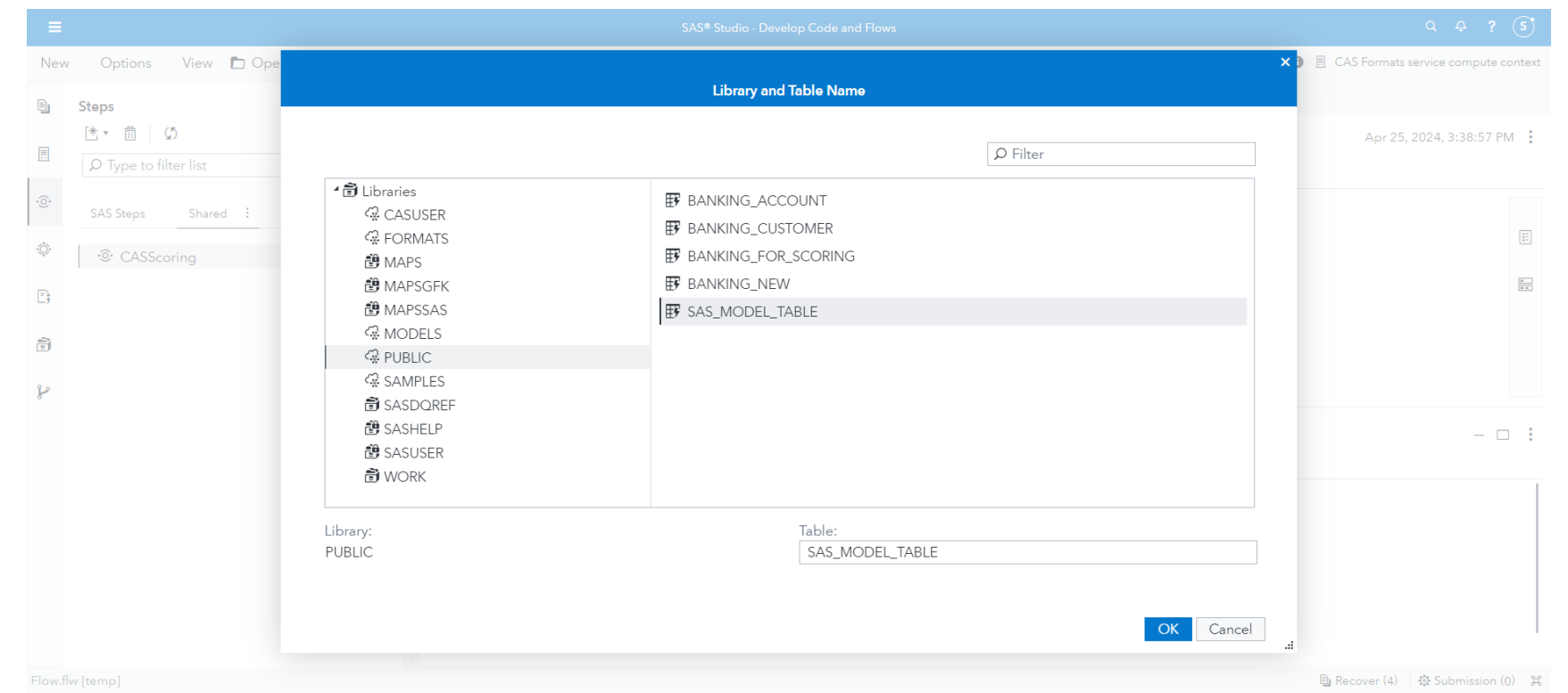
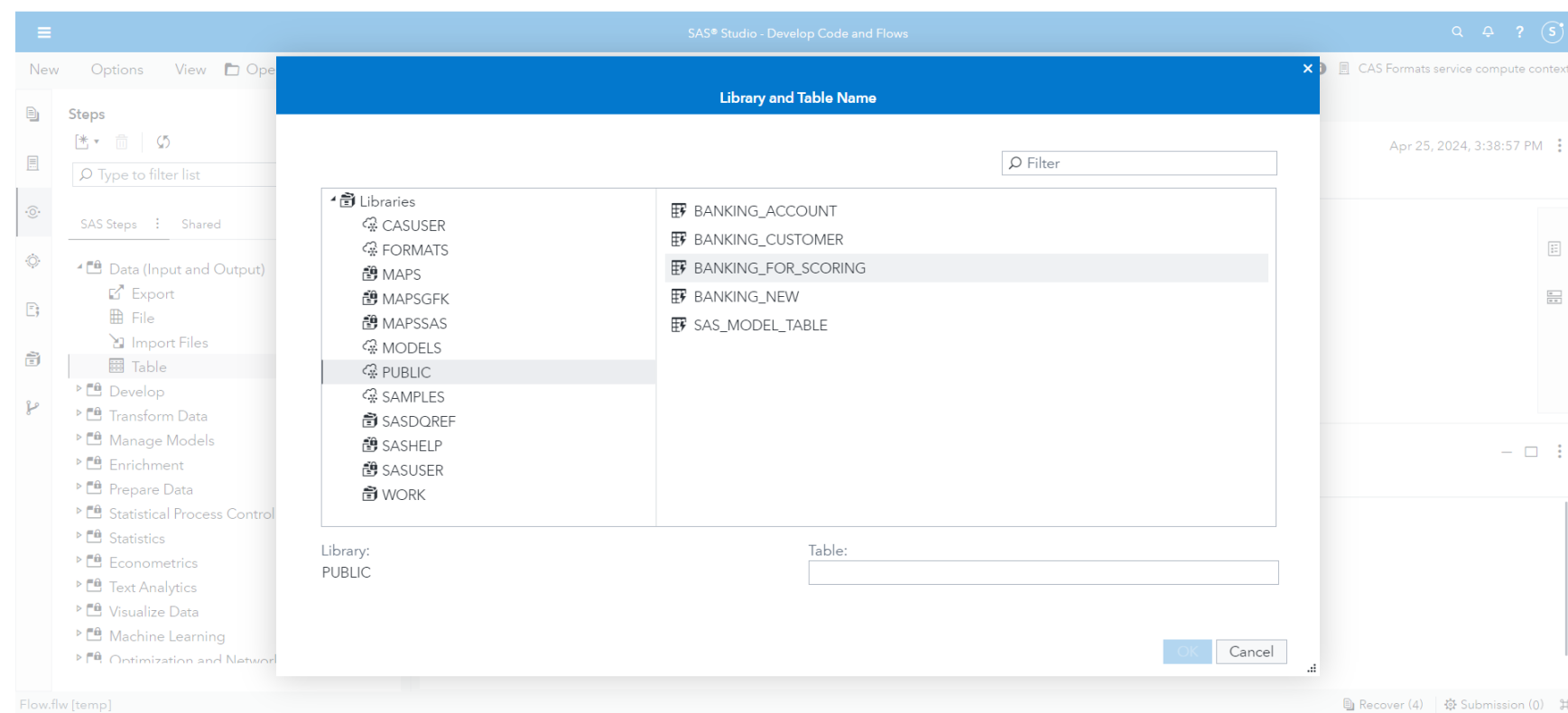
**Left Screenshot:** Shows the 'Steps' pane on the left with the 'Data (Input and Output)' category expanded and the 'Table' option selected. The main workspace displays a flow with a single step named 'Generate SAS librefs for caslibs'. The code editor below shows the following SAS code:

```
1 /******  
2 /* Create a default CAS session and create SAS librefs for existing caslibs */  
3 /* so that they are visible in the SAS Studio Libraries tree. */  
4 /******  
5  
6 cas;  
7 caslib_all_assign;  
8  
9  
10
```

**Right Screenshot:** Shows the 'Table' step configuration dialog. The 'Library:' field is highlighted with a red box, and the 'Select a library' dropdown menu is open. The 'Table name:' field is also highlighted with a red box.

