ALIBABA CLOUD

Alibaba Cloud

Machine Learning Platform for AI

Quick Start

Document Version: 20220623

Alibaba Cloud

Legal disclaimer

Alibaba Cloud reminds you to carefully read and fully understand the terms and conditions of this legal disclaimer before you read or use this document. If you have read or used this document, it shall be deemed as your total acceptance of this legal disclaimer.

1. You shall download and obtain this document from the Alibaba Cloud website or other Alibaba Cloud-authorized channels, and use this document for your own legal business activities only. The content of this document is considered confidential information of Alibaba Cloud. You shall strictly abide by the confidentiality obligations. No part of this document shall be disclosed or provided to any third party for use without the prior written consent of Alibaba Cloud.

2. No part of this document shall be excerpted, translated, reproduced, transmitted, or disseminated by any organization, company or individual in any form or by any means without the prior written consent of Alibaba Cloud.

3. The content of this document may be changed because of product version upgrade, adjustment, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and an updated version of this document will be released through Alibaba Cloud-authorized channels from time to time. You should pay attention to the version changes of this document as they occur and download and obtain the most up-to-date version of this document from Alibaba Cloud-authorized channels.

4. This document serves only as a reference guide for your use of Alibaba Cloud products and services. Alibaba Cloud provides this document based on the "status quo", "being defective", and "existing functions" of its products and services. Alibaba Cloud makes every effort to provide relevant operational guidance based on existing technologies. However, Alibaba Cloud hereby makes a clear statement that it in no way guarantees the accuracy, integrity, applicability, and reliability of the content of this document, either explicitly or implicitly. Alibaba Cloud shall not take legal responsibility for any errors or lost profits incurred by any organization, company, or individual arising from download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, take responsibility for any indirect, consequential, punitive, contingent, special, or punitive damages, including lost profits arising from the use or trust in this document (even if Alibaba Cloud has been notified of the possibility of such a loss).

5. By law, all the contents in Alibaba Cloud documents, including but not limited to pictures, architecture design, page layout, and text description, are intellectual property of Alibaba Cloud and/or its affiliates. This intellectual property includes, but is not limited to, trademark rights, patent rights, copyrights, and trade secrets. No part of this document shall be used, modified, reproduced, publicly transmitted, changed, disseminated, distributed, or published without the prior written consent of Alibaba Cloud and/or its affiliates. The names owned by Alibaba Cloud shall not be used, published, or reproduced for marketing, advertising, promotion, or other purposes without the prior written consent of Alibaba Cloud. The names owned by Alibaba Cloud include, but are not limited to, "Alibaba Cloud", "Aliyun", "HiChina", and other brands of Alibaba Cloud and/or its affiliates, which appear separately or in combination, as well as the auxiliary signs and patterns of the preceding brands, or anything similar to the company names, trade names, trademarks, product or service names, domain names, patterns, logos, marks, signs, or special descriptions that third parties identify as Alibaba Cloud and/or its affiliates.

6. Please directly contact Alibaba Cloud for any errors of this document.

Document conventions

Style	Description	Example
⚠ Danger	A danger notice indicates a situation that will cause major system changes, faults, physical injuries, and other adverse results.	⚠ Danger: Resetting will result in the loss of user configuration data.
🔔 Warning	A warning notice indicates a situation that may cause major system changes, faults, physical injuries, and other adverse results.	🔔 Warning: Restarting will cause business interruption. About 10 minutes are required to restart an instance.
🔈 Notice	A caution notice indicates warning information, supplementary instructions, and other content that the user must understand.	🔈 Notice: If the weight is set to 0, the server no longer receives new requests.
? Note	A note indicates supplemental instructions, best practices, tips, and other content.	? Note: You can use Ctrl + A to select all files.
>	Closing angle brackets are used to indicate a multi-level menu cascade.	Click **Settings**> **Network**> **Set network type**.
Bold	Bold formatting is used for buttons , menus, page names, and other UI elements.	Click **OK**.
`Courier font`	Courier font is used for commands	Run the `cd /d C:/window` command to enter the Windows system folder.
Italic	Italic formatting is used for parameters and variables.	`bae log list --instanceid` *Instance_ID*
[] or [a|b]	This format is used for an optional value, where only one item can be selected.	`ipconfig [-all|-t]`
{} or {a|b}	This format is used for a required value, where only one item can be selected.	`switch {active|stand}`

Table of Contents

1.Overview

This topic describes how to create a pipeline in Machine Learning Platform for AI (PAI).

Machine Learning Designer allows you to create a pipeline by using a template or create a custom pipeline.

Machine Learning Designer provides demos to show you how to create a custom pipeline and create a pipeline by using a template.

- Demo for a custom pipeline

  Machine Learning Designer provides more than one hundred algorithm components that you can use to create custom pipelines and supports access to a variety of data sources, such as MaxCompute and Object Storage Service (OSS). This improves your modeling efficiency. The following procedure provides an example on how to create a custom pipeline:

  i. Prepare data

  Upload raw data to MaxCompute or OSS, and configure the data source for the pipeline.

  ii. Prepare and preprocess data

  Preprocess the raw data to generate a model training set and a model prediction set.

  iii. Data visualization

  Process the source data or the intermediate results in a visualized manner to obtain data analysis results.

  iv. Algorithm modeling

  Create a model by using the algorithm components that meet your business requirements and the model training set.

  v. Evaluate the model

  Make predictions by using the trained model and the model prediction set. Evaluate the quality of the model based on the correct answers in the model prediction set.

