SAS® Model Manager 15.1: User's Guide

SAS® Documentation
December 21, 2017

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2017. *SAS® Model Manager 15.1: User's Guide*. Cary, NC: SAS Institute Inc.

SAS® Model Manager 15.1: User's Guide

SAS Institute Inc., SAS Campus Drive, Cary, NC 27513-2414

December 2017

15.1-P1:mdlmgrug

Contents

1

Introduction to SAS Model Manager

About Managing Models

Using SAS Model Manager, you can store models in a common model repository, and organize them within projects and folders. You can also evaluate models for champion model selection, monitor performance of models, and publish models. All model development and model maintenance personnel, including data modelers, validation testers, scoring officers, and analysts, can use SAS Model Manager.

Here are some of the services SAS Model Manager provides:

- Use a single interface to access all of your business modeling projects. All models are stored in a common model repository. Models can also be accessed in one place using the **Models** category.
- Import models that you develop using a SAS application, such as Model Studio, as well as SAS code, PMML models, and R models. You can also create a new model with the model's files in a folder or project.
- Compare models to assess candidate models.
- Run tests to validate models for scoring.
- Publish models to CAS, Hadoop, SAS Micro Analytic Service, and Teradata for scoring by external applications or interfaces.
- Create custom workflow definitions to meet your business requirements and to match your business processes. You can then start a workflow process to track the progress of your project.

Data sources are an integral part of the modeling process. Data sources are used for scoring, testing, and performance monitoring. Performance data can be created from your operational data, provided that it has the required structure (for example, the data contains a target variable). For information about preparing your data, see *Getting Started with SAS Data Preparation for SAS Viya*.

Sign In

To sign in to SAS Model Manager:

1. In the address bar of your web browser, enter the URL for SAS Home (for example, http://host_name/SASHome) and press **Enter**. The **Sign In** page appears.

   Note: If you are in a single sign-on environment, you are not prompted to sign in.
2. Enter your user ID and password.

3 Click **Sign In**. The SAS Home **Welcome** page appears.
4 Click **Manage Models**.

Note: Alternatively, you can use the direct default URL http://host_name/SASModelManager. Contact your system administrator if you need the URL for SAS Home or SAS Model Manager.

Manage Application Settings

You can use the Settings window to edit user preferences or customize accessibility settings for all SAS web applications. In addition, you can manage model repositories from within the Settings window.

To access the Settings window, click your name in the application bar and select **Settings**.

For information about settings, see the following documentation:

- "Settings" in General Usage Help for SAS Viya Web Applications
- "Managing Model Repositories"

2

Managing Model Repositories

About Model Repositories

You can use model repositories to separate your project and model content, as well as to set permissions for objects within a repository. Some examples are having different repositories for test and production environments, or for different organizations. Model repositories are managed within the Settings window. You can add, delete, and rename a repository. The default repository, **Repository 1**, can be renamed, but it cannot be deleted.

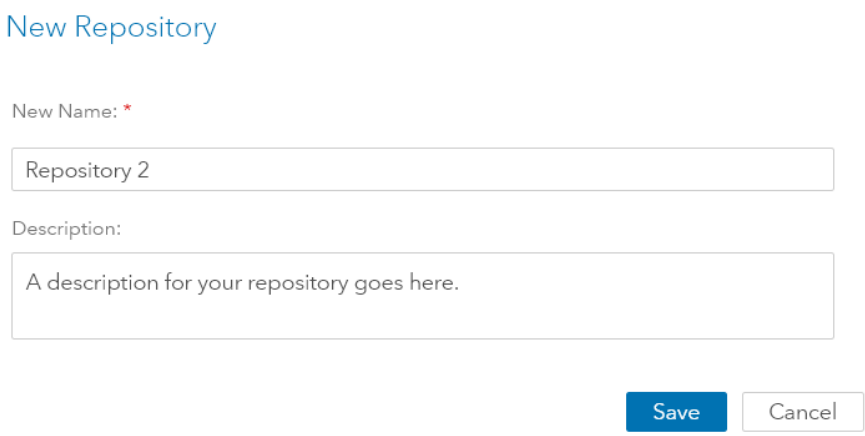
To access the **Repository** settings, click your name in the application bar and select **Settings** ⇨ **SAS Model Manager** ⇨ **Repository**.

Note: If you rename or delete the default repository `Repository 1`, the Model Repository service needs to be restarted by a system administrator. For more information, see "Restarting the Model Repository Service" in SAS Model Manager: Administrator's Guide .

Create a New Repository

1 Click .
2 Enter a name for the repository.
3 (Optional) Enter a description for the repository.

   Note: After you save the new repository, the description cannot be edited.



4 Click **Save**.

Rename a Repository

1 Select a repository, click ⋮, and select **Rename**.
2 Enter a new name for the repository.
3 Click **Rename**.

Delete a Repository

Note: By default, authenticated users can delete only repositories that they have created.

1 Select a repository and click 🗑.
2 In the confirmation message, click **Delete**.

3

Working with Projects

About Projects

A project consists of the models, variables, performance results, and other resources that you use to determine a champion model. For example, a banking project might include models, data, and tests that are used to determine the champion model for a home equity scoring application. The home equity scoring application predicts whether a bank customer is an acceptable risk for granting a home equity loan.

You create projects within folders. The models within a project are associated with a project version. A project version enables you to group models based on business requirements. The grouping of the models can be for a specific period of time.

Create a New Project

1. Click to navigate to the Projects category view.
2. Click **New Project**. The **New Project** window appears.
3. Enter a name for the project.

The initial version is displayed and reflects the level for sequential versions.

4 (Optional) Enter a description for the project.

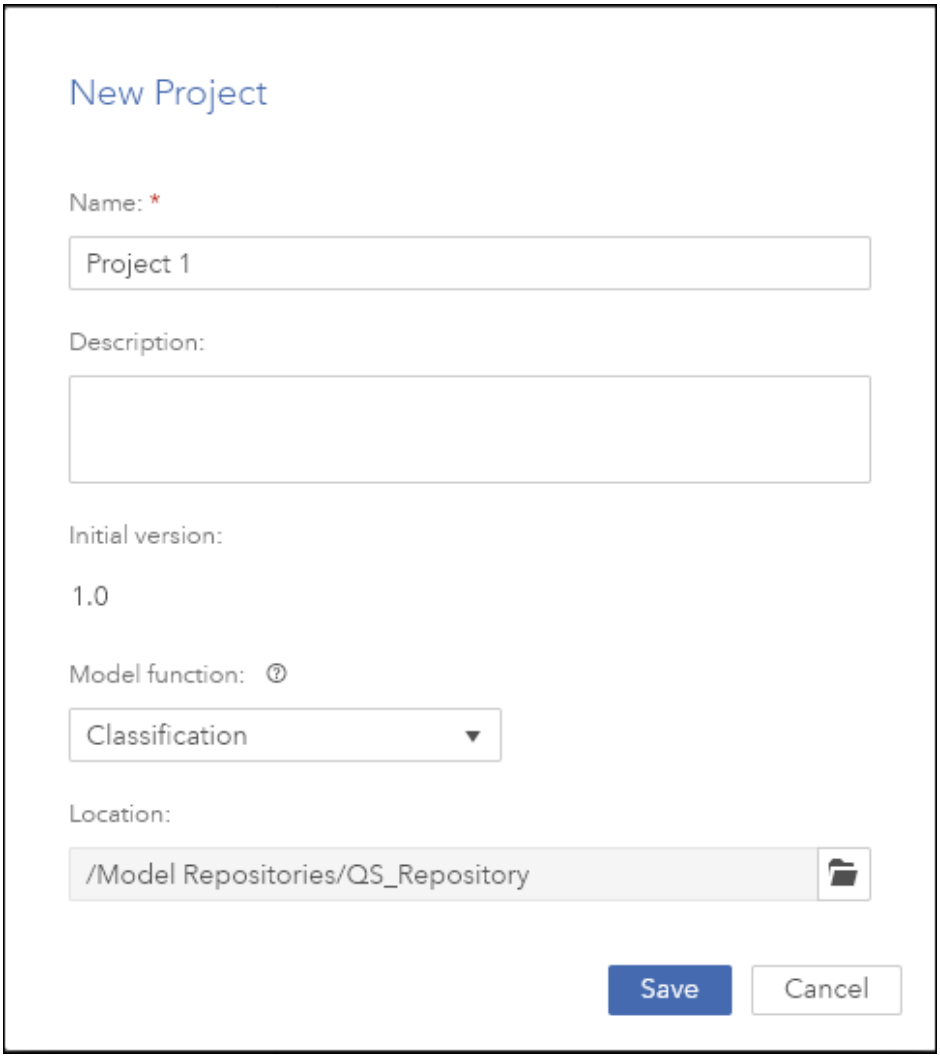
5 (Optional) Select a model function from the list or enter your own value. The model function indicates the type of output that your predictive model project generates.