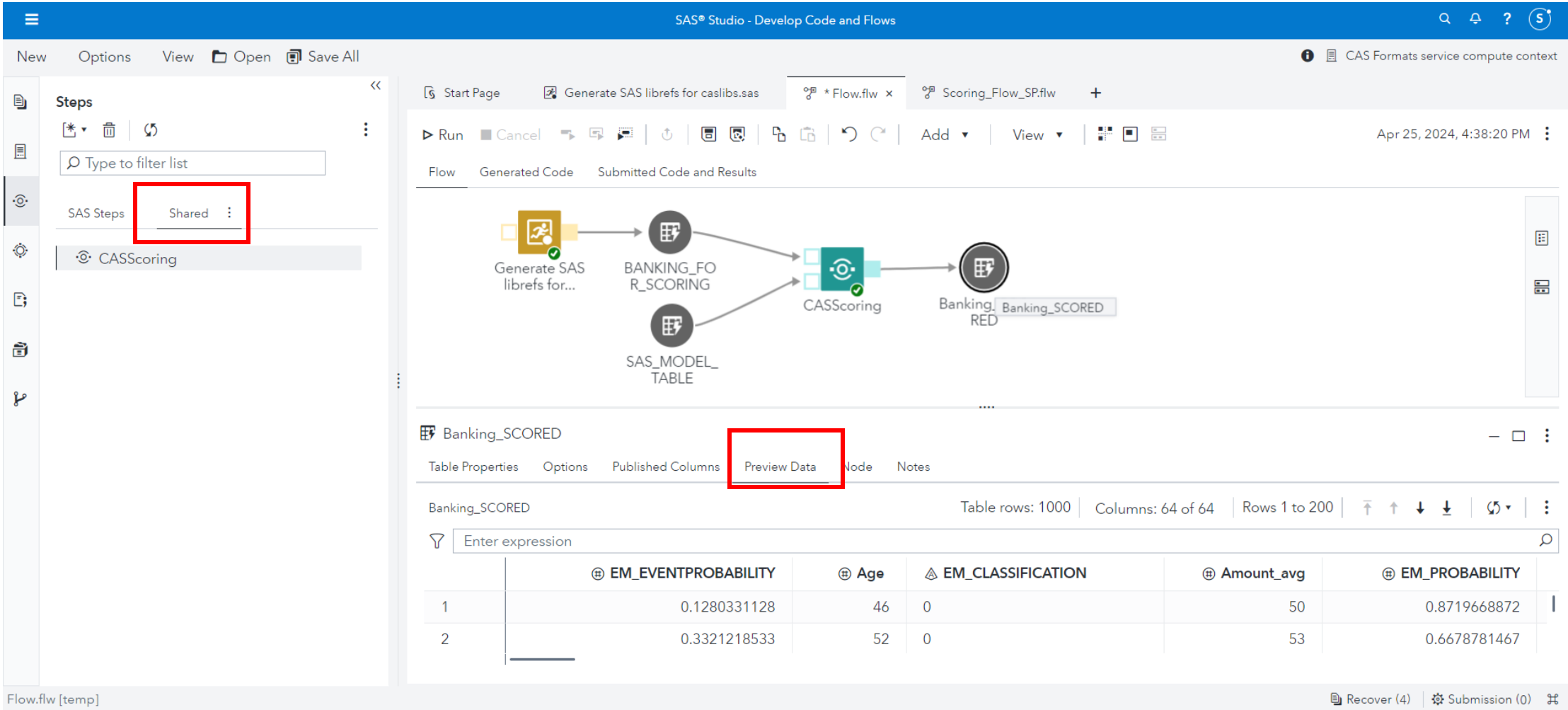
# Scheduling

- Navigate to “Public” and double-click the “Banking\_For\_Scoring” data. This data includes the observations that we want to score. Going back to the flow, right-click on the table step you created and select “Run to node.”
- Now move to the other “Table” step that we brought into the flow. Again, we must select the library and table where this data resides. This time, we’ll need to select the model to score the data. Navigate to “Public” and select “SAS\_MODEL\_TABLE.”



