

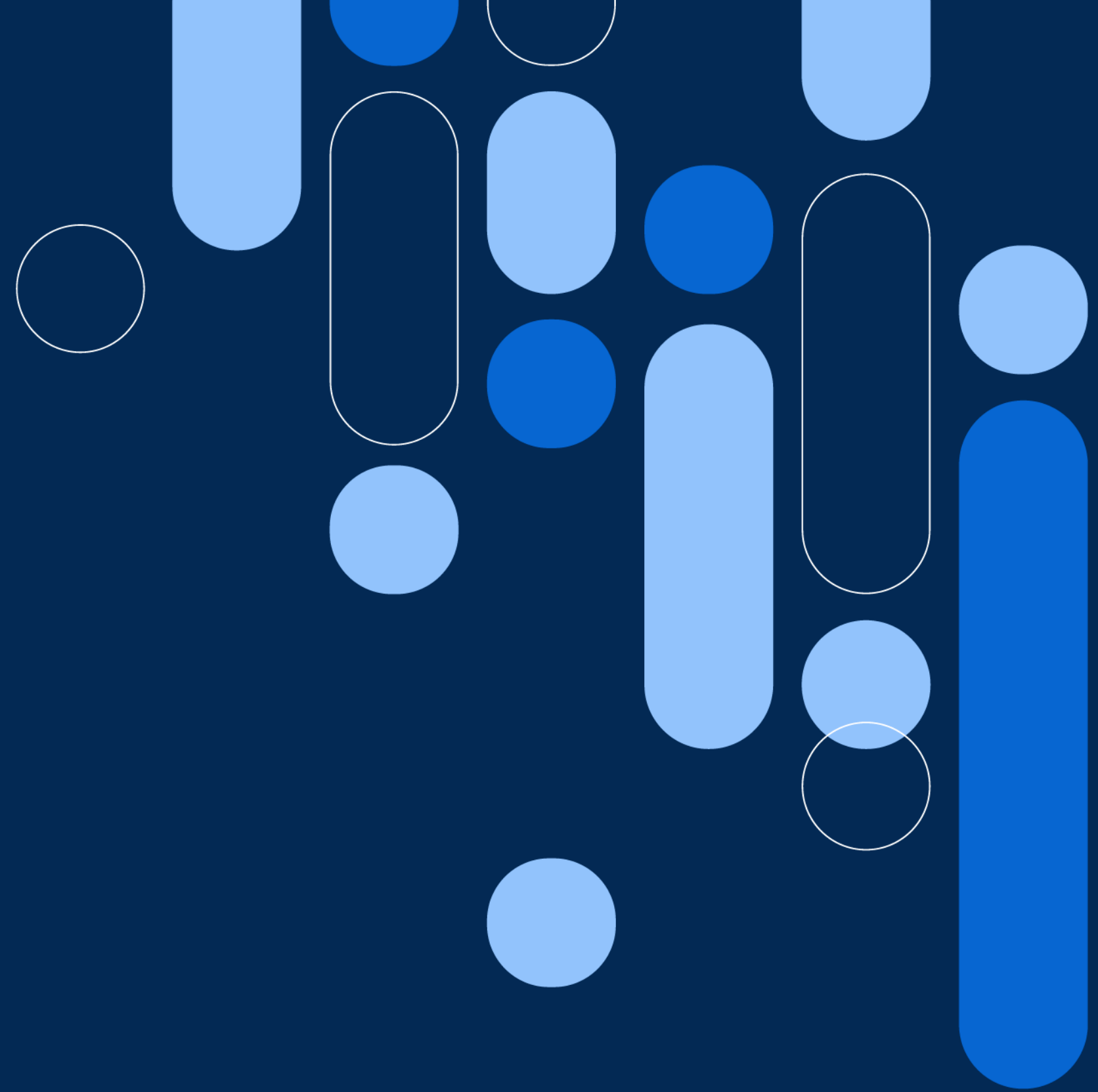
SAS[®] Viya[®] Trial

Deploy Insights Guide

MLOps Engineering and Business Analyst Tasks



Intro



Data and AI Life Cycle: Deploy Insights

The final stage in the data and AI life cycle is **Deploy Insights**. This stage focuses on operationalizing analytics at scale – putting models and decision logic into production so insights can drive consistent, real-world actions.

In this stage of the SAS Viya trial, you'll take validated models and make them available for use in production environments. You'll also see how models can be combined with business rules and data to create governed, auditable decision flows that support repeatable and explainable outcomes. Ongoing monitoring, versioning and updates help ensure analytics continue to perform as expected over time.

The Deploy Insights stage typically involves two key personas:

- **MLOps Engineers**, who are responsible for deploying models, managing versions, monitoring performance and maintaining reliability across environments.
- **Business Analysts or Decision Designers**, who design, test and govern decision flows that translate analytics into clear, consistent business actions.

This guide walks you through the key steps involved in deploying insights in SAS Viya, showing how to move from models and analysis to operational decisions that can be trusted, monitored and evolved as business needs change.

Resources

- [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?](#)
- [Ask the Expert: How Do I Use SAS Intelligent Decisioning?](#)
- [Ask the Expert: How do I Use SAS Intelligent Decisioning to Its Full Potential](#)
- [Ask the Expert: Why Do I Need SAS Intelligent Decisioning and SAS Model Manager to Achieve Analytics Success](#)
- [Ask the Expert: How do I Perform Customer Segmentation with SAS Intelligent Decisioning?](#)
- [Ask the Expert: Learning Snowflake with SAS Container Runtime + SAS Intelligent Decisioning](#)

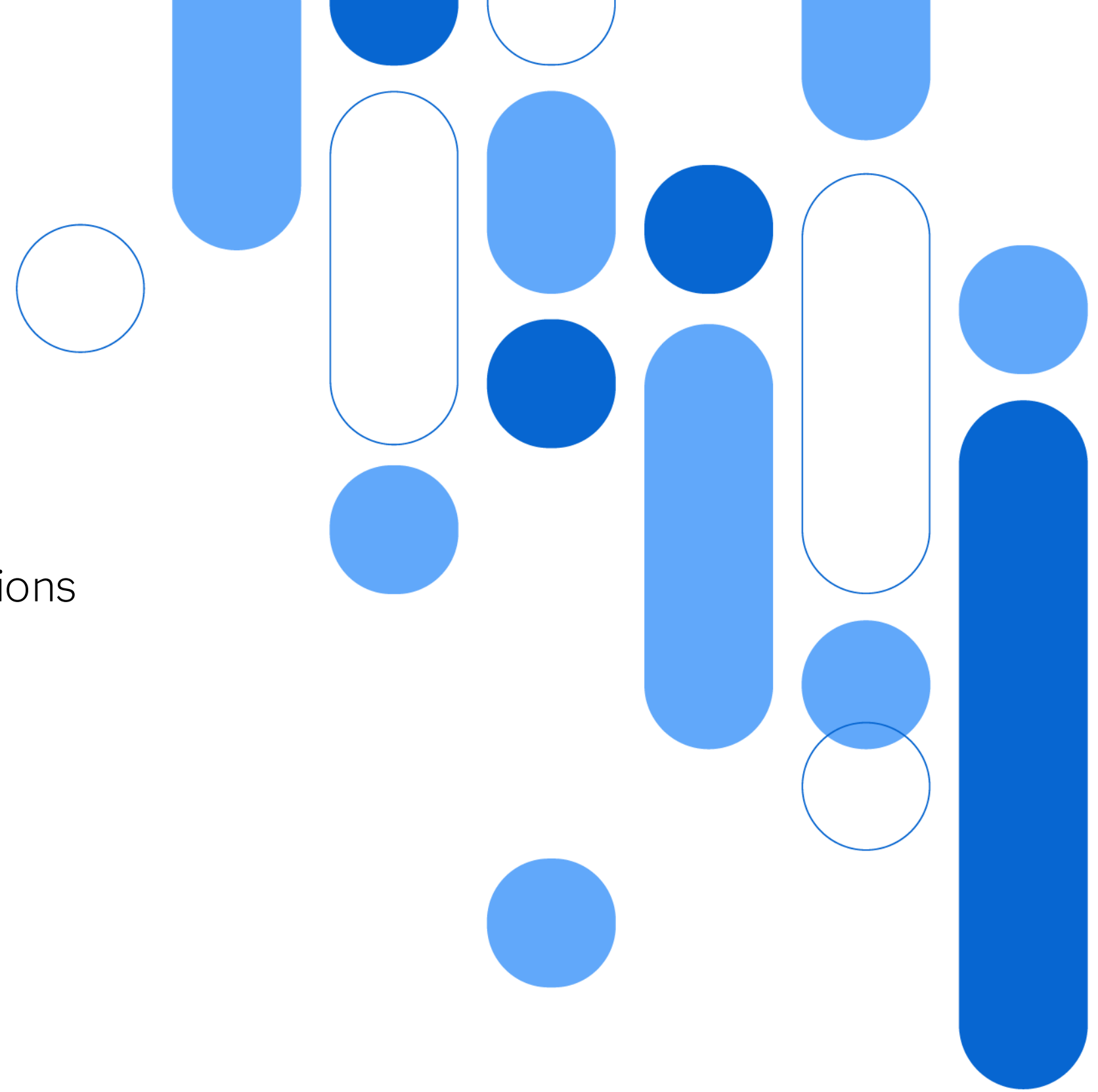
MLOps Engineer

Deploy and Maintain Models, Create Monitoring Reports

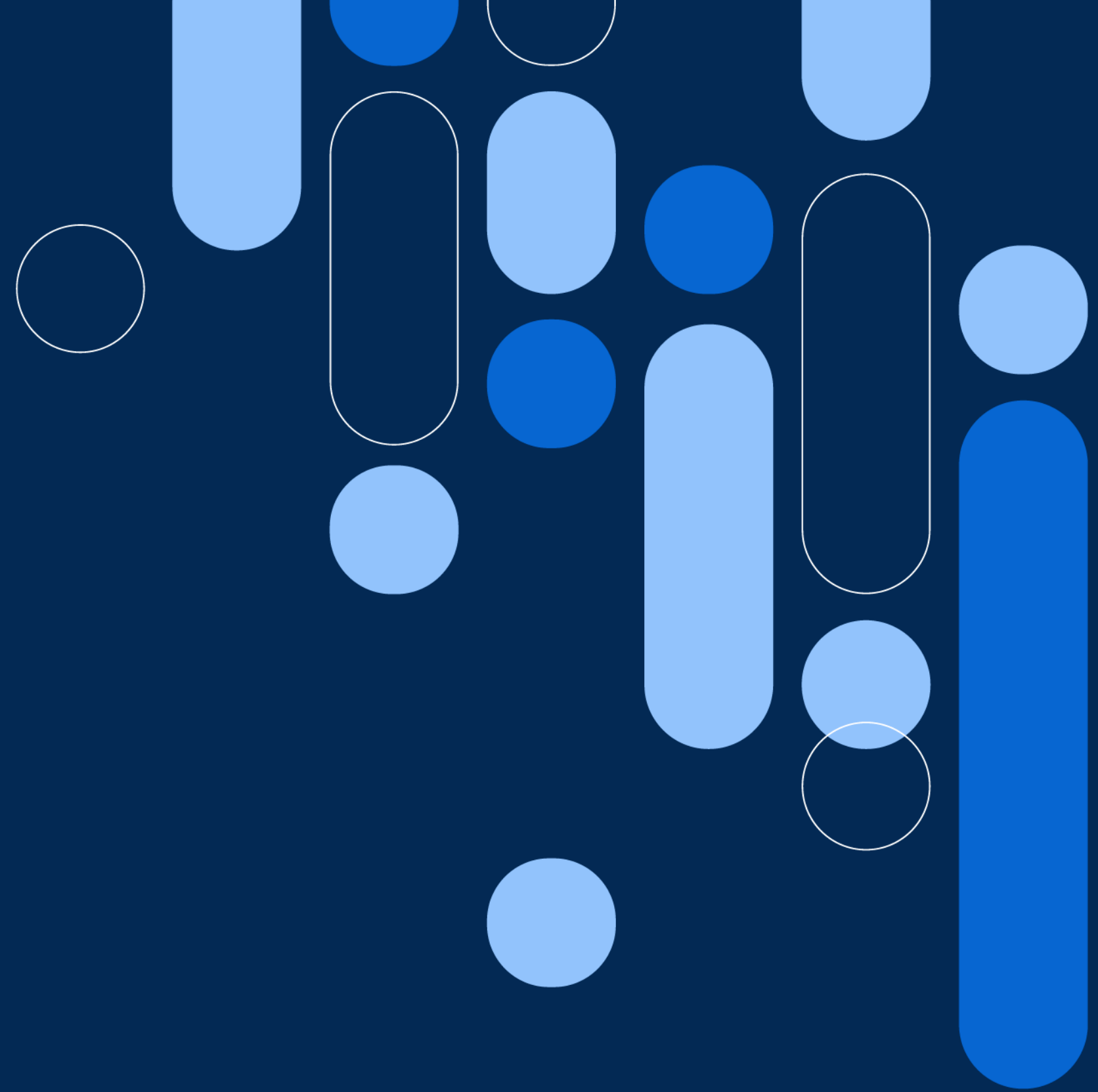


Tasks

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 Cards
7. Model Performance Reports
8. Retraining
9. Versioning
10. Alerting