6 Accept the default location or select a new location.

To select a new location, click 🗁, select the desired repository or folder, and then click **OK**.

Note: In the Choose a Location window, if you create a folder directly under the **Model Repositories** folder, it is considered a regular folder. Regular folders should not be used as a repository folder when creating projects. Repository folders must be created within the **Settings** window. For more information, see "Managing Model Repositories" on page 3.

Also, it is recommended that the `/DMRepository` repository folder should be reserved for models that are registered into the SAS Model Manager common model repository using Model Studio. For more information, see "Register Models" in SAS Visual Data Mining and Machine Learning: User's Guide .



7 Click **Save**.

Creating and Importing Models

After you create a project, you import models into a project version on the **Models** tab. A project can contain multiple versions. You can also copy a model from a folder or another project version. You can view models in all versions or in one selected version on the **Models** tab. After model evaluation, you set one of the candidate models as the champion model and can also set one or more models as challengers.

For more information, see the following:

- "Create a Model within a Project" on page 14
- "Import Models into a Project" on page 17
- "Copy a Model" on page 18
- "Evaluating Models" on page 25

Managing Variables

Input variables and output variables can be added to both project and model objects on the **Variables** tab. The same variable name cannot be used for both an input and output variable.

Add Variables from a Data Source

1. Click the **Variables** tab.
2. Click **Add Variable** and select **Data source**. The **Choose a Data Source** window appears.
3. Select the data source that you want to import the variables from and click **OK**. The **Select Variables** window appears.
4. Select input or output for the variable type.
5. Select the variables that you want to add and click ➡. You can also click ⏭ to add all of the variables from the available items list.
6. Click **OK**.
7. Click 💾.

Add Custom Variables

1. Click the **Variables** tab.
2. Click **Add Variable** and select **Custom variable**. The **Add Custom Variables** window appears.
3. Enter a name for the variable.
4. Select a data type and variable type.
5. Expand the **Optional** section to specify a length, measurement, and description for the variable.
6. Click **Add**.
7. Repeat steps 3 through 6 for each variable that you want to add.
8. Click **OK**.

Edit Variables

To edit variables:

1. Click on the name of the variable that you want to edit. The Edit Variable window appears.
2. Edit the properties as needed and click **OK**.

3 Click 💾.

Delete Variables

1 Select the check box for the variables that you want to delete, click ⋮, and then select **Delete**.
2 Click 💾.

Modifying Project Properties

Project properties contain the project metadata. Project metadata includes information such as the name of the project, the type of project, the project owner, the project identifier, the name and path of the repository, and of the tables and variables that are used by project processes. The project properties are organized into three types: General, Tags, and User-Defined.

Set Project General Properties

General Properties contains both system-defined properties that you cannot modify, and project specific properties that can be modified, such as the description of the project. None of the project properties are required, except for the name and location.

To set the project general properties, click the **Properties** tab, modify the property values, and then click 💾.

Table 3.1 List of General Properties

Property	Description
Name	Specifies the name of the project. A project can be renamed only from the Projects category view.
Description	Specifies the description of the project.
Model function	Specifies the type of output that your predictive model project generates. After it has been declared, the **Model function** property for a project cannot be changed. Ensure that the types of models that you are going to use in the project fit within the selected model function type. For more information, see Table 3.2 on page 9.
Location	Specifies the location of the project in the common model repository.
Champion version	Specifies the project version that contains the champion model.
Champion model name	Specifies the name of the model that is set as the project champion.
Default train table	Specifies the **Default train table** is also used to validate scoring functions or scoring model files when a user publishes the associated project champion model or challenger models to a database. This property is optional.
Training target variable	Specifies the name of the target variable that was used to train the model.

Property	Description
Target event value	Specifies the target variable value that defines the desired target variable event.
Target values	For class, nominal, ordinal, or interval targets, the set of possible outcome classes, separated by commas. For example, binary class target values might be `1, 0` or `Yes`, `No`. Nominal class target values might be `Low, Medium, High`. These values are for information only.
Target level	Specifies the target level of binary, nominal, ordinal, or interval.
Output event probability variable	Specifies the output event probability variable name, when the **Model function** property is set to **Classification**, **Analytical**, **Forecasting**, or **Transformation**.
Output prediction variable	The output prediction variable name, when the **Model function** property is set to **Prediction**, **Analytical**, **Forecasting**, or **Transformation**.
Output segmentation variable	The output segmentation variable name, when the **Model function** property is set to **Clustering**, **Analytical**, **Forecasting**, or **Transformation**.
UUID	Specifies the universally unique identifier for a project object.
External URL	Specifies a user-defined URL to a project object in another application or to documentation related to the project.
External project ID	Specifies the project ID for a project that was registered from an external application, such as Model Studio.

Table 3.2 *Types of Model Functions*

Model Function	Description
Analytical	Function for any model that is not Prediction, Classification, or Segmentation.
Classification	Function for models that have target variables that contain binary, categorical, or ordinal values.
Clustering	Function for segmentation or clustering models.
Forecasting	Function for models used to forecast future data based on past data.
Prediction	Function for models that have interval targets with continuous values.
Transformation	Function for models used to determine mathematical functions that can be used to stabilize variances, remove nonlinearity, and correct non-normality in variables to improve the fit of your model.

Add Tags

You can add one or more tags to a project. When you add tags to a project, they are added to a master list of tags that is available to be added to other projects within the same repository.

To add a tag:
1 On the **Properties** tab of a project, select **Tags**.
2 Select an existing tag or enter a name for a tag, and then click **+**.

   Note: The tag name can contain only alphanumeric characters, double-byte characters, the underscore (_), the hyphen (-), and the period (.). Spaces are not allowed.

   Repeat this step for additional tags.
3 Click 🖫.

To delete a tag, click 🗑 at the right-side of the row, and then click 🖫. The tag is removed from the project. It is also removed from the list of available tags, if it is not being used by another project.

Add User-Defined Properties

You can add your own project or model properties. The property-value pair is metadata for the project or model.

To add user-defined properties:
1 On the **Properties** tab, select **User-Defined**.
2 Click **Add Property**. The Add Property window appears.