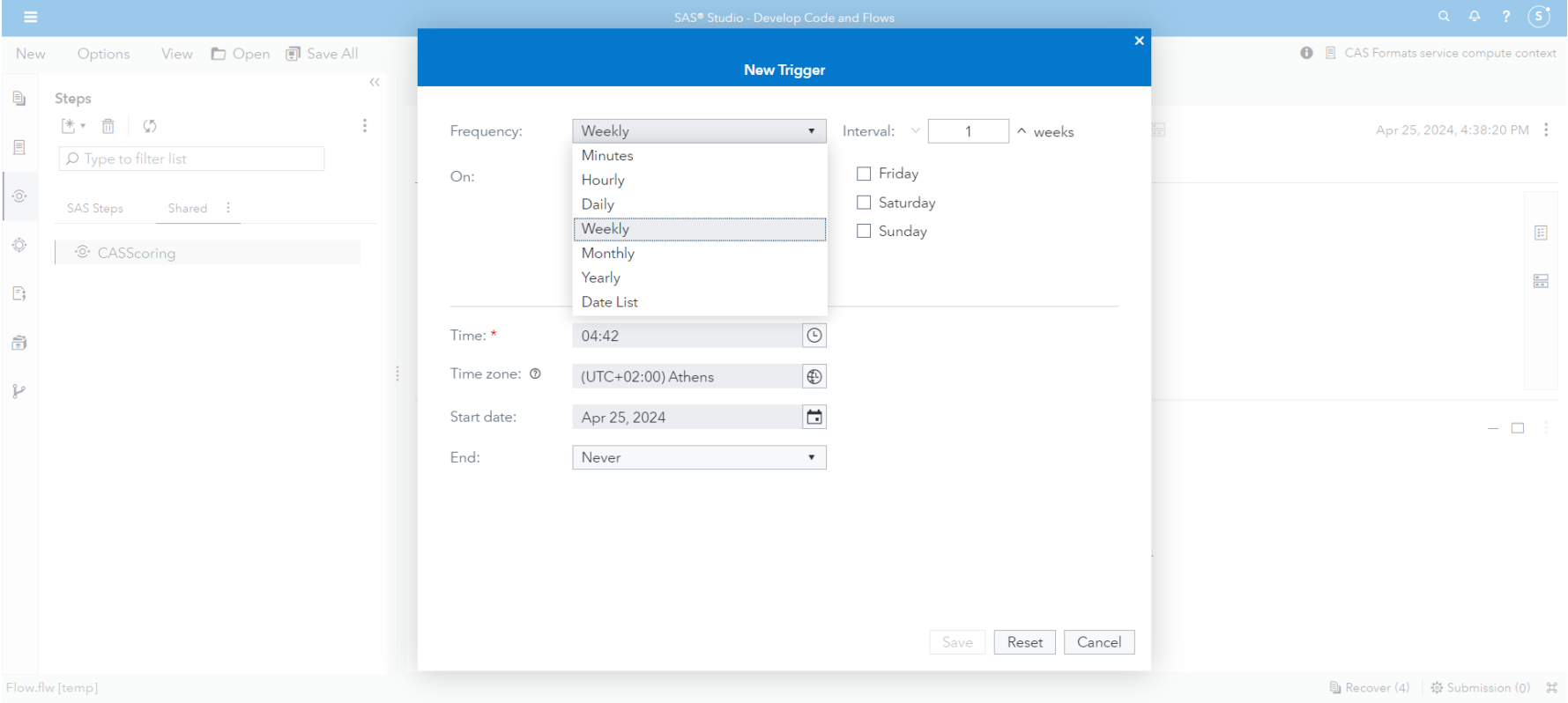
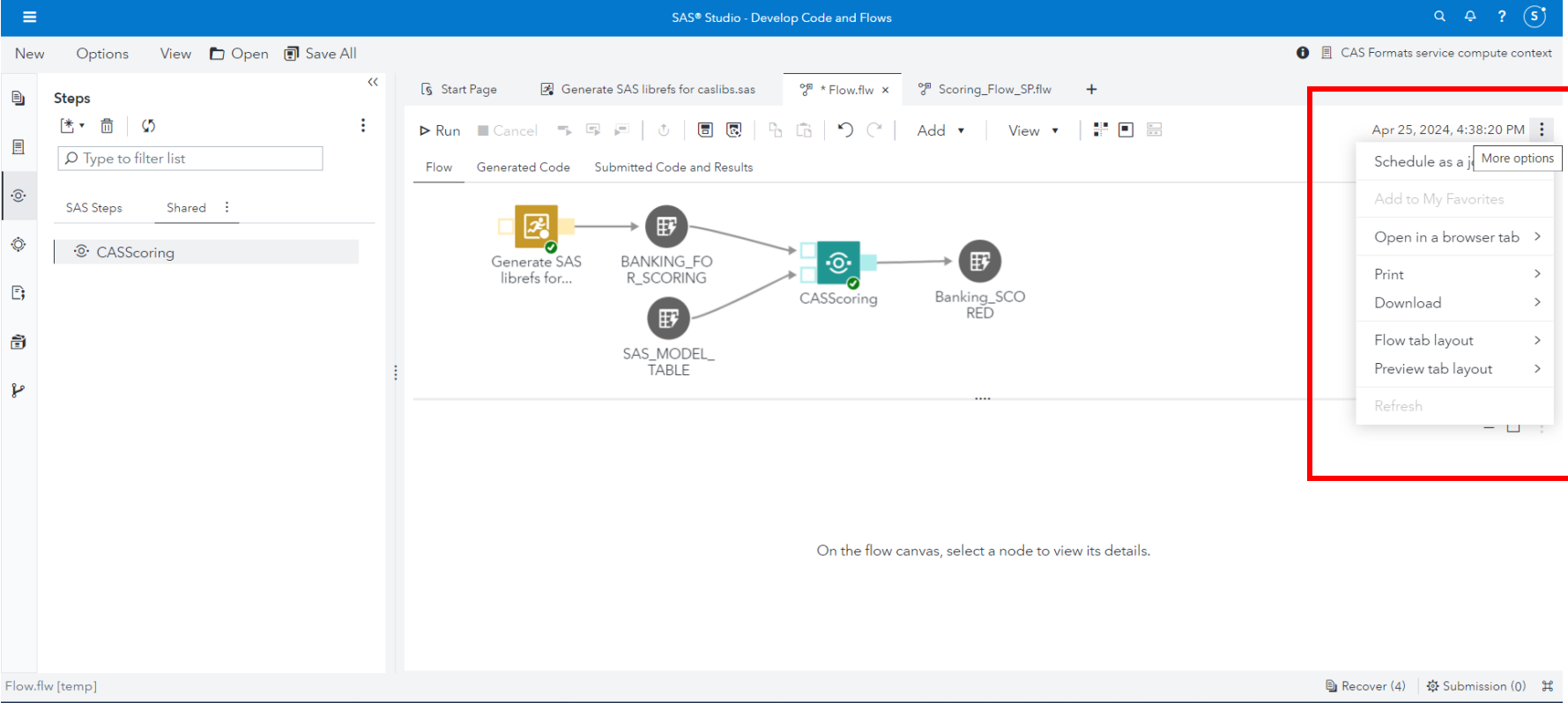
# Scheduling

- Now, we need to bring in our next step, which includes the scoring. In “Shared Steps,” drag and drop the “CASScoring” step to the flow. Connect the two table steps that we created to the two ports of the “CASScoring” step, as shown below. Click the CASScoring step, and in the “Options,” select the model name you created when you published your model in the previous steps (“Forest\_MLOps,” for example).
- The final thing you need to do is to connect the output of the “CASScoring” step with a new “Table” step so you can save the output of the flow. Go back to your “SAS Steps” pane and double-click on the “Table” step. Connect this with the “CASScoring” step. Select the library to save your output files. Select “Public” and then in the table field, “Banking\_Scored\_Current,” so you can create a new table and always save the latest version. Click OK. Then right-click on the final step you created and select “Run to Node.” After the run is complete, click on the final table step and select “Preview Data.” You should be able to see that your table is scored with the probability of churn for each customer.



# Scheduling

It's time to schedule the flow to run in a regular cadence based on our business needs and score new data. Click on the top right menu, as in the picture below, and select "Schedule as a job." Here, we can schedule our job to run at the interval we need (from the minute to any date we want), and our job will run automatically based on the settings in this pane. At this stage, the work of the MLOps Engineer is done.



**Thank you!**