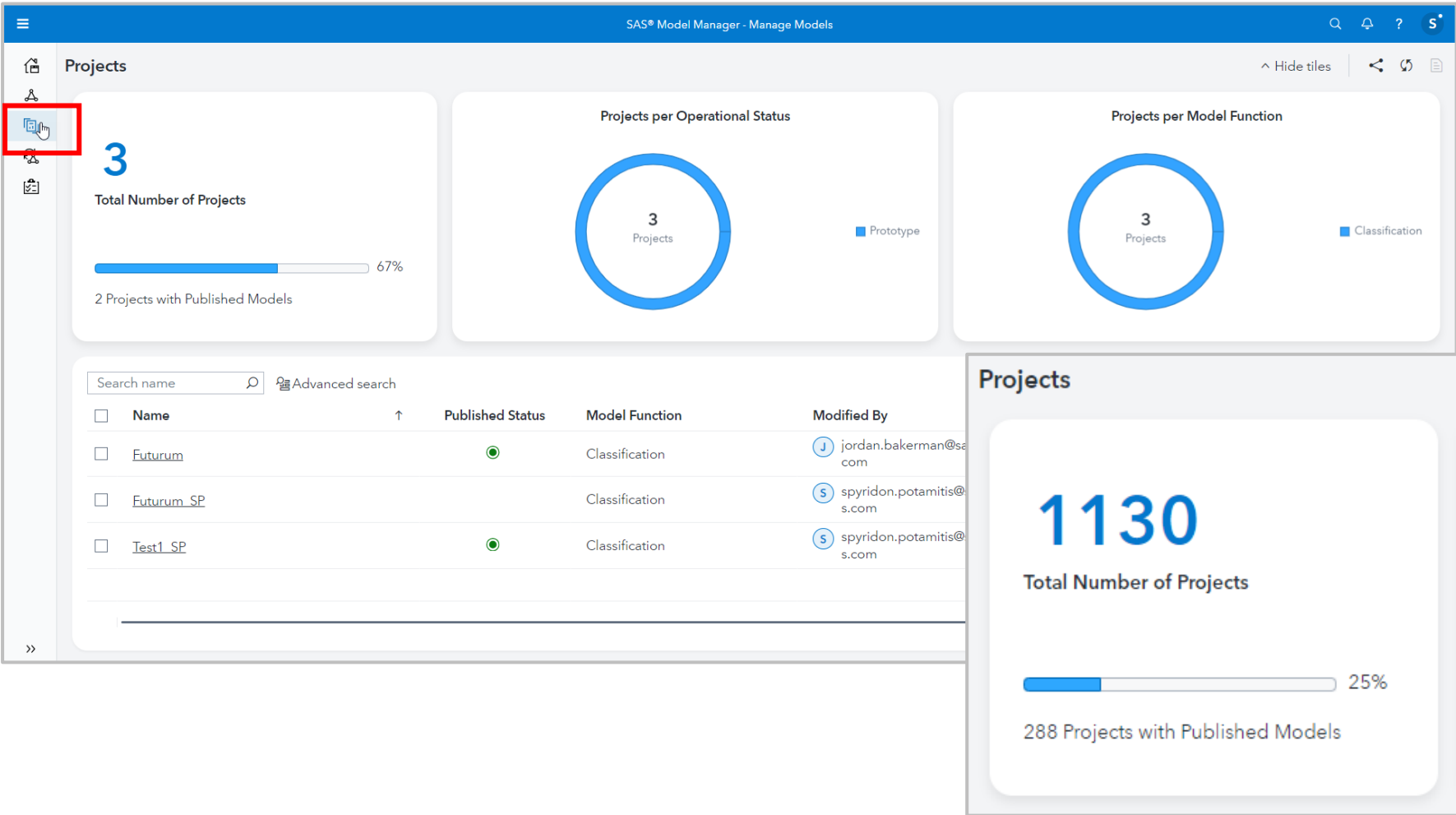


Central Model Repository

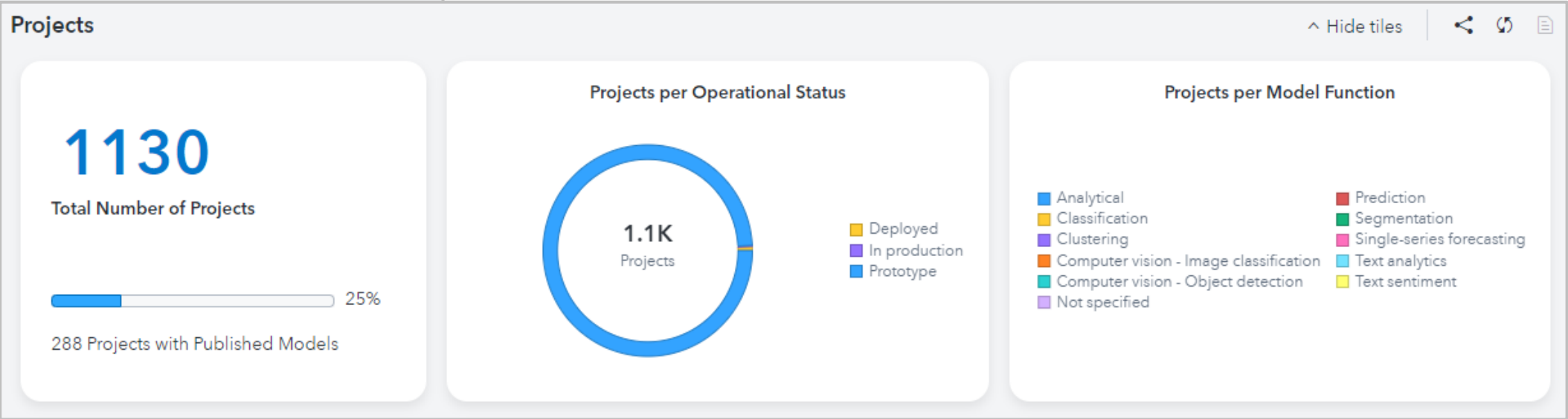


Central Model Repository – SAS Model Manager

- 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

- Deployed
- In production
- Prototype

Projects per Model Function
1.1K Projects

- Analytical
- Classification
- Clustering
- Computer vision - Image classification
- Computer vision - Object detection
- Not specified
- Prediction
- Segmentation
- Single-series forecasting
- Text analytics
- Text sentiment

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).

The screenshot shows the SAS Model Manager interface for a project named 'Futurum'. The interface includes 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 version dropdown menu set to 'Version 1 (1.0)'. The main content is a table listing models. The table has columns for Name, Role, Model Function, Project Version, Score Code Type, Algorithm, Date Modified, Modified By, and Tags. One model is listed: 'Forest (Pipeline 2)' with a role of 'Champion' (indicated by a star icon), a model function of 'Classification', project version 'Version 1 (1.0)', score code type 'DS2 multi-type', algorithm 'Forest', date modified 'Apr 8, 2024, 03:38:49 PM', modified by 'Jordan.Bakerman@sas.com', and a tag 'DS2 multi-type'.

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

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.

<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.

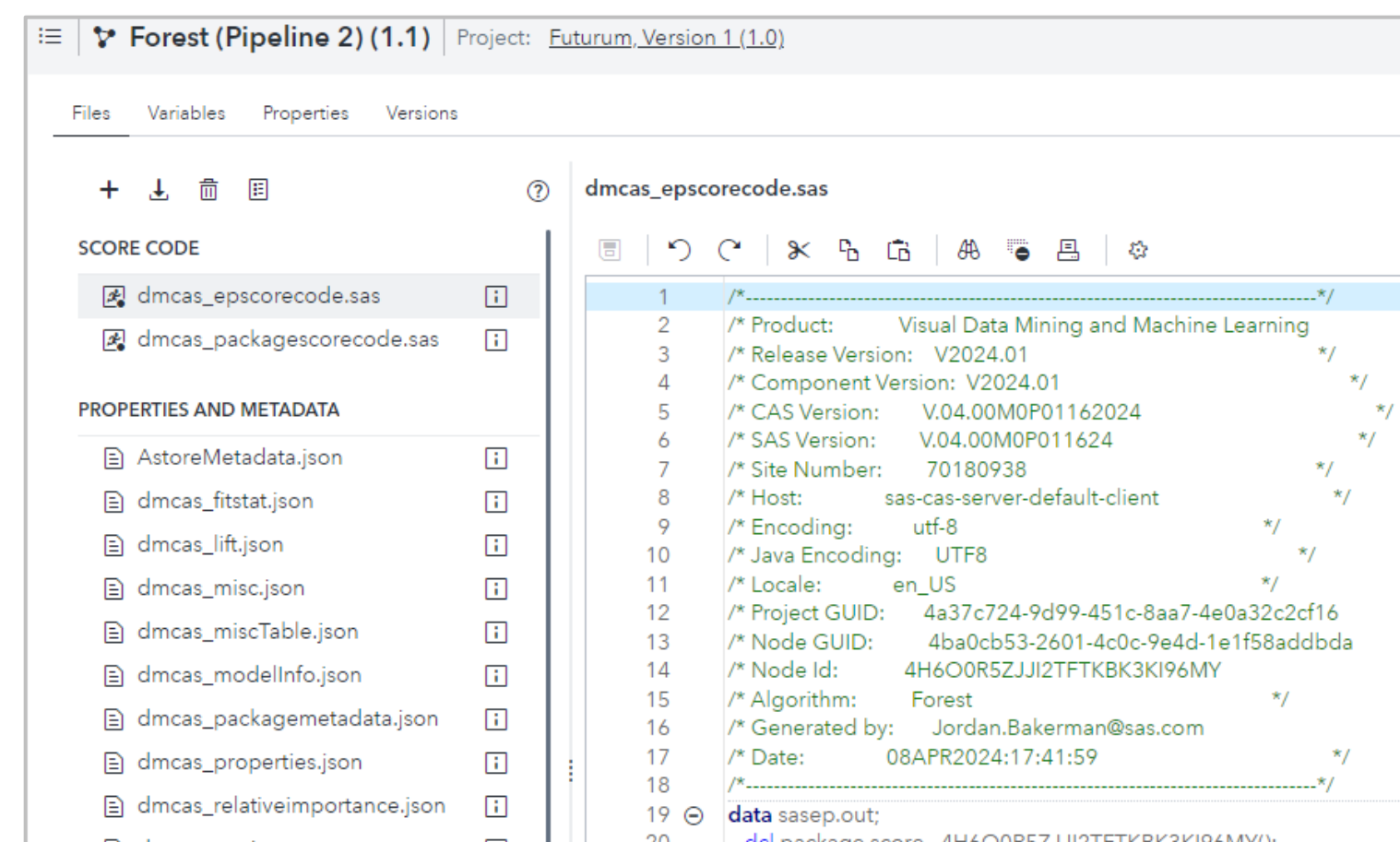
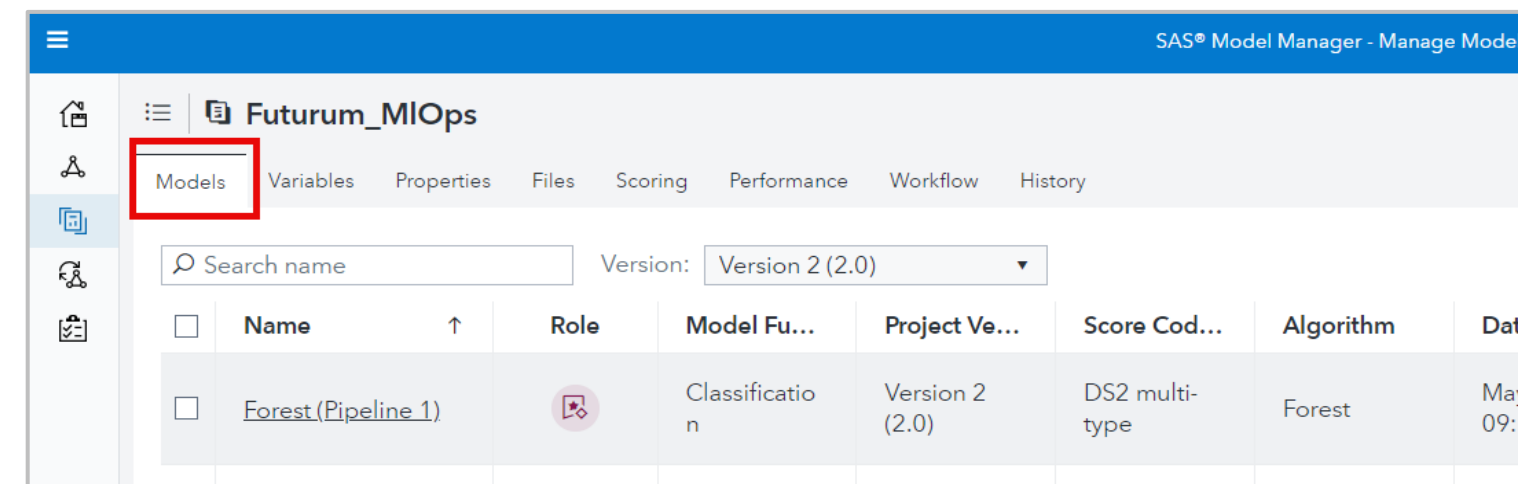
Model Evaluation	
Default training table:	<input type="text" value="cas-shared-default/CASUSER(Jordan.Bak..."/> <input type="button" value="Clear"/>
Target variable:	<input type="text" value="Churn"/>
Target level:	<input type="text" value="Binary"/>
Target event value:	<input type="text" value="1"/>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

General	
Name:	Futurum
Description:	<input type="text"/>
Model function: ⓘ	<input type="text" value="Classification"/>
Operational status: ⓘ	<input type="text" value="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:	<input type="text" value="4a37c724-9d99-451c-8aa7-4e0a32c2cf16"/>

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.



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.

The screenshot displays the SAS Viya interface for a model named "Forest (Pipeline 1) (2.0)" within the project "Futurum, Version 1 (1.0)". The interface is divided into several sections:

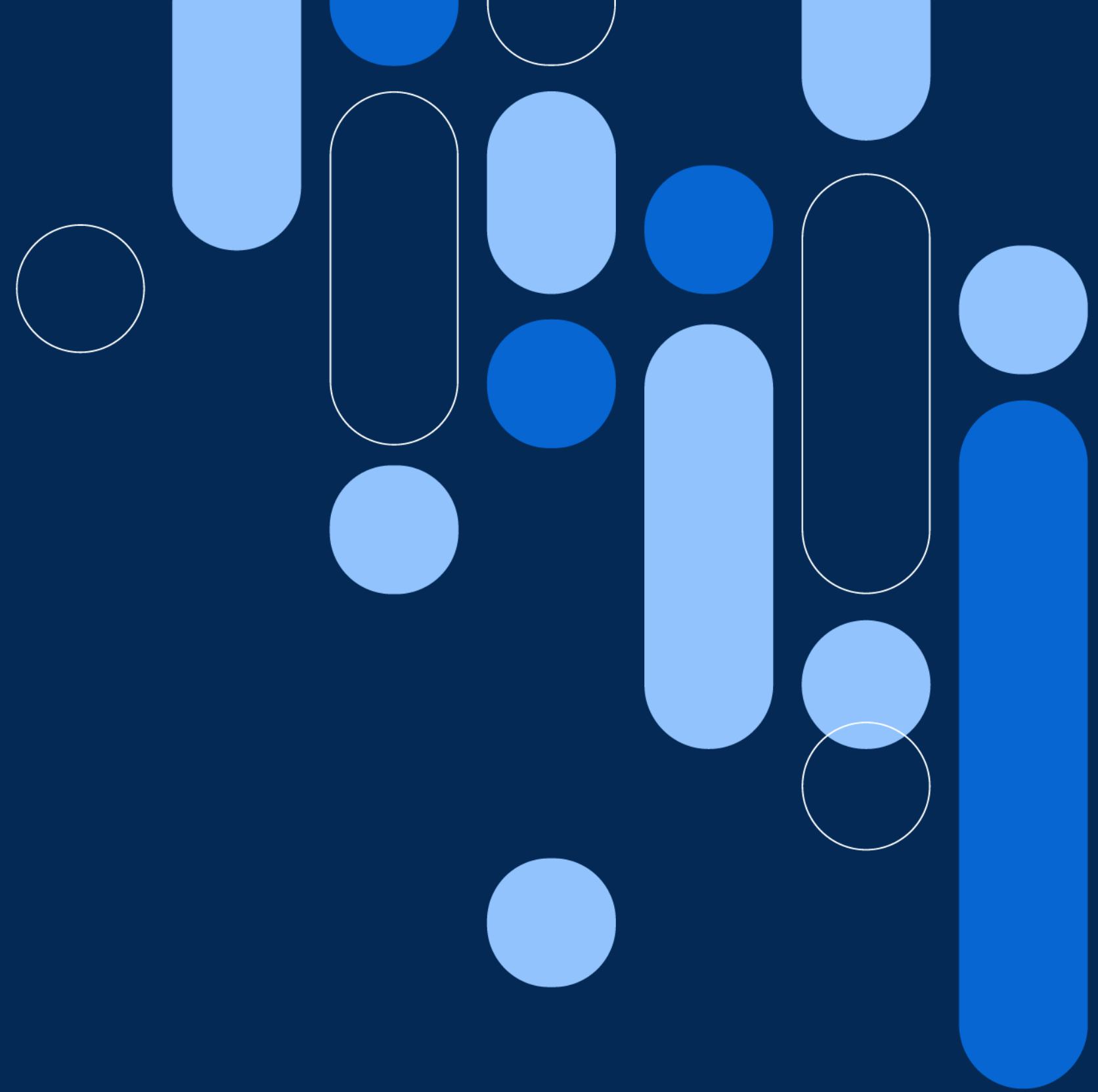
- Top Left:** A table listing variables and their data types for "Forest (Pipeline 2) (1.1)".
- Bottom Left:** A "General" tab showing model metadata.
- Center:** A "Versions" tab showing a table of model versions.
- Top Right:** Action buttons for "New Version" and "Set Version".