   TIP If user-defined properties already exist, click **Add** above the table.

   a Enter a name for the property.

      Note: Spaces in the name of a property are currently not supported for the decimal and character data types.
   b Select a data type for the property.
   c Enter a value for the property.
   d Click **Add** to add the property to the list.
   e Repeat steps a through d for each property that you want to add.
3 Click **Add**.
4 Click 🖫.

To edit a property:
1 Click on a property name within the table.
2 Edit the name or the data type of the property.

   Note: To edit the value of a user-defined property after it has been created, you must delete the property and add a new property.
3 Click **OK**.
4 Click 🖫.

To delete properties, select one or more properties in the table, and then click **Delete**.

Managing Project Versions

A project version is a container of models. An initial version is created automatically when you create a project. You can view a list of the project versions on the **Models** tab in the **Version** drop-down list. The latest version is displayed by default. You can also choose to display all versions, create a new version, or manage existing

versions from the **Version** drop-down list. When you create a new project version, you can specify a name and description for the version, such as a time interval for a project cycle.

A version is a sequential number that increments by one each time you add a new version. A project can contain multiple editable versions. A project version is used to differentiate collections of models that are meant to solve the project's problem over time-boundaries. Your version might represent a calendar year, a retail season, or a fiscal quarter. A version contains all of the candidate model resources that you need to determine a champion model as well as all champion model resources. For example, you might develop models for a scoring program that determines whether a customer is eligible for a home equity loan.

Create a New Project Version

1. On the **Models** tab, click ▾ in the **Version** drop-down list, and select **New version**. The **New Project Version** window appears.
2. Enter a name for the version, or accept the default name (for example, Version 2).
3. (Optional) Enter a description for the version.
4. Click **Save**.

Manage Project Versions

In addition to creating a new project version, you can edit the description of a version, rename a version, or delete a version.

1. On the **Models** tab, click ▾ in the **Version** drop-down list, and select **Manage versions**. The **Manage Project Versions** window appears.
2. (Optional) Edit the description of a version.
3. Create a new project version.
   a. Click ✱. The **New Project Version** window appears.
   b. Enter a name for the version, or accept the default name (for example, Version 2).
   c. (Optional) Enter a description for the version.
   d. Click **Save**.
4. Rename a project version.
   a. Select a version and click **Rename**. The Rename window appears.
   b. Enter a new name for the version.
5. Delete a project version.

   Note: When only one project version exists, it cannot be deleted. You must also have the appropriate permissions to delete a version.

   a. Select a version and click **Delete**.
   b. Click **Delete** in the confirmation message.
6. Click **Close**.

Delete a Project

Note: You must also have the appropriate permissions to delete a version.

1. In the Projects category view, select one or more projects.

   Note: Open objects cannot be deleted.
2. Click ⋮ and select **Delete**.
3. In the confirmation message, click **Delete**.

Rename a Project

1 In the Projects category view, select a project, click ⋮, and select **Rename**.

   Note: Open objects cannot be renamed.
2 Enter a new name for the project.
3 Click **Rename**.

Search for Projects

In the **Projects** category view, you can search for projects by name using the search field above the projects list. You can also search for project objects across applications using the search field in the application bar.

For more information about searching, see “Search” in General Usage Help for SAS Viya Web Applications .

4

Working with Models

About Models

You can create new models or import existing models using the SAS Model Manager web application. Models can be stored within a folder or project version in the SAS Model Manager common model repository.

The Models category view enables you to access all of the models in the common model repository in one place. The models can be located in a folder, or a project version. You can import models, create new models, compare models, and export models. You can also search for models. The search field above the list of models enables you to filter the list by model name. The search field in the application bar enables you to search across the web applications that you have access to. You can also filter the search results by object type, modified by, and date modified.

Create a New Model

You can create a new model from one or model files and store it within a folder or within a project version. When you create a new model in the Models category view, you can choose a repository and folder to store the new model. When you create a model from the **Models** tab of a project, you can select a project version to store the new model.

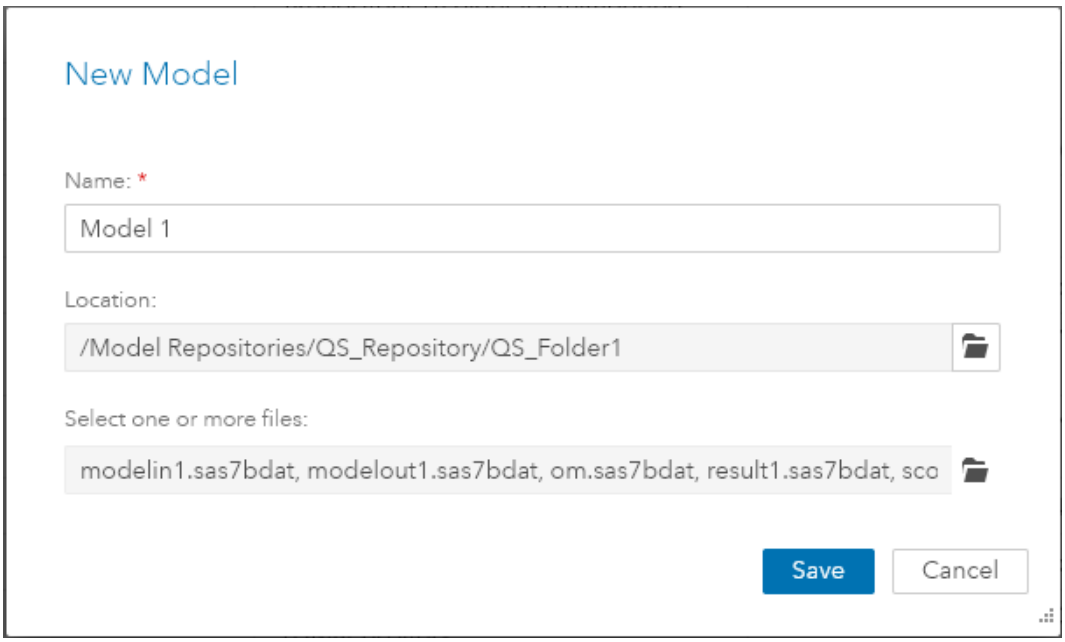
Create a Model within a Folder

To create a new model in a repository or folder:

1 Click to navigate to the Models category view.
2 Click **New Model**. The **New Model** window appears.
3 Enter a name for the new model.
4 Click to select a location to store the new model, and then click **OK**.

   Note: In the Choose a Location window, if you create a folder directly under the **Model Repositories** folder, it is considered a regular folder and should not be used as a repository folder when creating objects or importing models. Repository folders must be created within the **Settings** window. For more information, see "Managing Model Repositories" on page 3.

   Also, it is recommended that the `/DMRepository` should be reserved for projects and models that are registered from Model Studio. For more information, see "Register Models" in SAS Visual Data Mining and Machine Learning: User's Guide .
5 (Optional) Click , select one or more files to include within the new model, and then click **Open**.



6 Click **Save**. The new model object opens.

Create a Model within a Project

To create a new model within a project version:

Note: By default models are created within the latest project version. You can select a different project version from the **Version** drop-down list.

1 Click to navigate to the Projects category view.
2 Open a project.
3 Click **New Model**. The **New Model** window appears.
4 Enter a name for the new model.
5 (Optional) Click , select one or more files to include within the new model, and then click **Open**.

6 Click **Save**. The new model object opens.

TIP After you are done editing the model content, to return to the project, click .

Import Models

You can import models into a project or into a folder. Only specific file types can be used to import models into a project or folder.