Names	Data Type	Input
Amount_avg	Decimal	✓
Balance_avg	Decimal	✓
Behavior_segment	Character	✓
Channel	Character	✓
Churn_num	Decimal	✓
Complaints_num	Decimal	✓

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: Futurum

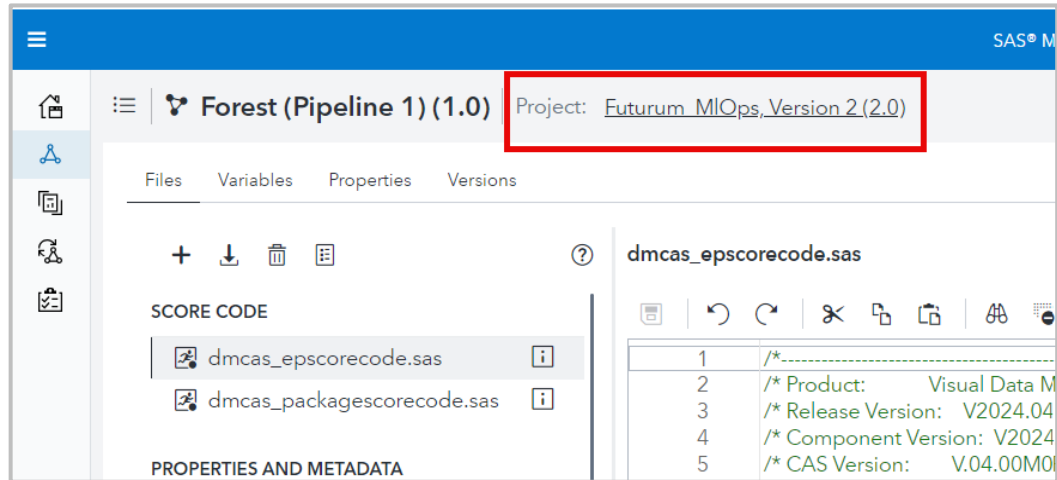
Versions	Displayed Ver...	Modified By	Published Status
2.0	✓	jordan.bakerman@sas.com	
1.1		jordan.bakerman@sas.com	Published
1.0		jordan.bakerman@sas.com	

Model Comparison Metrics and Graphs

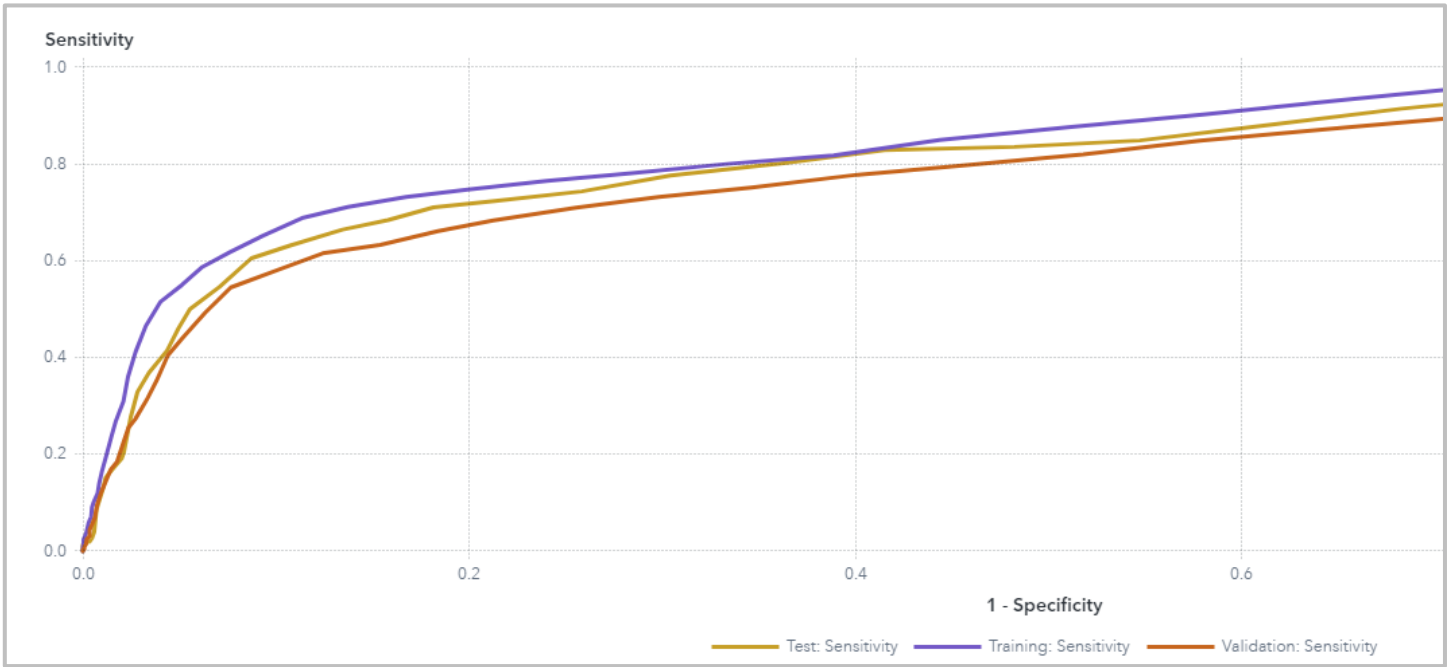


Model Comparison Metrics and Graphs

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.



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



Models	Forest (Pipeline 2) (1.1)
Model Properties	
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	

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 image consists of three screenshots from the SAS Viya interface. The first screenshot shows the 'Data Sources' tab with a list of tables. A red box highlights the 'Refresh' button. The second screenshot shows the 'New Test' dialog box with the following configuration: Name: Test 1, Description: Enter a description, Model: Forest (Pipeline 1), Version: 2.0 (champion), Input table: BANKING_NEW, and Output data library: CASUSER(spyridon.potamitis@sas...). The third screenshot shows the 'Scoring' tab with a table of test results.

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

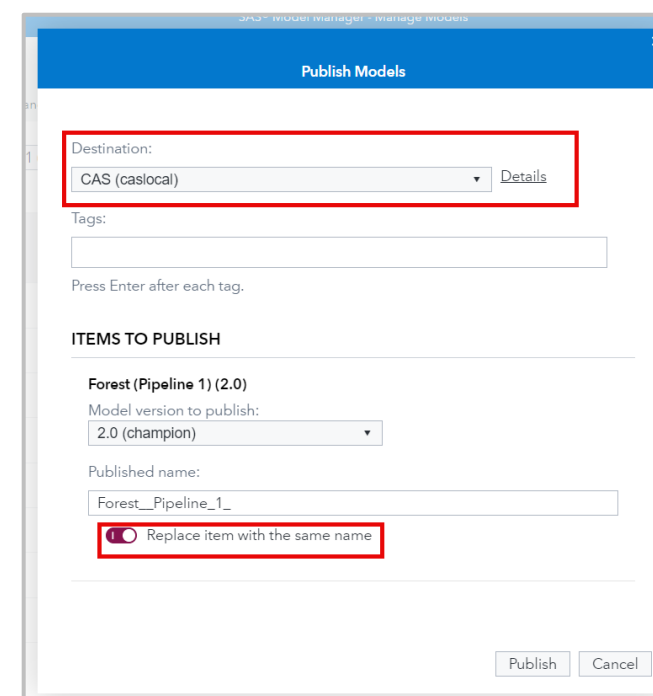
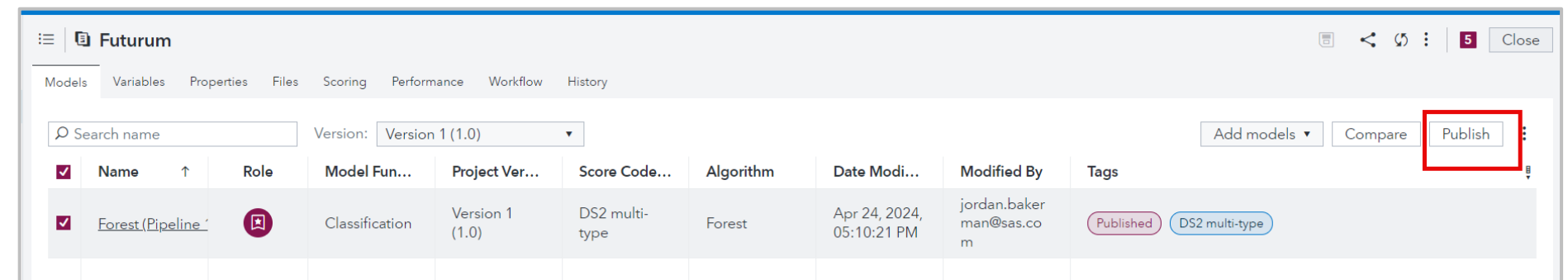
Churn	P_Churn1
1	0.1477686378
1	0.2666602709
0	0.1342149011
0	0.177621548
0	0.100215183
0	0.2206893598
0	0.2268064135
0	0.1029484225

One-Click Deployment – Multiple Deployment Destinations



One-Click Deployment

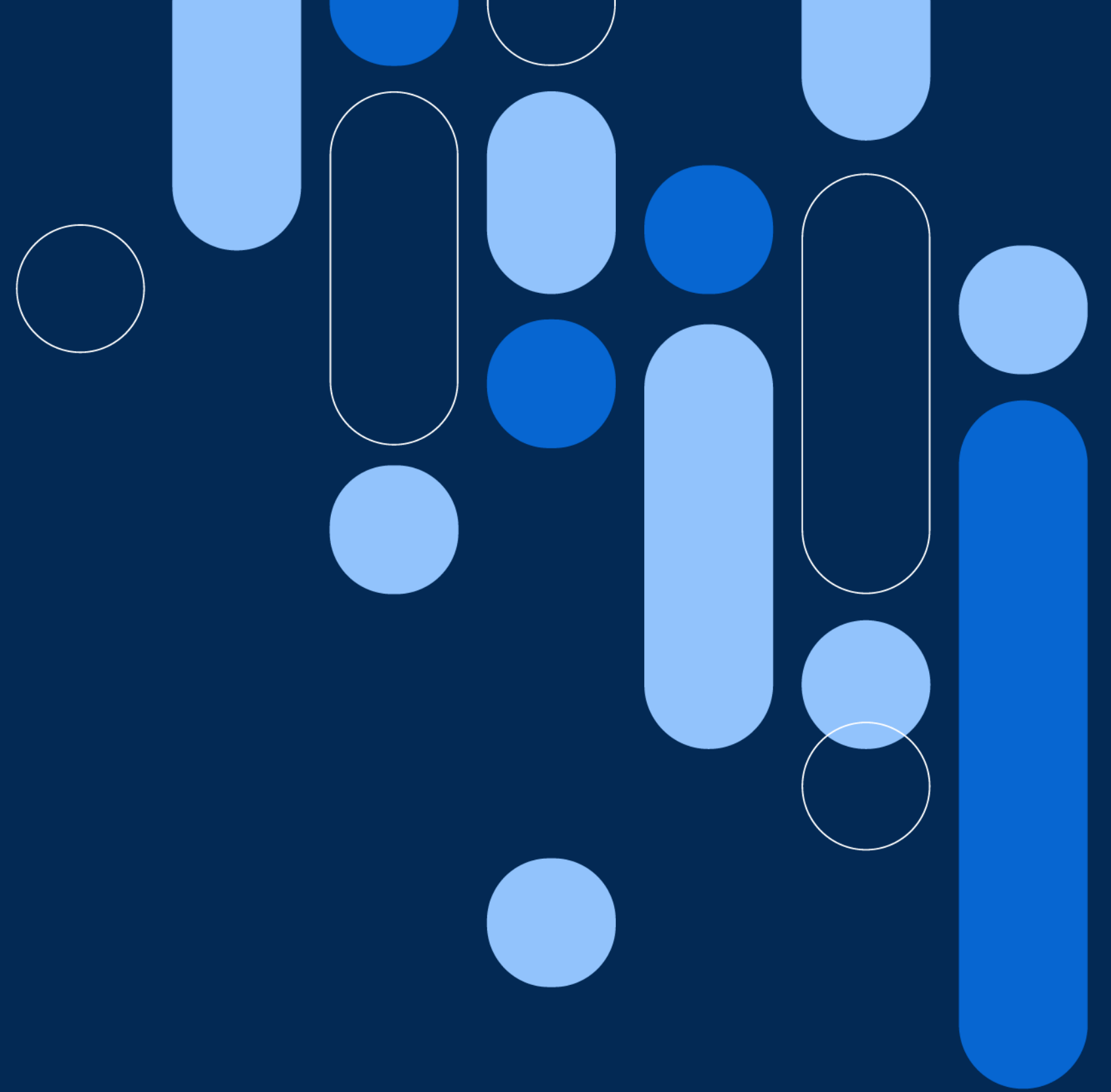
Go back to the main project page from the left of the screen and select the “Models” tab. Click the checkbox next to the champion model and then select “publish” on the top right of the project and choose “CAS (caslocal)” as the desired destination. (In our case we have configured CAS for batch scoring, 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.



Example with multiple publishing destinations

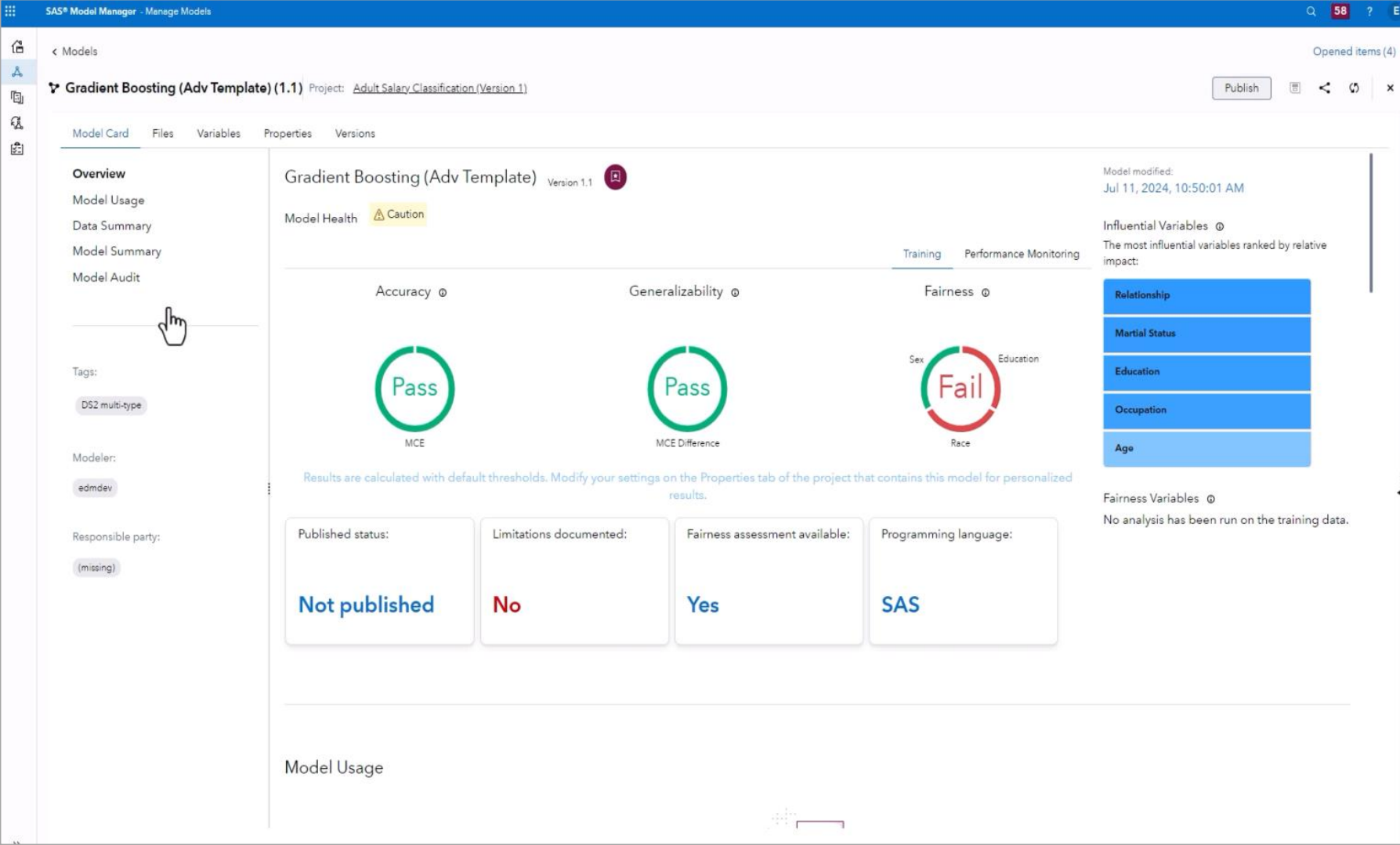
Publishing Results			
Name	Published Name	Status	Output
Forest (Pipeline 1)	Forest__Pipeline_1_	✔ Published successfully	Log

Model Cards



Review Model Health and Trustworthiness

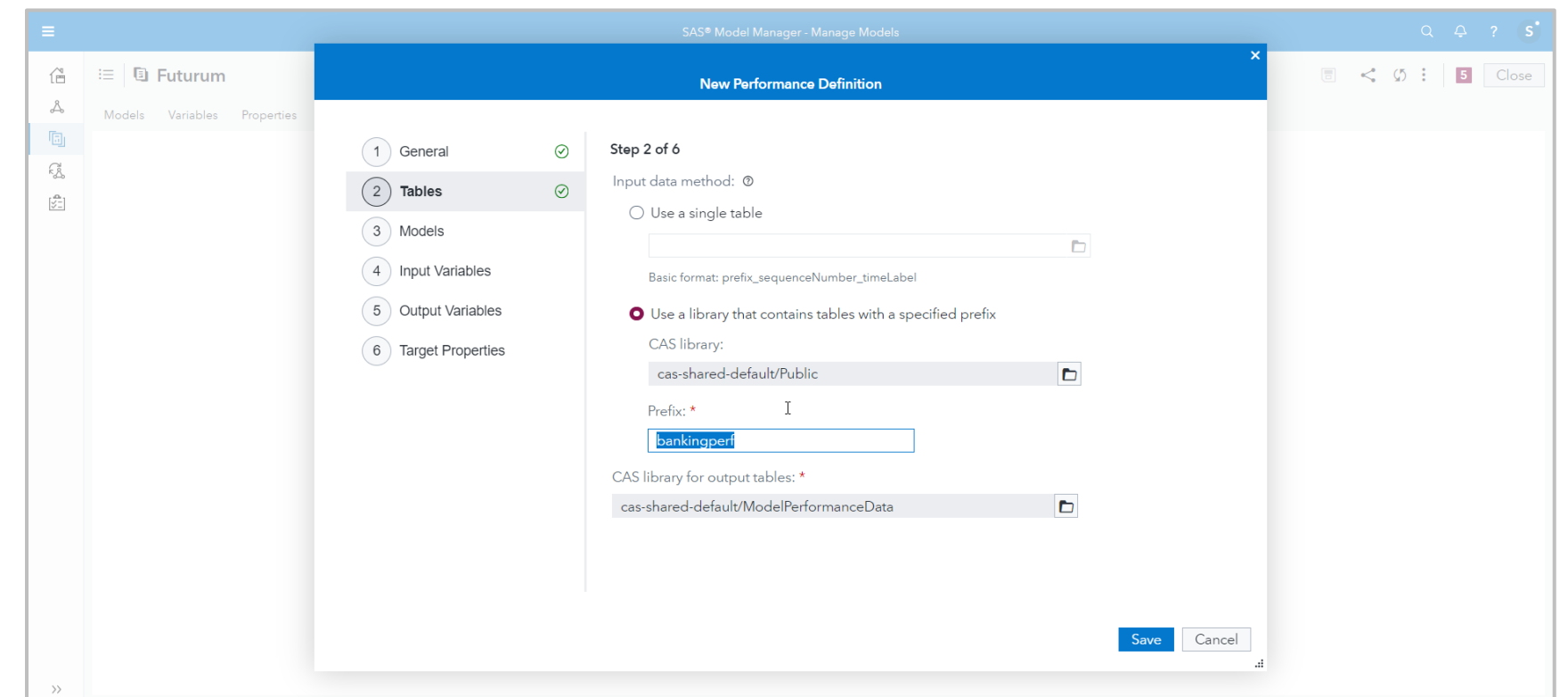
Once your model is registered in Model Manager, you can open the model card directly to review how the model performs and whether it can be trusted for use. The model card provides an at-a-glance view of key performance metrics, data characteristics, intended use and known limitations, along with indicators related to fairness, transparency and governance. Much of this information is automatically captured as part of the model lifecycle in Viya, making the model card a clear, shared view of model health and accountability. This built-in documentation is a core way SAS Viya supports trustworthy and responsible AI by helping you understand not just how a model performs, but how it should be used.



Model Performance Reports

Model Performance Reports

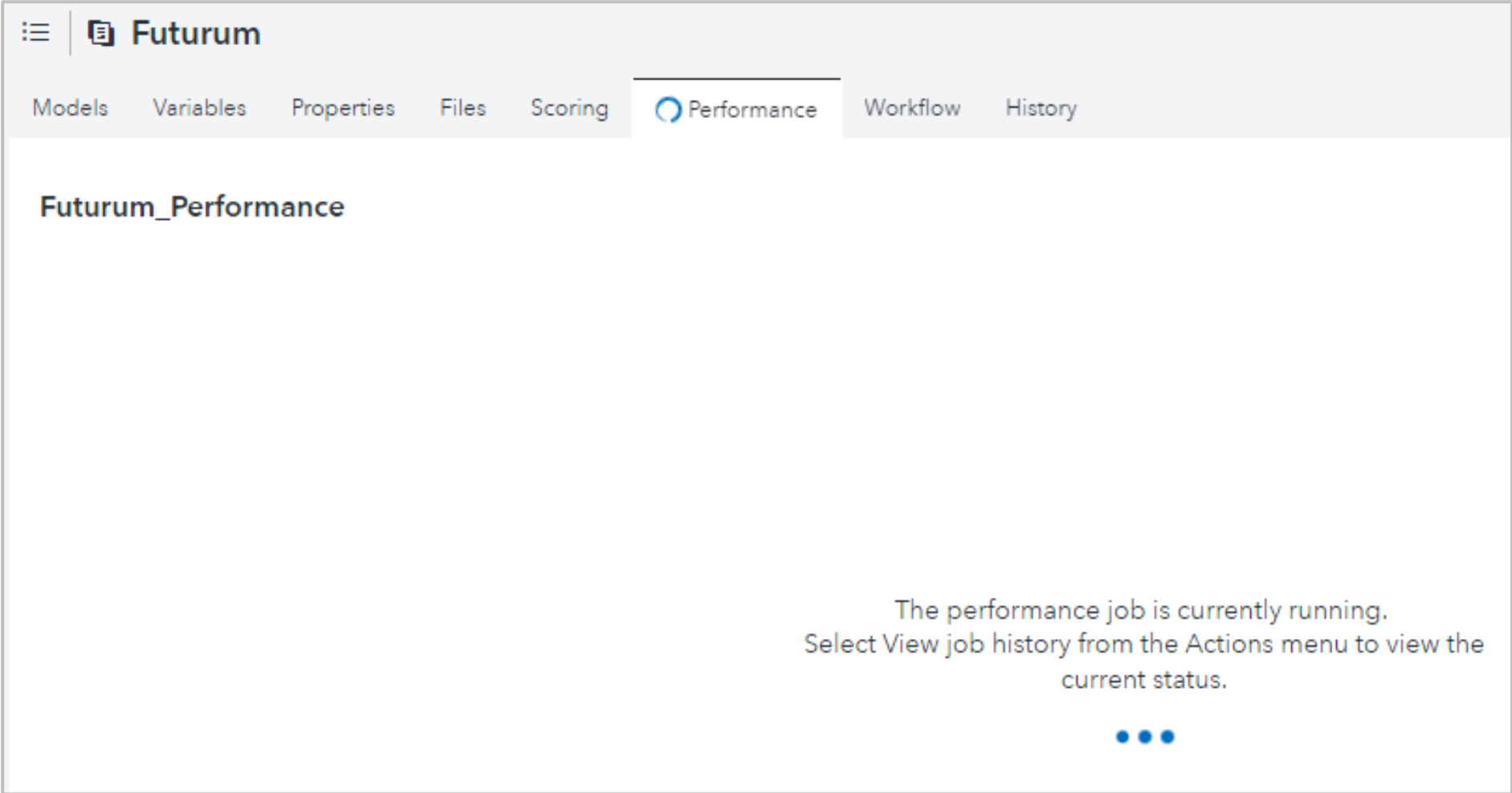
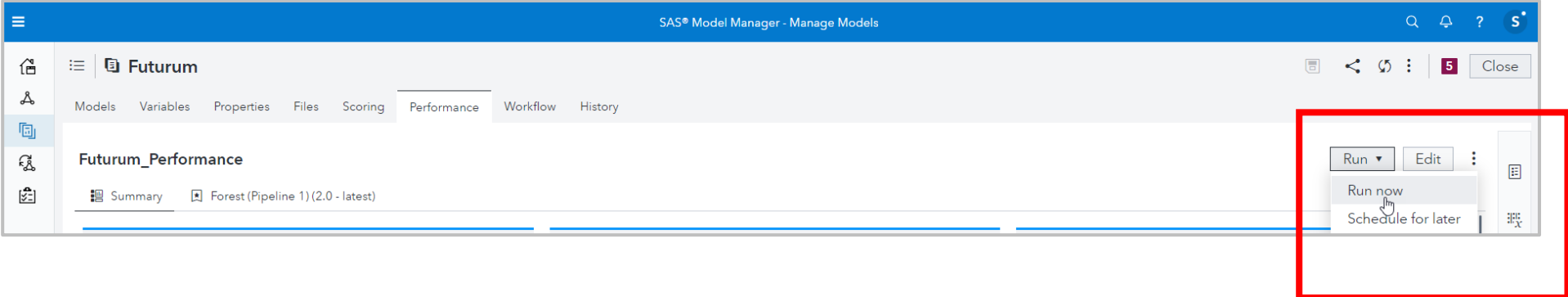
- Now that the model has been deployed, we must govern it and check for model degradation. Here, we assume new data has been collected over the following four quarters from deployment and has been placed in four new data sets (BANKINGPERF_1_Q1, BANKINGPERF_2_Q2, BANKINGPERF_3_Q3, BANKINGPERF_4_Q4). These 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.