Restrictions

Here are the file types that can be used to import models:

PMML
: a PMML XML file that contains model information. Predictive Modeling Markup Language (PMML) is an XML-based standard for representing data mining results. You can import PMML models that are produced by other applications. PMML 4.2 is supported. Models that are created using PMML 4.2 support DATA step score code. The file extensions can be .xml or .pmml, provided that the file contains valid PMML XML code.

SAS package (SPK) file
: a compressed container file that contains a mining result and model component files.

ZIP
: an archive file that contains model files. Model files that are associated with a specific model are stored within the ZIP file. The ZIP file can contain model folders at the same level or in a hierarchal folder structure. Each model folder within the ZIP file is imported as a separate model object that contains the contents of the model folder. When you import models from a ZIP file into a project version, the hierarchal folder structure is ignored.

Note: Exporting and re-importing analytic store models that were created using Model Studio is not supported.

Import Models into a Folder

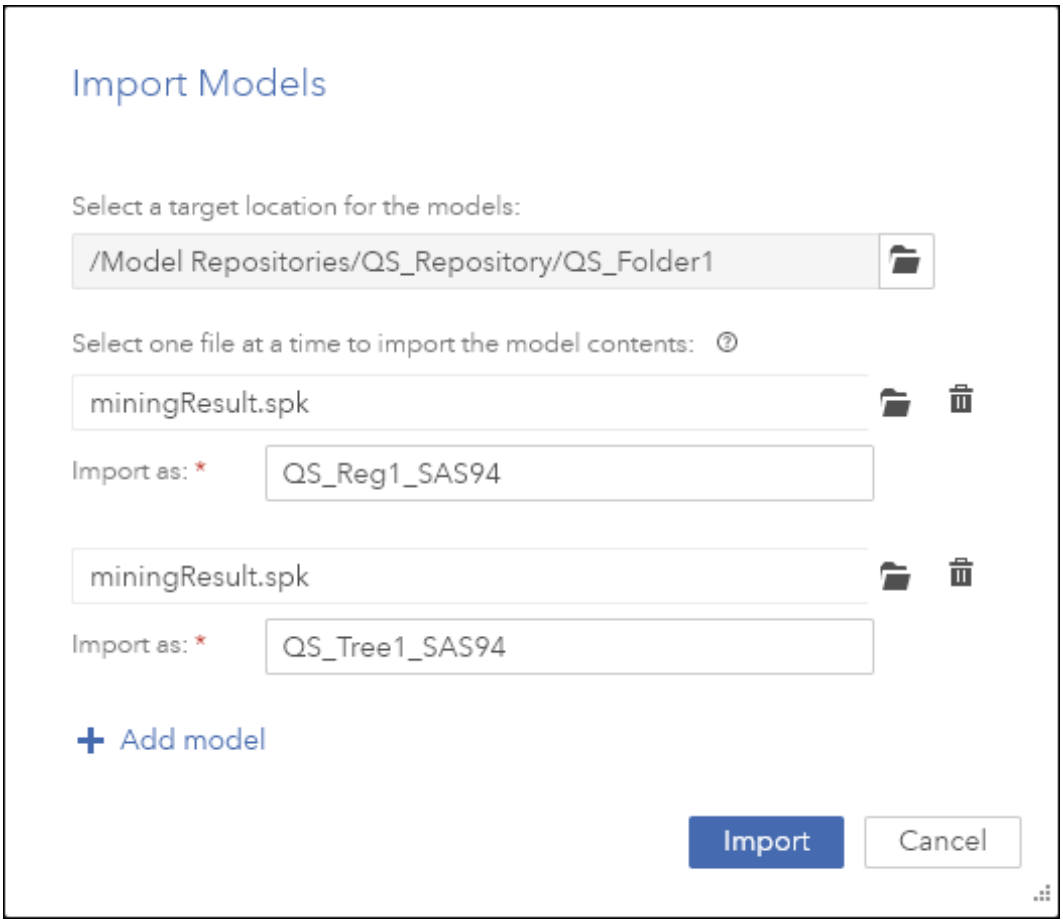
To import models into a folder:

1 Click to navigate to the Models category view.
2 Click **Import**. The **Import Models** window appears.
3 Click to select a location to store the models, and then click **OK**.

   Note: In the Choose a Location window, if you create a folder directly under the **Model Repositories** folder, it is considered a regular folder and should not be used as a repository folder when creating objects or importing models. Repository folders must be created within the **Settings** window. For more information, see "Managing Model Repositories" on page 3.

   Also, it is recommended that the `/DMRepository` should be reserved for projects and models that are registered from Model Studio. For more information, see "Register Models" in SAS Visual Data Mining and Machine Learning: User's Guide .
4 Click to select a file that contains your model contents. Select only one file at a time in the Open window. The name of the selected file is used as the default model name.

   Click **Open**.
5 Click **+ Add model** to add rows so that you can import more models.
6 Repeat steps 4 and 5 until you have selected all of the models that you want to import.



TIP To remove extra lines, click before you click **Import**.

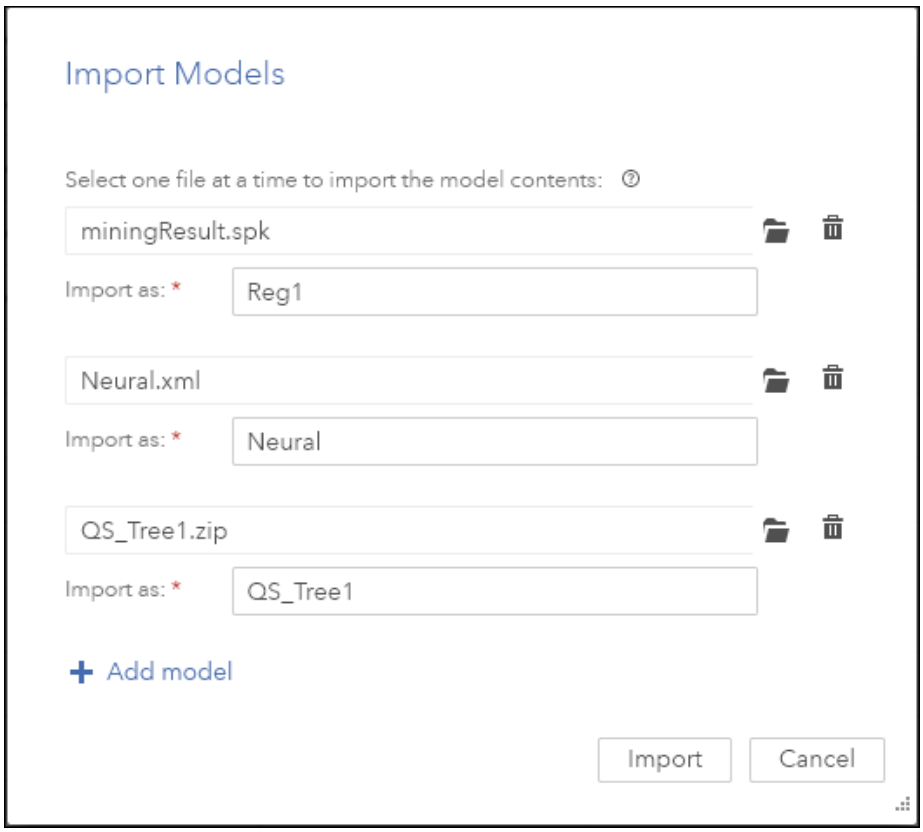
7 Click **Import**.