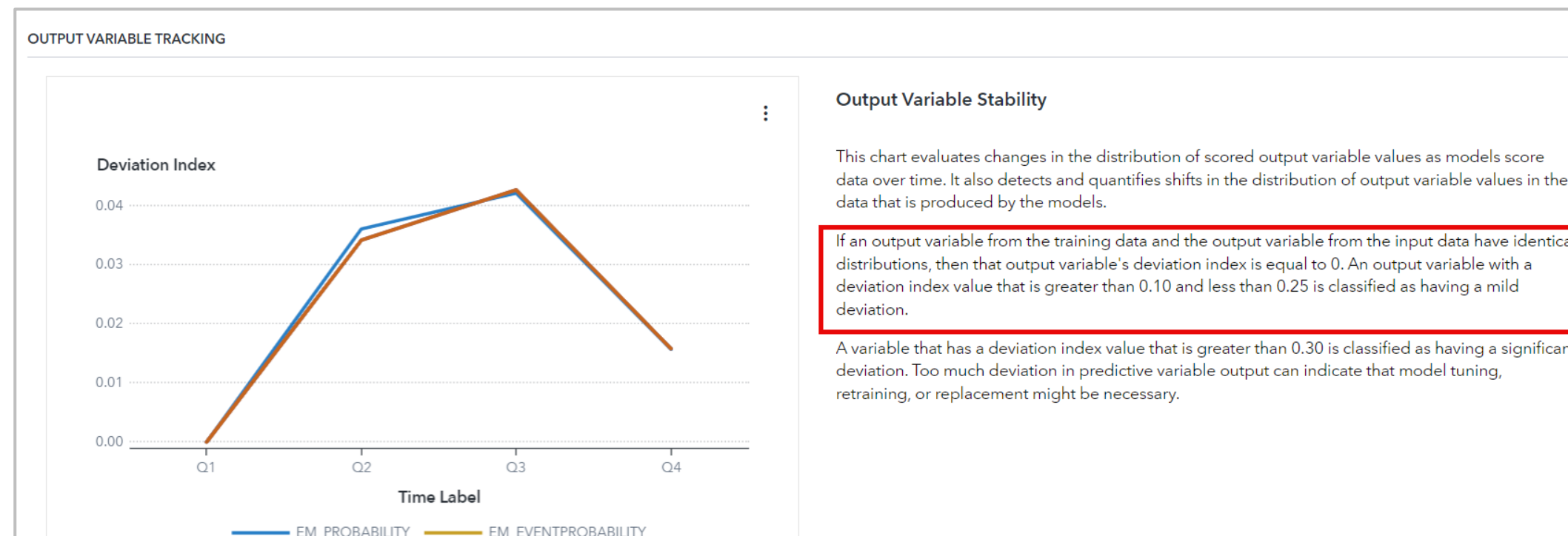
Model Performance Reports

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 shows the SAS Viya Futurum Performance Report interface. At the top, there is a navigation bar with tabs for Models, Variables, Properties, Files, Scoring, Performance (selected), Workflow, and History. Below the navigation bar, the main content area is titled 'Futurum_Performance' and has sub-tabs for Summary and Forest (Pipeline 2) (1.1 - latest). The main content is divided into three columns: 'Scored Observations' showing 4,000, 'Definition Details' showing metadata like 'Created by: Jordan.Bakerman@sas.com' and 'Last updated: Apr 11, 2024, 02:59:23 PM', and 'Model Performance Rank' showing rank 1 for 'Forest (Pipeline 2)'. Below these columns, there is a detailed view for 'Forest (Pipeline 2)' with a misclassification rate of 21.00%, a false positive rate of 9.49%, and a recall of 57.14%. At the bottom of this detailed view, there is a metadata section with 'Last updated: 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'.

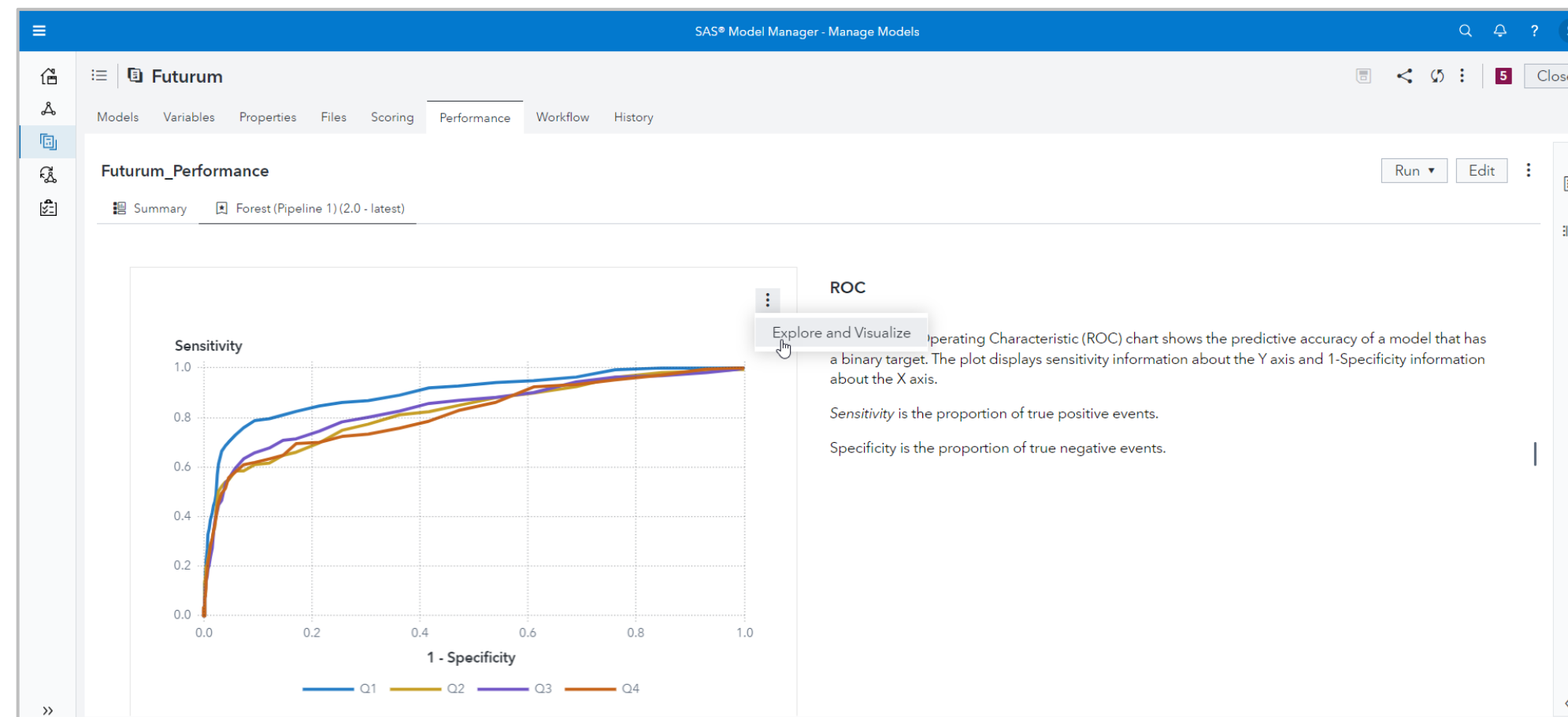
Model Performance Reports

- 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.

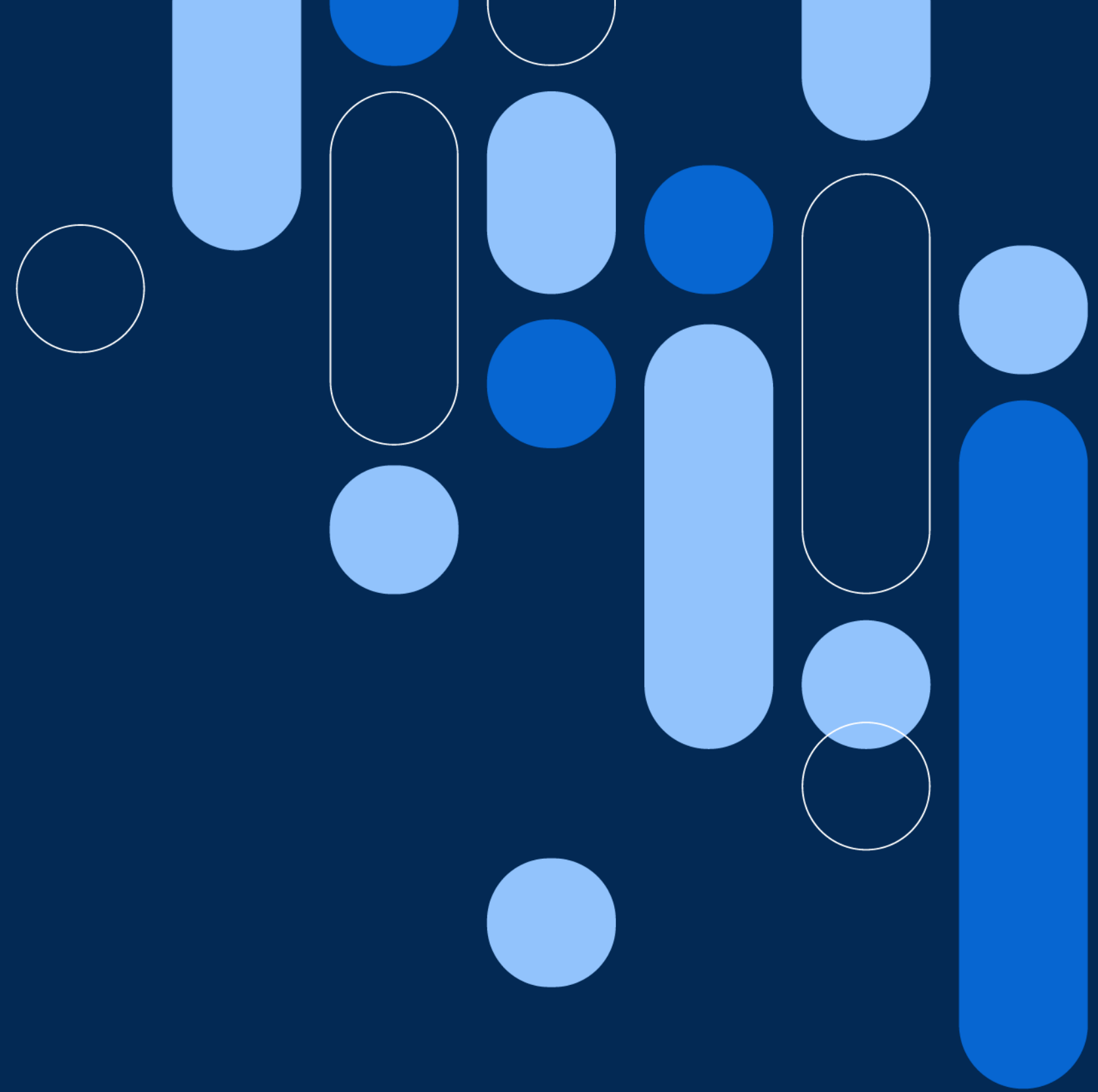


Model Performance Reports

- 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.

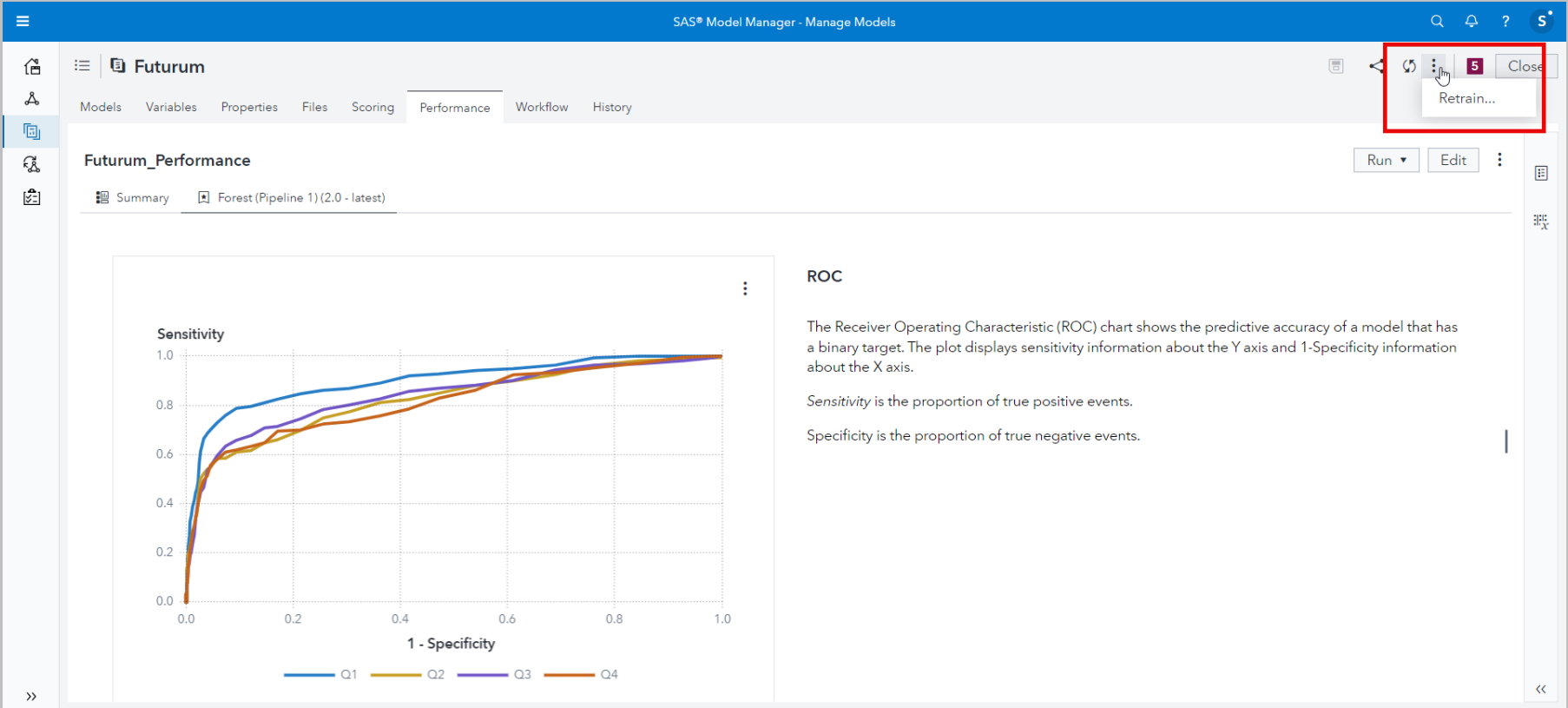


Retraining



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.