Import Models into a Project

To import model into a project version:

1 Click to navigate to the Projects category view.
2 Open a project.
3 Click **Import** and select **Import** from the drop-down list. The **Import Models** window appears.
4 Click to select a file that contains your model contents. Select only one file at a time in the Open window. The name of the selected file is used as the default model name.

   Click **Open**.
5 Click **+ Add model** to add rows so that you can import more models.
6 Repeat steps 2 and 3 until you have selected all of the models that you want to import.



TIP To remove extra lines, click before you click **Import**.

7 Click **Import**.

Export a Model

You can export one model at a time from the Models category view or from the **Models** tab of a project.

Note: Exporting and re-importing analytic store models that were created using Model Studio is not supported.

To export a model:

1 Select a model from the list.
2 Click and select **Export as ZIP**.

The contents of the model in a ZIP file is downloaded to your local machine.

Move or Copy a Model

You can move a model from a folder to another folder or project version using the Models category view. Only SAS Administrators and users that have Delete permission for the source location where the model resides and Write permission for the target location, can move a model. All other users by default, only can copy a model from a folder or another project from the **Models** tab of a project.

For more information, see "Managing Permissions" in SAS Model Manager: Administrator's Guide .

Move a Model

To move a model:
1 Click to navigate to the Models category view.
2 Select a model from the list.
3 Click ⋮ and select **Move**. The Choose a Location window appears.
4 Navigate to the folder or project version that you want to move the model to.
5 Click **OK**.

Copy a Model

To copy a model from a folder or another project:
1 Open a project and click the **Models** tab.
2 Click **Import** and select **Copy from** from the drop-down list. The **Choose a Model** window appears.
3 Click › to navigate to a folder or another project version.
4 Click › for the model folder and select the model object. The model object is indicated by the icon .
5 Click **OK**.

Note: Only the latest version of the source model is copied in to the project as a new mode object. The initial version for the model is 1.0.

Delete a Model

You can delete one or more models at a time from the Models category view or from the **Models** tab of a project.

Note: Open objects cannot be deleted.

1 Select one or more models.
2 Click ⋮ and select **Delete**.
3 In the confirmation message, click **Delete**.

Rename a Model

You can rename one model at a time from the Models category view or from the **Models** tab of a project.

Note: Open objects cannot be renamed.

To rename a model:
1 Select a model, click ⋮, and select **Rename**.
2 Enter a new name for the model.
3 Click **Rename**.

Manage Model Content and Versions

When you open a model, you can manage model files, add model input and output variables, modify the model properties, and add or view model versions. You can open a model from the Models category view and from the **Models** page of a project.

Managing Model Files

On the **Files** tab of a model, you can add, delete, and download files, as well as assign roles to model files.

You can add any type of file to a model. You can also edit supported file types in the code editor. File types that are not supported by the editor can be downloaded.

Add Files

1 Click +. The **Add Model Files** window appears.
2 Click 🗀, select one or more files to add to the model, and then click **Open**.
3 Click **Add**.

Assign Model File Roles

You can place your pointer over a model file to view the file properties. Roles might need to be assigned for your model files. To assign a role, select the file and click .

Some roles such as **Score code** are automatically assigned when you import or create new models based on their filenames.

Delete Model Files

Select the file and click . In the confirmation, click **Delete**.

Download Model Files

Select the file and click . The model file is downloaded to your local machine.

Managing Variables

Input variables and output variables can be added to both project and model objects on the **Variables** tab. The same variable name cannot be used for both an input and output variable.

Add Variables from a Data Source

1 Click the **Variables** tab.
2 Click **Add Variable** and select **Data source**. The **Choose a Data Source** window appears.
3 Select the data source that you want to import the variables from and click **OK**. The **Select Variables** window appears.

4 Select input or output for the variable type.
5 Select the variables that you want to add and click ➡. You can also click ⏭ to add all of the variables from the available items list.
6 Click **OK**.
7 Click 💾.

Add Custom Variables

1 Click the **Variables** tab.
2 Click **Add Variable** and select **Custom variable**. The **Add Custom Variables** window appears.
3 Enter a name for the variable.
4 Select a data type and variable type.
5 Expand the **Optional** section to specify a length, measurement, and description for the variable.
6 Click **Add**.
7 Repeat steps 3 through 6 for each variable that you want to add.
8 Click **OK**.

Edit Variables

To edit variables:
1 Click on the name of the variable that you want to edit. The Edit Variable window appears.
2 Edit the properties as needed and click **OK**.
3 Click 💾.

Delete Variables

1 Select the check box for the variables that you want to delete, click ⋮, and then select **Delete**.
2 Click 💾.

Modifying Model Properties

Model properties contain the model metadata. Model metadata includes information such as the name of the model, the type of model, the modeler, the model identifier, the name and path of the repository, and of the tables and variables that are used by model processes. The model properties are organized into two types: General and User-Defined.

Set Model General Properties

General Properties contains both system-defined properties that you cannot modify, and model specific properties that can be modified, such as the description of the project.

To set the model general properties, click the **Properties** tab, modify the property values, and then click 💾.

Table 4.1 List of General Properties

Property	Description
Name	Specifies the name of the model It only can be renamed from the Models category view or the **Models** tab of a project.
Description	Specifies the description of the model.

Property	Description
Location	Specifies the location of the model in the common model repository.
Project name	Specifies the name of the project that contains the model.
Project version	Specifies the project version that contains the model.
Function	Specifies the type of output that your model generates. For more information, see Table 3.2 on page 9.
Score code type	Specifies the type of score code that your model uses. A value must be specified in order for you to be able to publish a model, run a test, or monitor performance for a model. You can select a value from the list or enter your own value. User-defined values are not added to the list. Instead, they are stored within the model properties. When you are specifying a score code type, here are some restrictions to be aware of: ■ Only models with a score code type of DATA step, SAS program, DS2 package, DS2 embedded process, DS2 multi-type, and Analytic store can be scored. ■ Models with a score code type of SAS program cannot be published. ■ Models with a score code type of DS2 package cannot be published to CAS, Hadoop, or Teradata. ■ Models with score code types of Analytic store and DS2 embedded process cannot be published to SAS Micro Analytic Service.
Train table	Specifies the **Train table** that is used to validate scoring functions or scoring model files when a user publishes the associated project champion model or challenger models to a database. This property is optional.
Train code type	Specifies the type of train code that your model uses. This property is for informational purposes only. You can select a value from the list or enter your own value. User-defined values are not added to the list. Instead, they are stored within the model properties.
Algorithm	Specifies the computational algorithm that is used for the selected model.
Target variable	Specifies the name of the target variable.
Target event value	Specifies the target variable value that defines the desired target variable event.
Target level	Specifies the target level of binary, nominal, ordinal, or interval.
Output event probability variable	Specifies the output event probability variable name, when the **Model function** property is set to **Classification**, **Analytical**, **Forecasting**, or **Transformation**.

Property	Description
Output prediction variable	The output prediction variable name, when the **Model function** property is set to **Prediction**, **Analytical**, **Forecasting**, or **Transformation**.
Output segmentation variable	The output segmentation variable name, when the **Model function** property is set to **Clustering**, **Analytical**, **Forecasting**, or **Transformation**.
Modeler	Specifies the user ID for the user that built the model.
Tool	Specifies the tool that was used to build the model. An example is Model Studio.
Tool version	Specifies the version number of the tool that is specified in the **Tool** property.
UUID	Specifies the universally unique identifier for a model object.
External model ID	Specifies the model ID for a model that was registered from an external application, such as Model Studio.
External URL	Specifies a user-defined URL to a model object in another application or to documentation related to the model.

Add User-Defined Properties

You can add your own project or model properties. The property-value pair is metadata for the project or model.

To add user-defined properties:

1. On the **Properties** tab, select **User-Defined**.
2. Click **Add Property**. The Add Property window appears.

   TIP If user-defined properties already exist, click **Add** above the table.

   a. Enter a name for the property.

      Note: Spaces in the name of a property are currently not supported for the decimal and character data types.

   b. Select a data type for the property.
   c. Enter a value for the property.
   d. Click **Add** to add the property to the list.
   e. Repeat steps a through d for each property that you want to add.
3. Click **Add**.
4. Click 🖫.

To edit a property:

1. Click on a property name within the table.
2. Edit the name or the data type of the property.

   Note: To edit the value of a user-defined property after it has been created, you must delete the property and add a new property.
3. Click **OK**.
4. Click 🖫.

To delete properties, select one or more properties in the table, and then click **Delete**.

Managing Model Versions

The current version of a model is the latest version in which the model properties and file contents are editable. If you add a new model version manually or perform an action that automatically creates a new model version (such as setting it as the champion model or publishing a champion model from the project level), a snapshot of the model's contents is taken and a version number is assigned. However, the contents of the new model version that is created can no longer be edited. You can only view the contents of the new model version. Model versions cannot be deleted.

Set the Displayed Version

The displayed version is the version whose information is displayed on the other tabs, such as the **Files**, **Variables**, and **Properties** tabs. The version number for the displayed version appears next to the model name in the object title bar. On the **Versions** tab, a ✓ indicates the displayed version. To change the displayed version, select the version that you want to view, and click **Set Version**.

Note: Here are a few restrictions when creating a new model version.

- The current version of an object is the version that has the highest version number. When you create a new version, SAS Model Manager locks the current version before it creates the new version.
- You cannot save changes to a version that is locked. If you modify a version that is locked and click 💾, SAS Model Manager asks you if you want to save the changes to the current unlocked version.
- You cannot unlock a version.

Create a New Version

To create a new version:

1. On the **Versions** tab, click **New Version**. The New Version window appears.
2. Select the version type: **Minor** or **Major**. Version numbers follow the format *Major.Minor*. If you select **Major**, the number to the left of the period is incremented. If you select **Minor**, the number to the right of the period is incremented.
3. Click **Save**.

Search for Models

In the **Models** category view, you can search for models by name using the search field above the models list. You can also search for model objects across applications using the search field in the application bar.

For more information about searching, see "Search" in General Usage Help for SAS Viya Web Applications .

5

Evaluating Models

About Evaluating Models

The goal of a modeling project is to identify a champion model that an external scoring application uses to predict an outcome. SAS Model Manager provides tools to evaluate candidate models and declare a project champion model. You can compare and assess models, run a test on a model, and monitor performance of a model. You can also, publish models to CAS, Hadoop, SAS Micro Analytic Service, and Teradata so that you can be accessed by scoring applications.

Compare Models

You can compare and assess one or more models. When you compare models, the model comparison output includes model properties, user-defined properties, variables, fit statistics, and plots for the models. The fit statistics, as well as plots for lift and ROC, are produced using the ASSESS procedure. The fit statistics and plots are displayed only for the SAS Visual Data Mining and Machine Learning models that are created using Model Studio. If you were to look at the **Files** tab of a model object, you would see that JSON files (dmcas_fitstat.json, dmcas_lift.json, dmcas_roc.json) are included. These JSON files are used to show the fit statistics and plots when comparing models in SAS Model Manager.

To compare models:

1. Select one or more models.
2. From the Models category view, click ⋮ and select **Compare**.

   On the **Models** tab of project, click **Compare**.

The **Compare** page appears.

3 Click **Show Differences**. The default is to show all of the comparison model content.

4 Review the differences for the following model information:

- Model properties
- User-defined properties
- Input variables
- Output variables
- Target variable
- Fit statistics
- Lift and ROC plots

5 Click **Close**.

Test Models

The purpose of a test is to run the score code of a model and produce scoring results that you can use for scoring accuracy and performance analysis. The test uses the input data source to generate the test output table. If your environment has its own means of executing the score code, then your use of the SAS Model Manager scoring tests is mostly limited to testing the score code. Otherwise, you can use the tests both to test your score code and execute it in a production environment.

Only models that have the score code model file role assigned, and that have a score code type of DATA step, SAS program, DS2 package, DS2 embedded process, DS2 multi-step, and Analytic store can be scored. For more information, see "Assign Model File Roles" on page 19 and "Set Model General Properties" on page 20.

Create and Run a New Test

By default, only the user who creates a test definition can view, update, or delete the test definition, as well as run the test and view the test results. For more information, see "Default Permissions" in SAS Model Manager: Administrator's Guide .

1 On the **Tests** tab of a project, click **New Test**. The New Test window appears.

2 Enter a name for the test if you do not want to use the default name.

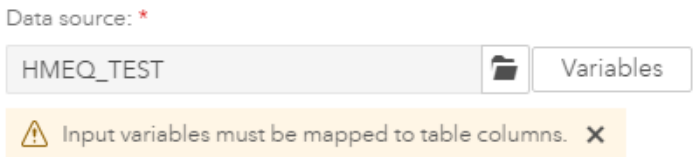
3 (Optional) Enter a description for the test.

4 Click **Choose Model** and select a model to test.

5 Click , select the input table for the test, and click **OK**.

6 Map variables.

Note: SAS Model Manager automatically maps model input variables to the columns in the input table when the names and data types of the variables match those of the table columns. If any input variables cannot be mapped automatically, a warning message is displayed.



TIP You can change the automatic variable mappings.

To map variables:

a Click **Variables**. The Variable Mappings window appears.

   b For each input variable, select the table column to which the variable should be mapped.
   c Click **OK**.
7 (Optional) Expand the **Advanced** section to display the advanced options.
8 (Optional) By default, the library location is the same as that for the input data source. Click to specify a different library to store the new test output table that is created when the test is run.
9 Click **Run** to run the test. Alternatively, click **Save** to save the test definition without running it.

   The status of the test is indicated by the icon in the **Status** column.

Icon	Status
	The test is not ready to run. The test definition is not complete, or it might contain errors.
	The test is defined correctly and is ready to run.
	The test is running.
	The test completed successfully.
	The test completed, but warnings were issued in the SAS log. The URI to the log file is shown on the **Test Results** page.
	The test did not run successfully. Check the SAS log for information. The URI to the log file is shown on the **Test Results** page.

10 Click in the **Results** column to view the results of the test. The **Test Results** page displays information about the test, including the URIs for the test definition and test results. It also includes URIs to the SAS code that was run by SAS Model Manager, the output data set, and the SAS log that was generated when the code was run.

   You can click the **Output**, **Code**, or **Log** pages to view the test result details.

   For more information, see "Working with Test Output Data" on page 27.

Edit a Test

1 Click the **Tests** tab of a project.
2 Click on a test name. The Edit Test window appears.
3 Edit the test properties as needed, and then click **Save** or **Run**.

Note: You can also select a check box for a test and click **Run** in the toolbar to rerun a test.

Delete a Test

1 Click the **Tests** tab of a project.
2 Select one or more tests and click .

Working with Test Output Data

After you run a test, you can work with the output table in other SAS applications to analyze the data, create and compare models, discover relationships hidden in the data, and generate reports based on the data.