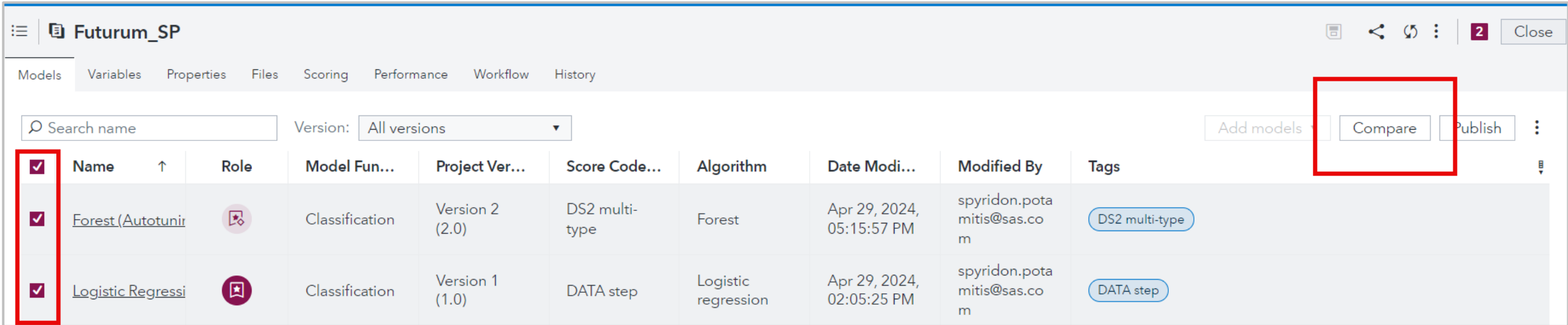
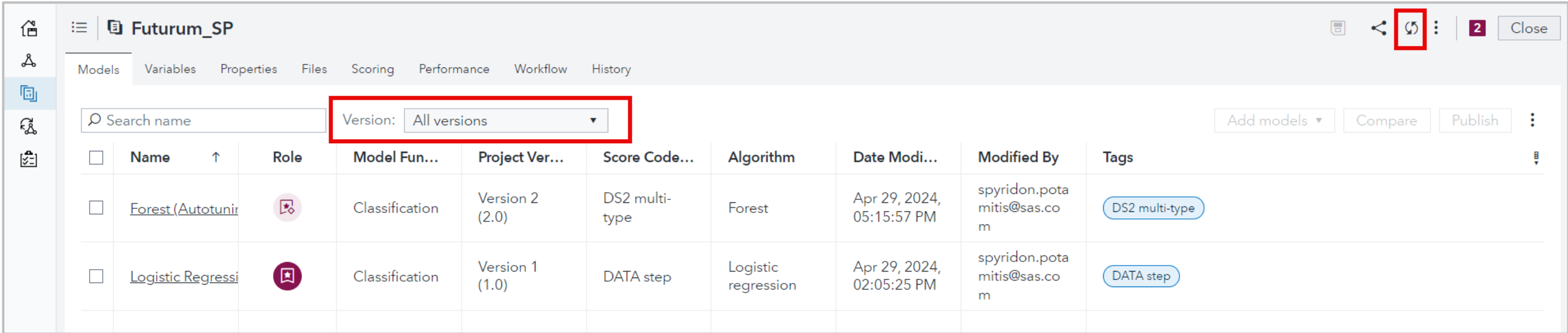


The screenshot shows a dialog box titled 'Retrain Project'. The dialog contains the following text and options:

- Header: Retrain Project
- Instruction: Select an option and send a request to the Model Studio application for retraining.
- Options:
 - Set the project retrain state to needed
 - Retrain now with a new data table
- Data Table Selection: A text field containing 'cas-shared-default/Public/BANKING_RETRAIN' with a folder icon to its right.
- Buttons: 'Send' and 'Cancel' buttons at the bottom right.

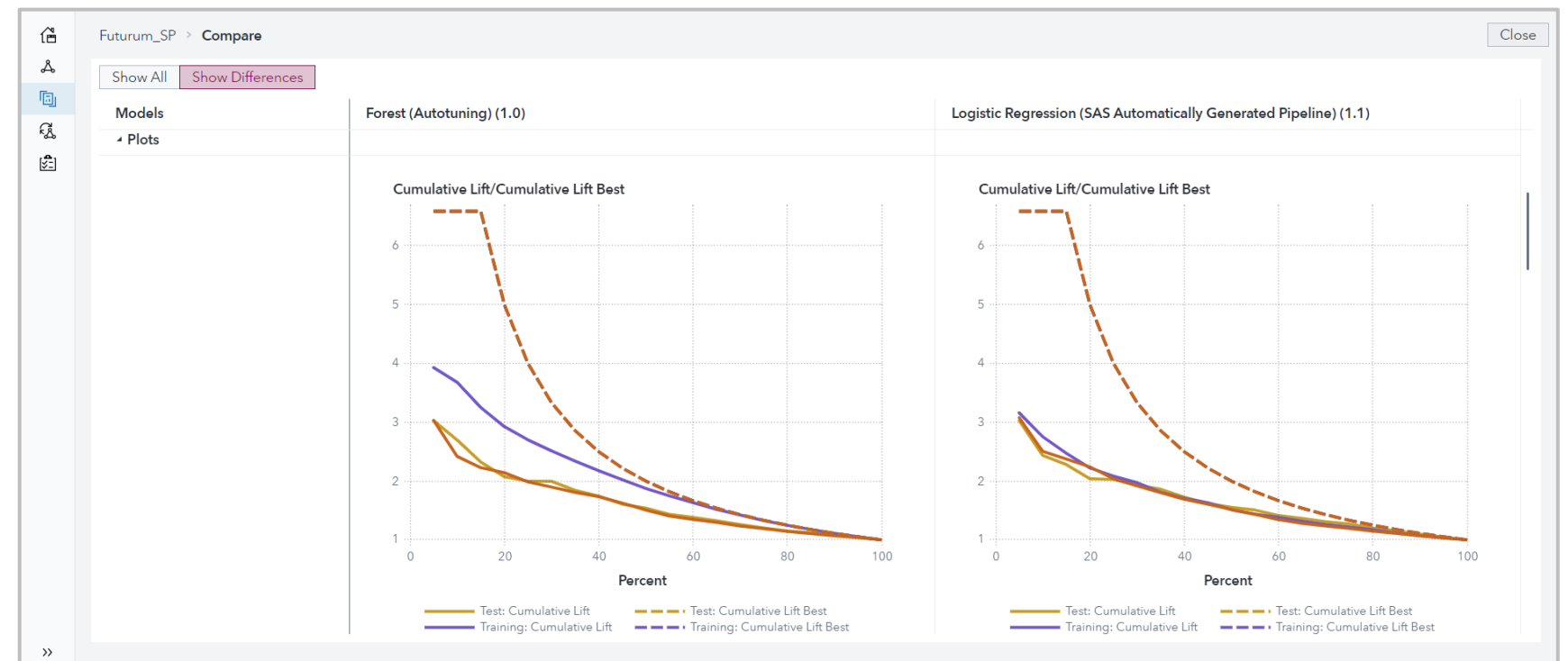
Retraining

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.”



Retraining

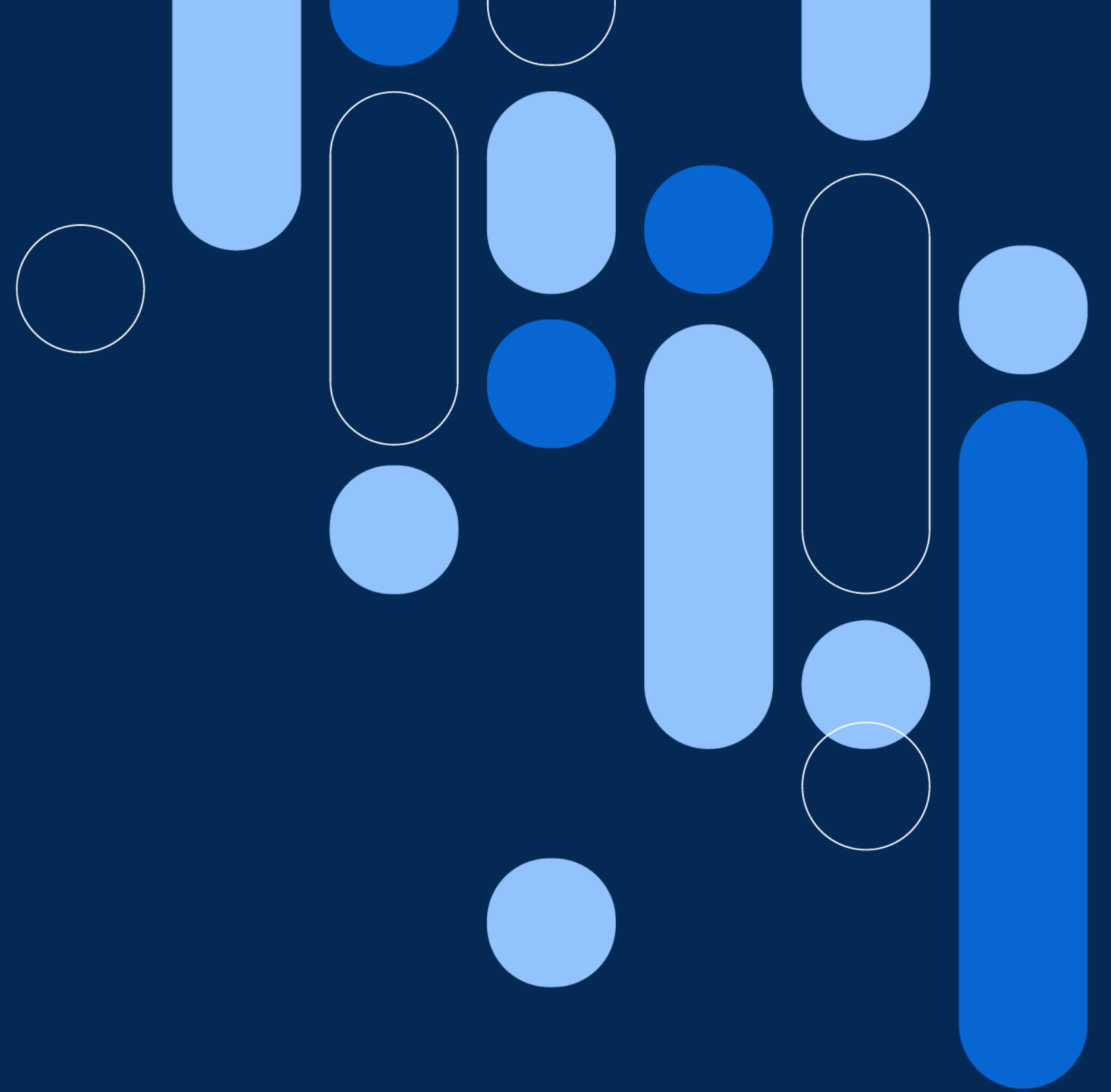
- Automatically, you will get a comparison of the two models. The older one and the new one. Click “Show differences” and investigate the results and the plots generated.
- After you compare the models, you can go back to the “Models” tab. Click on the new version of the model you created, and then click 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 for Name, Role, Model Function, Project Version, Score Code, Algorithm, Date Modified, Modified By, and Tags. 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.

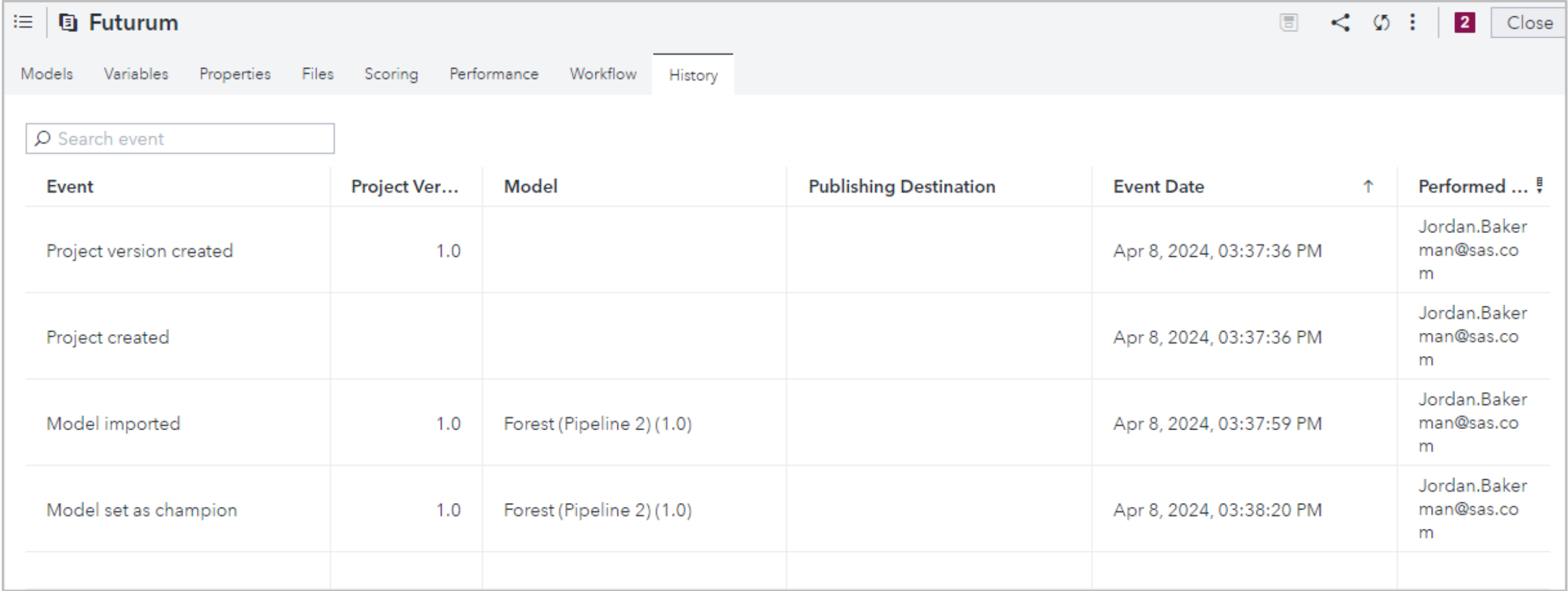
Name	Role	Model Fun...	Project Ver...	Score Code...	Algorithm	Date Modi...	Modified By	Tags
Forest (Autotuning)	Challenger	Classification	Version 2 (2.0)	DS2 multi-type	Forest	Apr 29, 2024, 05:15:57 PM	spyridon.potamitis@sas.com	DS2 multi-type
Logistic Regression	Champion	Classification	Version 1 (1.0)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



The screenshot shows the 'History' tab in the SAS Viya Futurum interface. The table lists the following events:

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, and the 'Model Evaluation' sub-tab is active. The 'Model Evaluation' section includes fields for 'Default training table' (cas-shared-default/Public/BANKING_NEW), 'Target variable' (Churn), 'Target level' (Binary), 'Target event value' (1), and 'Output event probability variable' (P_Churn1). Below this is the 'MODEL ASSESSMENT CRITERIA' table, which is highlighted with a red box. The table has the following structure:

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

Business Analyst

Designs, Test, Deploys and Governs Decision Flows



 sas viya[®]

Tasks

Business Analyst, Decision Scientist

1. Design decision flows combining models, rules, and data.
2. Version decisions and decision objects as reusable assets.
3. Test and validate decisions before deployment.
4. Deploy decisions to real-time and batch execution endpoints.
5. Monitor decision outcomes and business performance.
6. Manage change with approval workflows and audit history.