On the **Test Results** page, select the **Output** table in the navigation pane, click **Actions**, and select one of the following options:

Prepare Data
opens the output table in SAS Data Studio. SAS Data Studio enables you to perform data transforms such as joining tables, appending data to a table, transposing columns, creating calculated columns, and so on. For more information, see *SAS Data Studio: User's Guide*.

Manage Data
opens SAS Data Explorer. SAS Data Explorer enables you to import data, connect to databases, and load tables into memory. For more information, see *SAS Data Explorer: User's Guide*.

Explore and Visualize Data
opens the output table in SAS Visual Analytics. SAS Visual Analytics enables you to create, test, and compare models based on the patterns discovered during exploration of the data. You can export the model before or after performing model comparison for use with other SAS products or to put the model into production. SAS Visual Analytics supports a range of visualization, discovery, and reporting features. For more information, see *SAS Visual Analytics: Overview*.

Build Models
enables you to create a new project in Model Studio using the output table as the data source. Model Studio is an integrated visual environment that provides a suite of analytic data mining tools to facilitate end-to-end data mining analysis. SAS Model Studio is a common interface for SAS Visual Forecasting, SAS Visual Data Mining and Machine Learning, and SAS Visual Text Analytics. For more information, see *Model Studio: Getting Oriented*.

Explore Lineage
opens SAS Lineage Viewer. SAS Lineage Viewer enables you to better understand the relationships between objects in your SAS Viya applications. These objects include data, transformation processes, reports, and visualizations. For more information, see *SAS Lineage Viewer: User's Guide*.

Note: The actions available to you depend on the applications that are available at your site.

Set Champion and Challenger Models

The champion model is the best predictive model that is chosen from a pool of candidate models. Before you identify the champion model, you can evaluate the structure, performance, and resilience of candidate models. When a champion model is ready for production scoring, you set the model as the champion model. The project version that contains the champion model becomes the champion version for the project. You can publish the champion model to CAS, Hadoop, SAS Micro Analytic Service, and Teradata.

You use challenger models to test the strength of champion models. The champion model for a project can have one or more challenger models. A model can be flagged as a challenger model only after a champion model for the project has been selected. A challenger model can be located in any version of a project.

Set a Champion Model

1. Click the **Models** tab of a project.
2. Select a model, click ⋮, and select **Set as champion**.
3. If the **Select Project Output Variables** window appears, select the model output variables to use as project level output variables. You can use the same variable names or specify different names for the project output variables.

   Click **Save**.
4. If the model input variables are not project input variables, you are prompted to add the input variables to the project.

   In the confirmation message, click **Yes**.

   Note: If you click **No**, the model is not set as the project champion.

Set a Challenger Model

1 Click the **Models** tab of a project.
2 Select a model, click ⋮, and select **Set as challenger**.
3 If the **Select Project Output Variables** window appears, select the model output variables to use as project level output variables. You can use the same variable names or specify different names for the project output variables.

   Click **Save**.
4 If the model input variables are not project input variables, you care prompted to add the input variables to the project.

   In the confirmation message, click **Yes**.

   Note: If you click **No**, the model is not set as a challenger model.

Clear Model Role

1 Click the **Models** tab of a project.
2 Select the champion model or a challenger model, click ⋮, and select **Clear role**.

Monitor Performance

To ensure that a champion model in a production environment is performing efficiently, you can collect performance data that has been created by the model at intervals that are determined by your organization. A performance data set is used to assess model prediction accuracy. It includes all of the required variables as well as one or more actual target variables. For example, you might want to create performance data sets monthly or quarterly and then use SAS Model Manager to create a performance definition that includes each time interval.

You can create the performance monitoring output by writing your own SAS program using the performance monitoring macros that are provided with SAS Model Manager. You can then submit your program using SAS Studio. The performance results tables that are produced using the macros can then be viewed in SAS Studio, SAS Environment Manager, or SAS Visual Analytics.

Performance monitoring can be performed on champion, challenger, and candidate models.

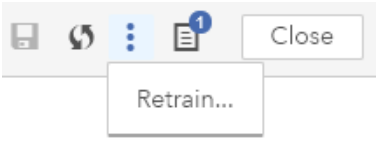
For more information, see "Performance Monitoring Macros" in SAS Model Manager: Macro Reference .

Retrain a Project from Model Studio

Only projects that have been registered from Model Studio into the SAS Model Manager common model repository can be retrained. Model studio projects and their models are stored in the `DMRepository` repository folder when they are registered.

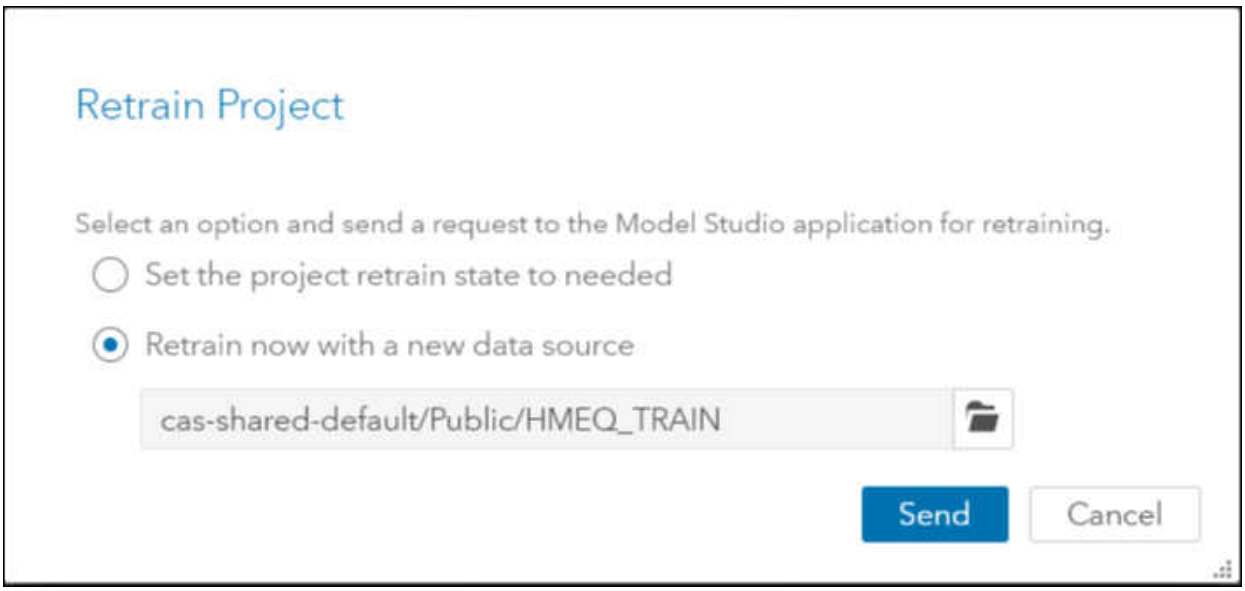
To send a retrain request for a Model Studio project:
1 Click to navigate to the Projects category view.
2 Open a project that was registered from Model Studio.

   Note: If a project has been registered from Model Studio, an actions menu button appears to the left of the open items icon.



3 Click ⋮ and select **Retrain**.
4 Select an option to send a retrain request to Model Studio.

- Select the **Set the project retrain state to needed** option if you want to indicate that the project and its models are ready for retrain.
- Select the **Retrain now with a new data source** option, if you want to select a new data source and send a request to Model Studio to retrain the project now. Click 🗀 to choose a data source.



5 Click **Send**.

6

Publishing Models

About Publishing Models

You can publish models to a publish destination, so that it can be used by other applications for tasks such as scoring. Models can be published to destinations that are defined for CAS, Hadoop, SAS Micro Analytic Service, and Teradata.

Models can be published from different locations within the SAS Model Manager web application. The project champion model can be published from the Projects category view. Models can also be published from the **Models** tab of a project.

Publish destinations are defined by a SAS Administrator. For more information, see "Configuring Publish Destinations" in SAS Model Manager: Administrator's Guide .

Requirements and Restrictions

Before you can publish a model, you must first set the score code type for a model. Only models with a score code type of DATA step, DS2 package, DS2 embedded process, DS2 multi-type, and Analytic store can be published.

Here are some other restrictions to be aware of as well:

- Models with a score code type of **SAS program** or user-defined values cannot be published.
- Models with a score code type of **DS2 package** cannot be published to CAS, Hadoop, or Teradata.
- Models with score code types of **Analytic store** and **DS2 embedded process** cannot be published to SAS Micro Analytic Service.

For more information, see "Set Model General Properties" on page 20.

Publish a Project Champion Model

1 Click to navigate to the Projects category view.

2 Select a project, click ⋮, and select **Publish**. The **Publish Models** window appears.
3 Select a publish destination from the list of destinations.
4 (Optional) If you have previously published the project champion model, expand the **Items to Publish** section, and select the check box in the **Replace** column.
5 Click **Publish**. The Publishing Results window appears. The status of the publishing request is displayed in the **Status** column.
6 When the status changes to `Published successfully`, click **Close**.

Publish Models

You can publish one or more models from the **Models** tab of a project, including the champion model.

1 Click to navigate to the Projects category view.
2 Open a project.
3 Select one or more models.
4 Click ⋮, and select **Publish**. The **Publish Models** window appears.
5 Select a publish destination from the list of destinations.
6 (Optional) If you have previously published a model, expand the **Items to Publish** section, and select the check box in the **Replace** column for each model that you want to replace.
7 Click **Publish**. The Publishing Results window appears. The status of the publishing request is displayed in the **Status** column.

   Note: When you are publishing to a SAS Micro Analytic Service destination, the **Micro Analytic Module** column is also displayed with a URL to the published model.
8 When the status changes to `Published successfully`, click **Close**.

7

Using SAS Workflow with SAS Model Manager

About Using Workflows

SAS Model Manager uses the **Workflow** tab within a project and the Tasks category view to interface with SAS Workflow. A workflow is an instance of a workflow definition. A workflow can be used to track the progress of objects, such as projects. An authorized user can use SAS Workflow Manager to create workflow definitions and to make them available to SAS Model Manager for use. Workflow definitions contain the set of tasks, participants, and data objects that comprise a business task. The status that you select when completing a task determines the next task in the workflow. All users can access the Tasks category view.

For information about creating workflow definitions, see *SAS Workflow Manager: Quick Start Tutorial*.

Note: To start a workflow or work with tasks you must be in the SAS Workflow Process Administrators group. For more information, see "Managing Permissions" in SAS Model Manager: Administrator's Guide .

Start a New Workflow

When you start a new workflow, it is associated with the project. For a specific project, only one workflow can be in progress at a time. The tasks within a workflow must be completed or the in-progress workflow process must be terminated, before a new workflow can be started.

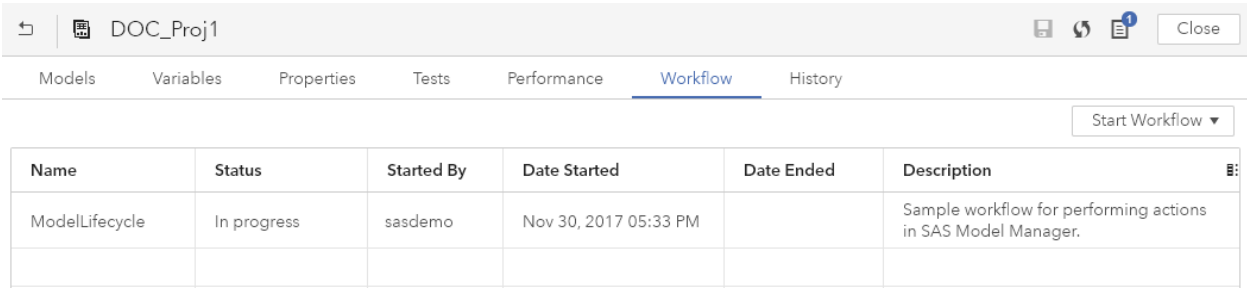
To start a workflow:

1. Open a project and click the **Workflow** tab.
2. Click **Start Workflow** and select a workflow definition from the list. The Start Workflow window appears.
3. Specify values for any prompts that are displayed.

   Note: What is displayed in the **Start Workflow** window depends on what is configured in the workflow definition start node. If prompts are not configured for the start node, the default text is "Are you sure you want to start this workflow?".

For example, specify a value for the project name.

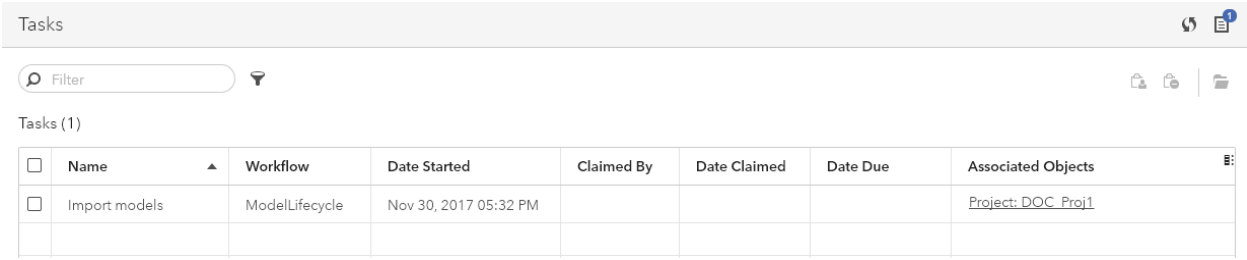
4 Click **Start**. The workflow is added to the list with a status of “In progress”.



Working with Tasks

About Tasks

The Tasks category view displays the tasks for workflows that are in progress, and that you have been assigned to as a potential owner and that have been claimed by you.



In the Tasks category view, you can perform the following:

- claim a task
- open a task
- release a task

Complete a Task

1 Click on a task to open it.
2 Click to claim the task.
3 Specify values for any prompts that are displayed on the **Prompts** tab.
4 Click the **Properties** tab to view task properties, including the associated object.
5 Click on the name of the associated object to open it.
6 Complete any actions that are associated with the task. An example is importing models into a project.
7 Click to switch back to the task object.
8 Click **Complete**.
9 Click **Complete** in the confirmation message.

Note: Alternatively, you can select one or more tasks and can click to open them or click to claim them.

Release a Task

1 Select one or more tasks.
2 Click .

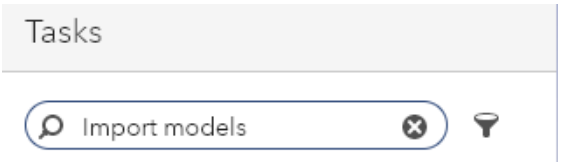
Note: Alternatively, if you already have the task open, you can click .

Filter Tasks

From the Tasks category view, you can filter the tasks that are displayed in the list. Here are the two options available for filtering tasks:

- Enter a value in the **Filter** field above the list to filter the list by task name.

  Figure 7.1 *Example of Filtering by Task Name*

- Click to filter the list by workflow name, date started, date claimed, or date due.