Key Capabilities

Product

1. Visual decision flow authoring (no-code/low-code)
2. Centralized decision repository with versioning
3. Decision testing and simulation
4. Integrated approval and governance workflows
5. Monitoring of decision performance and outcomes
6. Full auditability and traceability across the decision life cycle

Design and Author Decision Flows

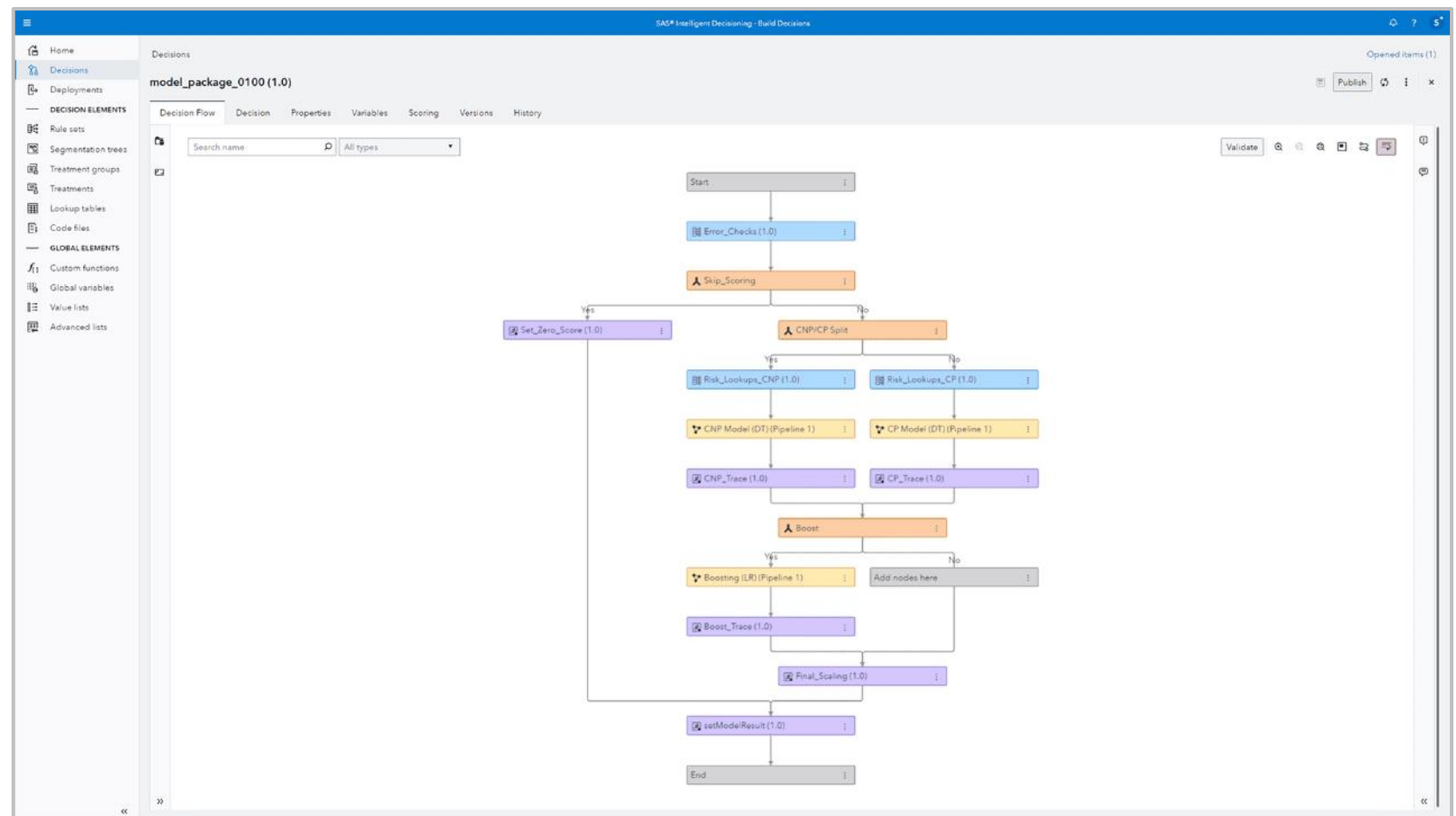
Outcome: A decision flow that consistently applies business logic and analytics to real-world scenarios

What you do

Design decision flows that operationalize analytics by combining models, rules and data into a single, executable decision.

Steps

1. Create a new decision flow using a visual, no-code/low-code canvas.
2. Add decision logic, including rules, model invocations, branches and data mappings.
3. Define decision outcomes and business actions.
4. Validate the flow structure and logic before testing.



Version Decision and Decision Assets (Optional)

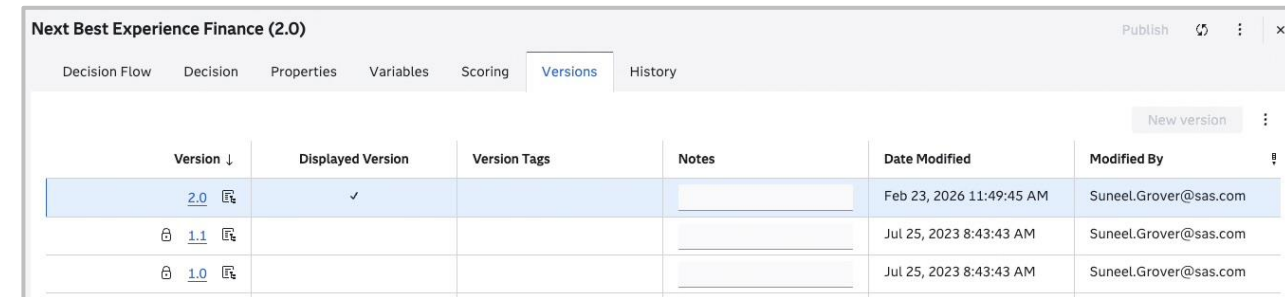
Outcome: A governed decision asset with full version history and traceability

What you do

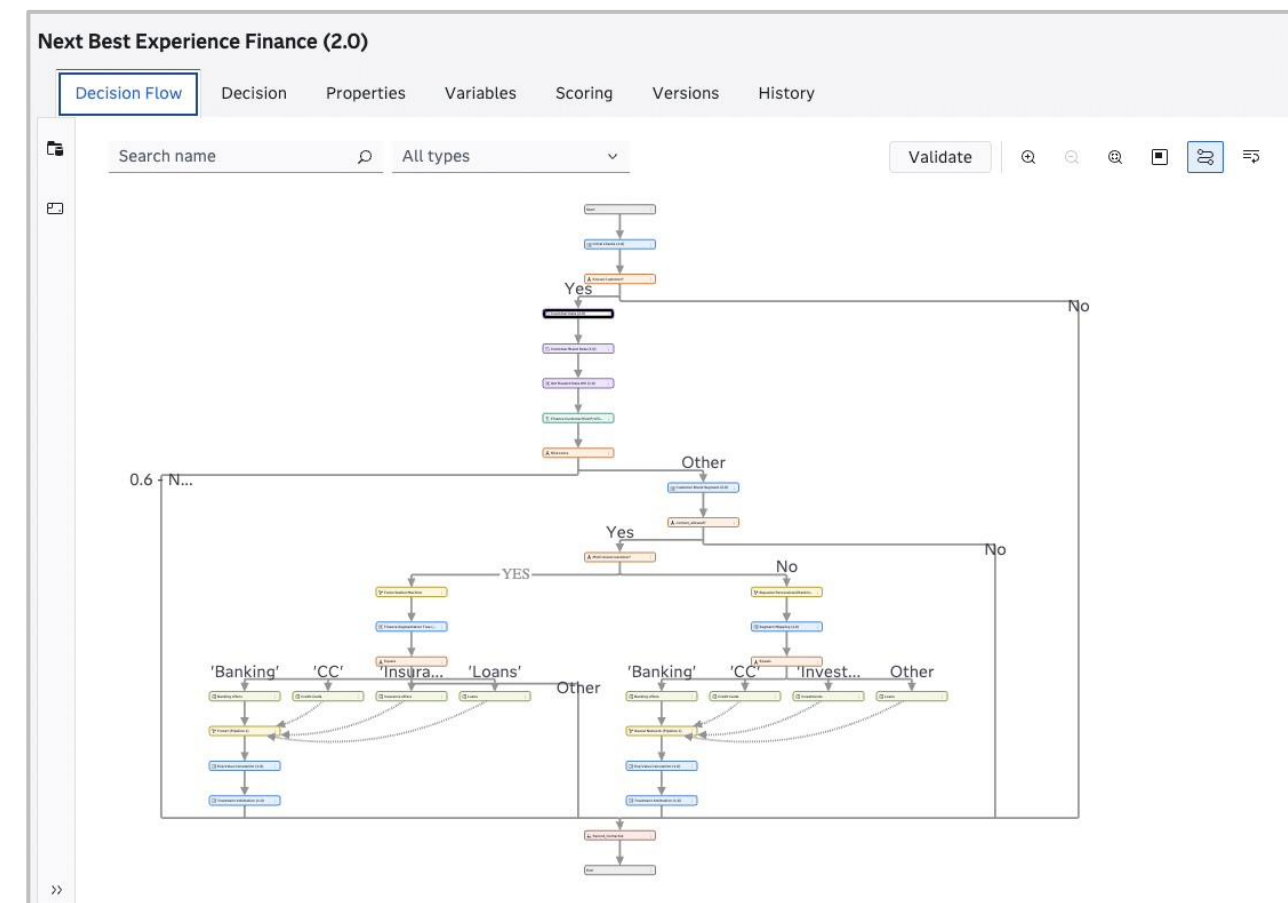
Validate that decisions produce the right outcomes using test data before deployment.

Steps

1. Create a new decision flow version in the repository for existing decision.
2. Capture metadata such as descriptions and ownership.
3. Create new versions as changes are introduced.
4. Retain access to prior versions for audit and rollback.
5. Check decision version and decision components lineage, with the Used By Report capability.



Version	Displayed Version	Version Tags	Notes	Date Modified	Modified By
2.0	✓			Feb 23, 2026 11:49:45 AM	Suneel.Grover@sas.com
1.1				Jul 25, 2023 8:43:43 AM	Suneel.Grover@sas.com
1.0				Jul 25, 2023 8:43:43 AM	Suneel.Grover@sas.com



Test and Simulate Decisions

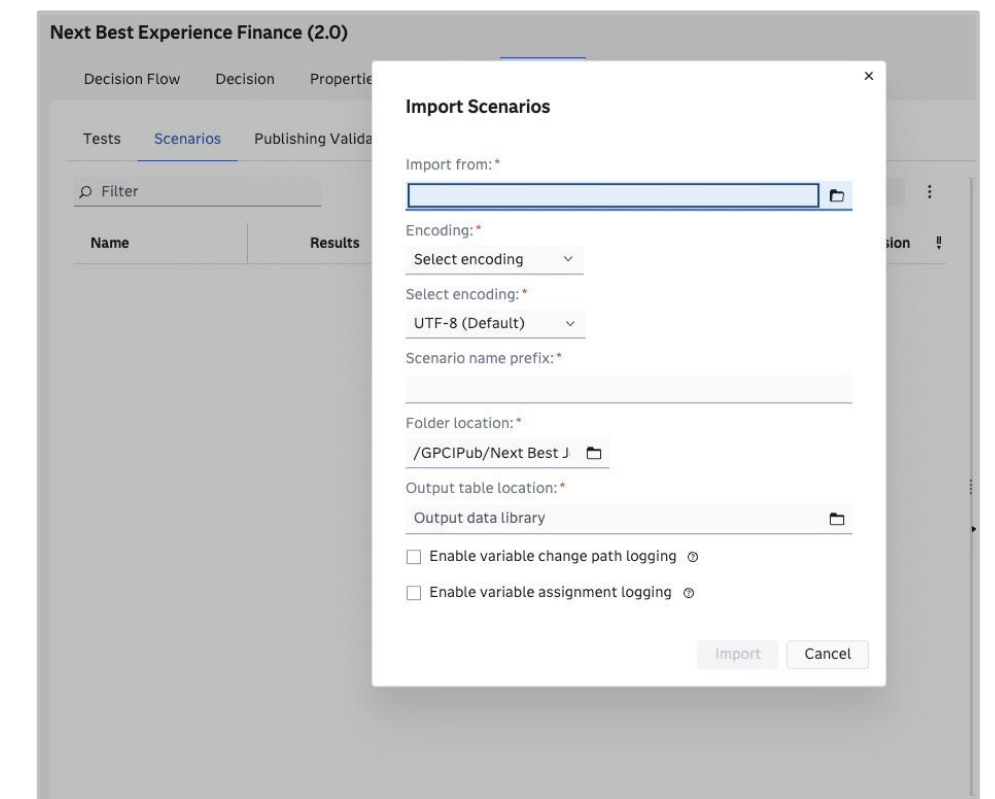
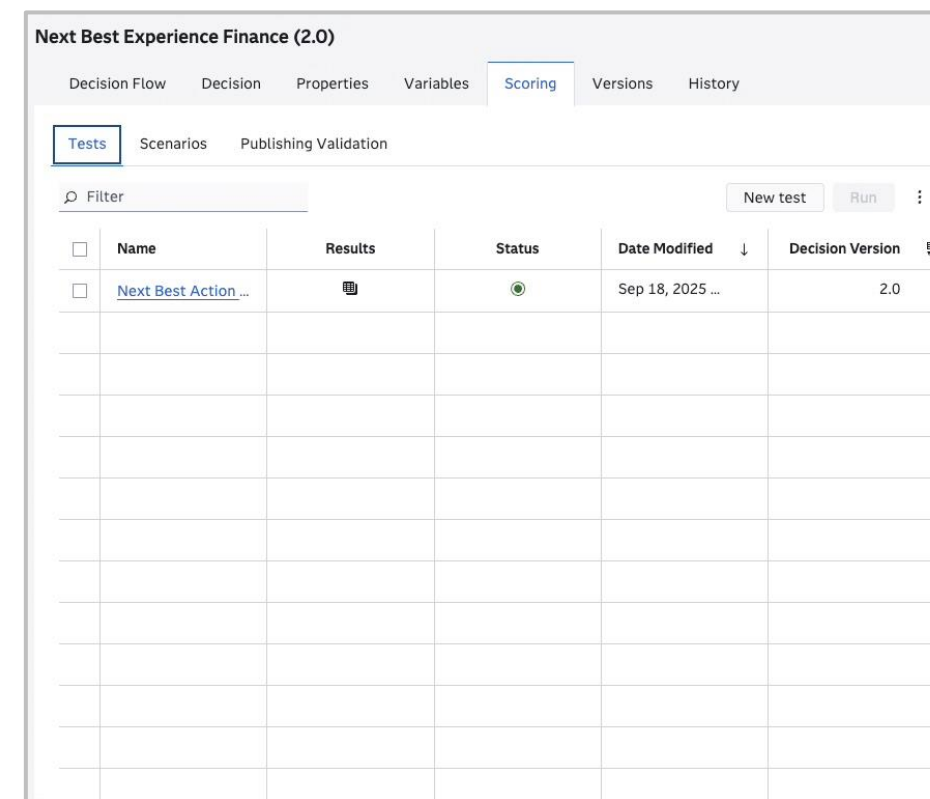
Outcome: Confidence that decisions behave as expected before being deployed to production

What you do

Validate that decisions produce the right outcomes using test data before deployment.

Steps

1. Run test cases using scenario definitions, sample or historical data.
2. Review decision paths, rules fired and outputs.
3. Identify and adjust logic or thresholds as needed.
4. Confirm decision readiness for deployment.



Approve and Govern Decisions

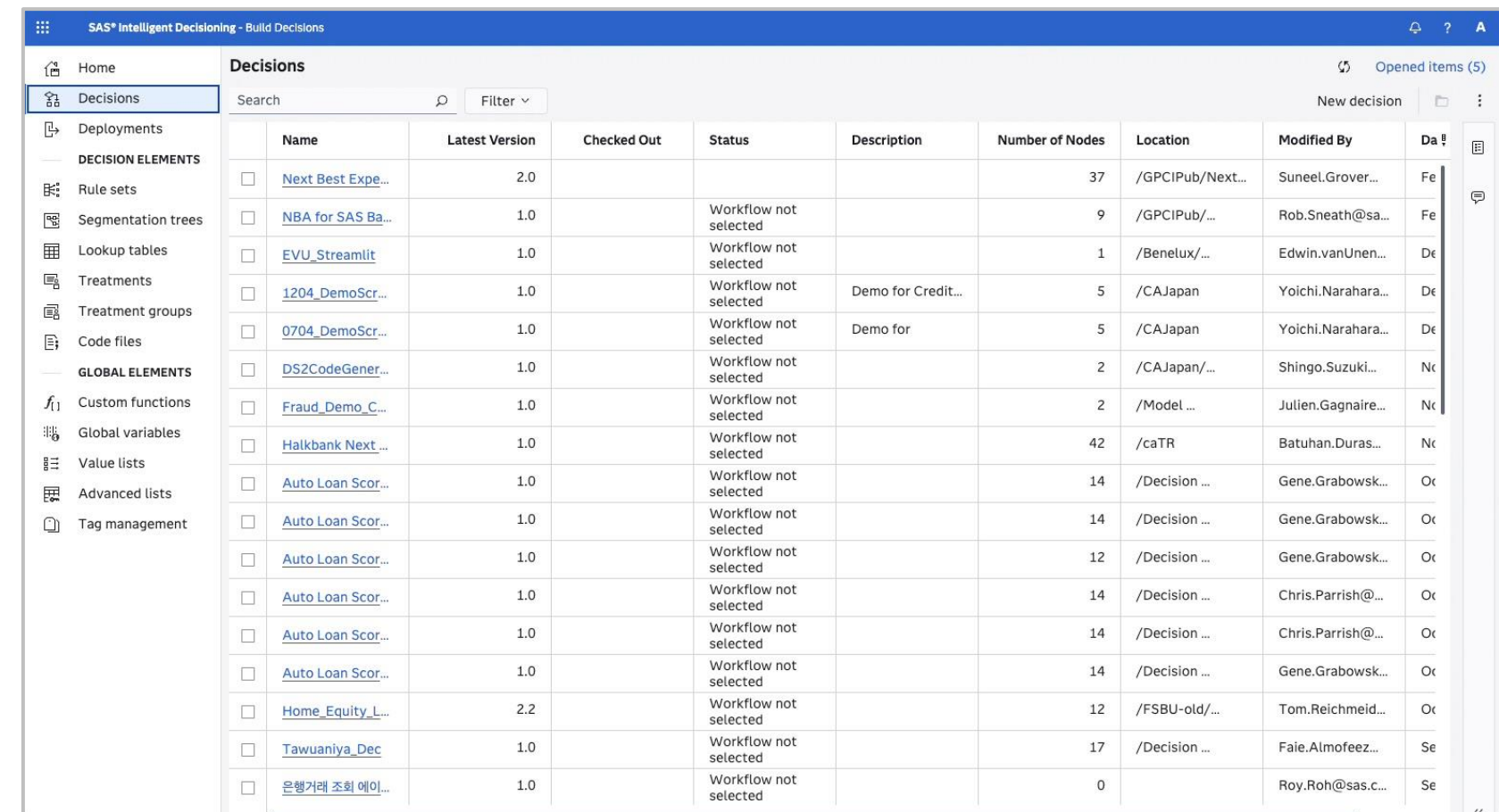
Outcome: Decisions that meet business, regulatory and organizational standards

What you do

Ensure decisions are reviewed, approved and auditable before production use.

Steps

1. Assign default workflow to a decision
2. Submit decisions for review and approval.
3. Collaborate with reviewers using annotations and comments.
4. Track approval status and decision history.
5. Ensure decisions are explainable and compliant.



The screenshot shows the SAS Intelligent Decisioning - Build Decisions interface. The main area displays a table of decisions with columns for Name, Latest Version, Checked Out, Status, Description, Number of Nodes, Location, Modified By, and Date. The table lists various decision items, many of which are marked as 'Workflow not selected'.

Name	Latest Version	Checked Out	Status	Description	Number of Nodes	Location	Modified By	Date
<input type="checkbox"/> Next Best Expe...	2.0				37	/GPCIPub/Next...	Suneel.Grover...	Fe
<input type="checkbox"/> NBA for SAS Ba...	1.0		Workflow not selected		9	/GPCIPub/...	Rob.Sneath@sa...	Fe
<input type="checkbox"/> EVU_Streamlit	1.0		Workflow not selected		1	/Benelux/...	Edwin.vanUnen...	De
<input type="checkbox"/> 1204_DemoScr...	1.0		Workflow not selected	Demo for Credit...	5	/CAJapan	Yoichi.Narahara...	De
<input type="checkbox"/> 0704_DemoScr...	1.0		Workflow not selected	Demo for	5	/CAJapan	Yoichi.Narahara...	De
<input type="checkbox"/> DS2CodeGener...	1.0		Workflow not selected		2	/CAJapan/...	Shingo.Suzuki...	Nc
<input type="checkbox"/> Fraud_Demo_C...	1.0		Workflow not selected		2	/Model ...	Julien.Gagnaire...	Nc
<input type="checkbox"/> Halkbank Next ...	1.0		Workflow not selected		42	/caTR	Batuhan.Duras...	Nc
<input type="checkbox"/> Auto Loan Scor...	1.0		Workflow not selected		14	/Decision ...	Gene.Grabowsk...	Ox
<input type="checkbox"/> Auto Loan Scor...	1.0		Workflow not selected		14	/Decision ...	Gene.Grabowsk...	Ox
<input type="checkbox"/> Auto Loan Scor...	1.0		Workflow not selected		12	/Decision ...	Gene.Grabowsk...	Ox
<input type="checkbox"/> Auto Loan Scor...	1.0		Workflow not selected		14	/Decision ...	Chris.Parrish@...	Ox
<input type="checkbox"/> Auto Loan Scor...	1.0		Workflow not selected		14	/Decision ...	Chris.Parrish@...	Ox
<input type="checkbox"/> Auto Loan Scor...	1.0		Workflow not selected		14	/Decision ...	Gene.Grabowsk...	Ox
<input type="checkbox"/> Home_Equity_L...	2.2		Workflow not selected		12	/FSBU-old/...	Tom.Reichmeid...	Ox
<input type="checkbox"/> Tawuniya_Dec	1.0		Workflow not selected		17	/Decision ...	Faie.Almofeez...	Se
<input type="checkbox"/> 은행거래 조회 예이...	1.0		Workflow not selected		0		Roy.Roh@sas.c...	Se

Select Key Performance Indicators (Optional)

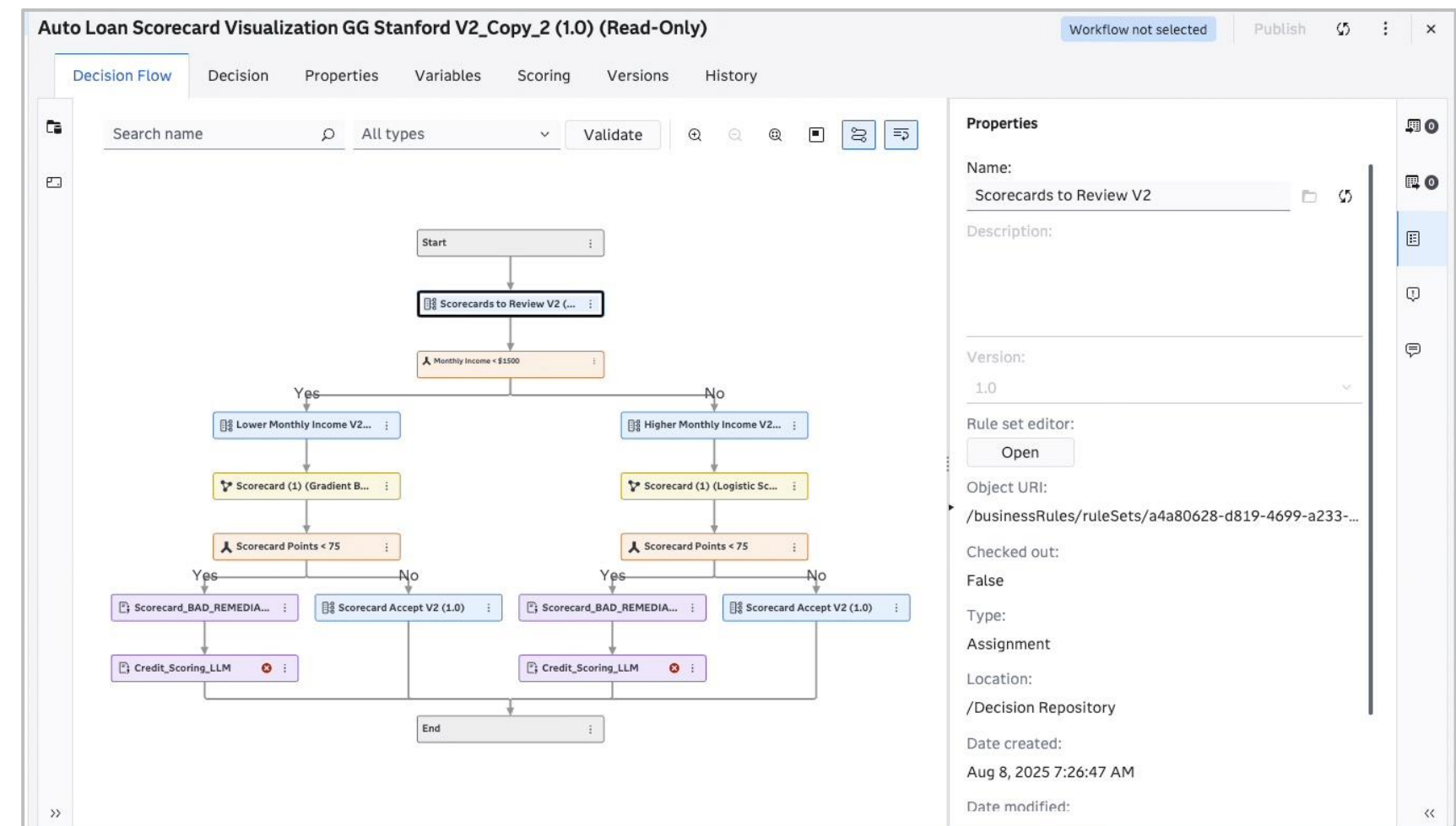
Outcome: Visibility into how decisions perform and impact business outcome

What you do

Ensure key metrics are saved for decision impact and performance tracking.

Steps

1. Add a Record Contact Node to the decision flow.
2. Select key metrics (variables) to be tracked at runtime.
3. Read about querying the Subject Contact Database.



Deploy Decisions to Production

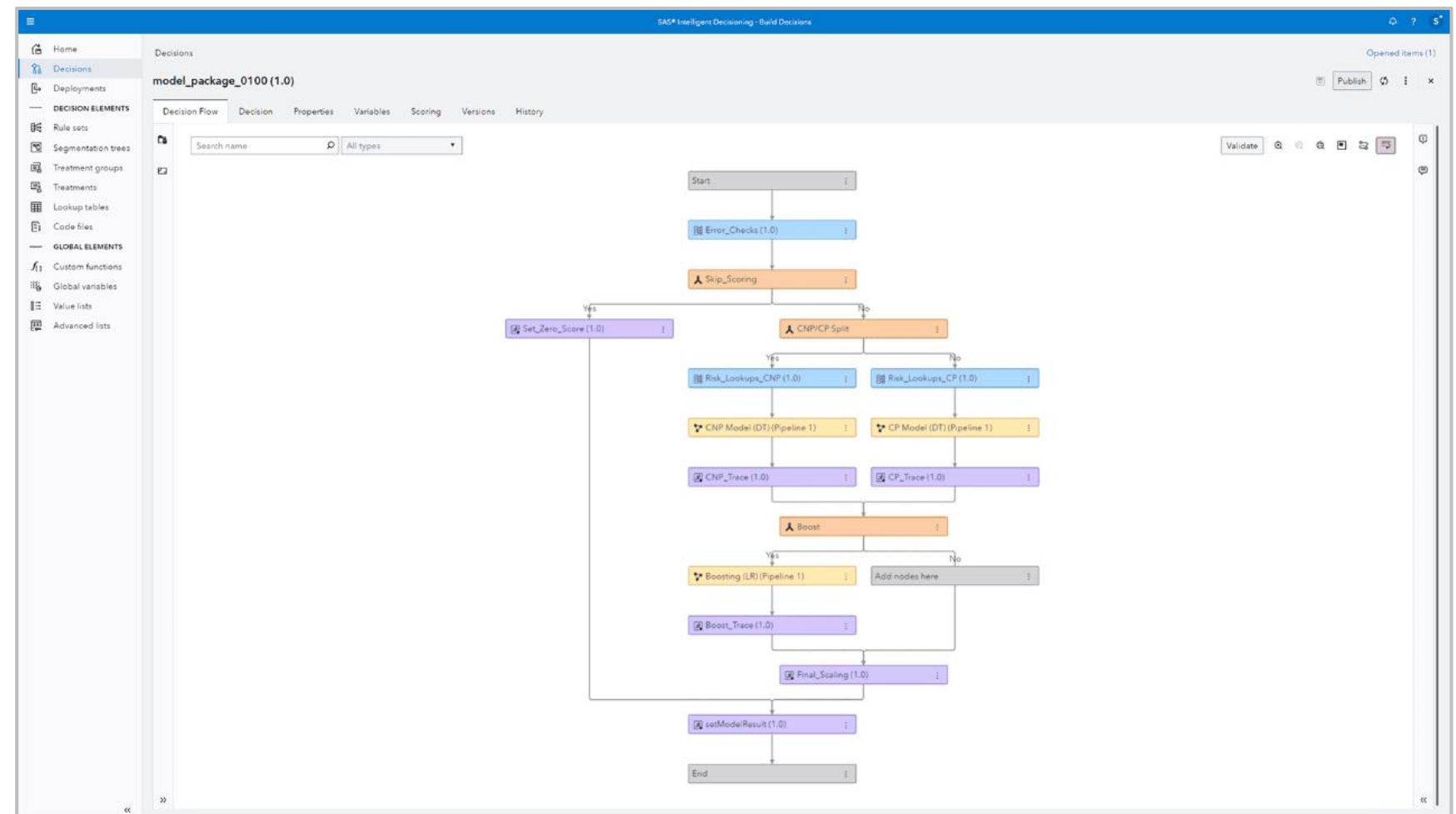
Outcome: A live decision that can be executed consistently at scale

What you do

Operationalize decisions so they can be executed in real-time or batch processes.

Steps

1. Select the target execution environment also known as Publish Destination.
2. Deploy the approved decision flow.
3. Verify successful deployment.
4. Make the decision available to applications or processes.
5. Track and manage published decisions by destination.



Thank you!