- Demo for a pipeline created by using a template

  You can create a pipeline by using a template. After the pipeline is created and works as expected, you can deploy the model. For more information about the demo, see Create experiments based on templates.

2.Demo for a custom pipeline

2.1. Prepare data

This topic describes how to prepare data by using Integrated Development Environment (IDE) to upload data.

Prerequisites

A project is created. For more information, see Create a project.

Context

Machine Learning Platform for AI allows you to store data in MaxCompute and Object Storage Service (OSS).

- Data stored in MaxCompute is used by general-purpose algorithm components.

> ? **Note** When the data size is smaller than 20 MB, we recommend that you use IDE to upload data. When the data size is larger than 20 MB, we recommend that you use the command-line tool to upload data. For more information, see Tunnel command usage.

- Structured or unstructured data stored in OSS is used by algorithm components of deep learning.

Procedure

1.
2. Upload data.
   i. In the left-side navigation pane, click **Data Source**.
   ii. In the lower-left corner, click **Create Table**.
   iii. In the Create Table dialog box, set the **Table name** and **Lifecycle (Days)** parameters.
   iv. In the **Schema** section, click the ⊕ icon. Then, enter a name in the **Column Name** column and specify the data type in the **Type** column.
   v. Click **Next**.
   vi. Click **Select File** and follow the instructions to upload on-premises files.
   vii. Click **OK**.
3. Create an experiment.
   i. In the left-side navigation pane, click **Home**.
   ii. In the upper-right corner, click **New** and select **New Experiment**.
   iii. In the New Experiment dialog box, set the **Name** parameter and click **OK**.
4. Configure the data source.
   i. In the left-side navigation pane, click **Components**.
   ii. In the Components pane, click **Data Source/Target**. Then, drag the **Read MaxCompute Table** component to the canvas.

iii. Click the **Read MaxCompute Table** component on the canvas. On the **Select Table** tab on the right side, enter the name of the created table in the **Table Name** field.

iv. Click the **Fields Information** tab to view the columns, data type, the value range of the first 100 rows of the table.

What's next

After data preparation is complete, you need to preprocess the data. For more information, see Prepare and preprocess data.

2.2. Prepare and preprocess data

In most cases, you need to prepare and preprocess the data that is required to build and test a model. The prepared data is then further processed based on your business requirements for model development. This topic describes how to prepare and preprocess data in Machine Learning Platform for AI (PAI). In this example, public data provided by PAI is used.

Prerequisites

A pipeline is created. For more information, see Prepare data.

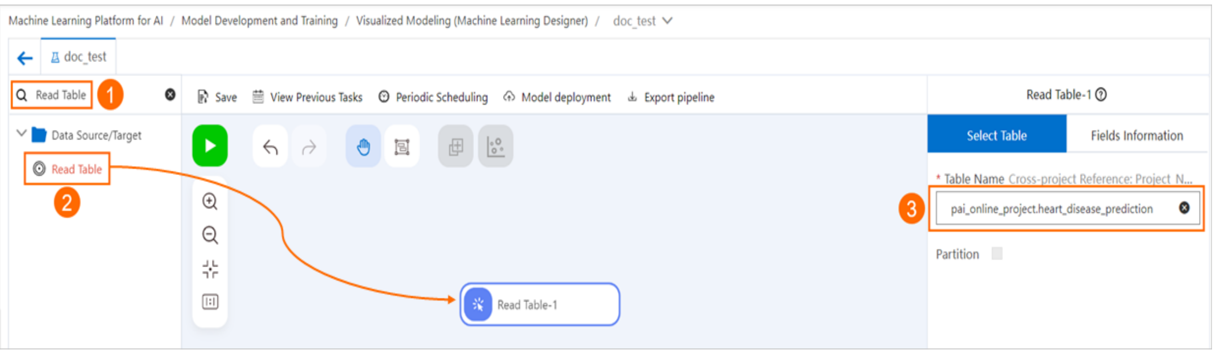
Step 1: Go to the pipeline configuration page

1. Log on to the PAI console. In the left-side navigation pane, click **Workspaces**. On the page that appears, click the name of the workspace that you want to use.
2. In the left-side navigation pane, choose **Model Training > Visualized Modeling (Designer)**. The Visualized Modeling (Machine Learning Designer) page appears.
3. On the Visualized Modeling (Machine Learning Designer) page, select the pipeline that you have created and click **Enter pipeline**.

Step 2: Prepare data

In this example, public data provided by PAI on heart disease cases is used. You can use the **Read Table** component to read the public data without the need to create a table or write data to the table.

> **Note** During your own development, you often need to prepare a table in MaxCompute or Object Storage Service (OSS). Then, you need to use a **Data Source/Target** component such as **Read Table**, **Write Table**, or **Read File Data** to query or write data to the table. For more information, see the topics in Component reference: data source or destination.



* Table Name Cross-project Reference: Project N...

pai_online_project.heart_disease_prediction

1. In the left-side component list, enter a keyword in the search box to search for the **Read Table**

component.

2. Drag the **Read Table** component to the canvas on the right. A pipeline node named **Read Table-1** is automatically generated.
3. Click the **Read Table-1** node. On the Select Table tab in the right-side pane of the canvas, set the **Table Name** parameter to `pai_online_project.heart_disease_prediction` .

   You can click the Fields Information tab to view the details of the fields in the public data.

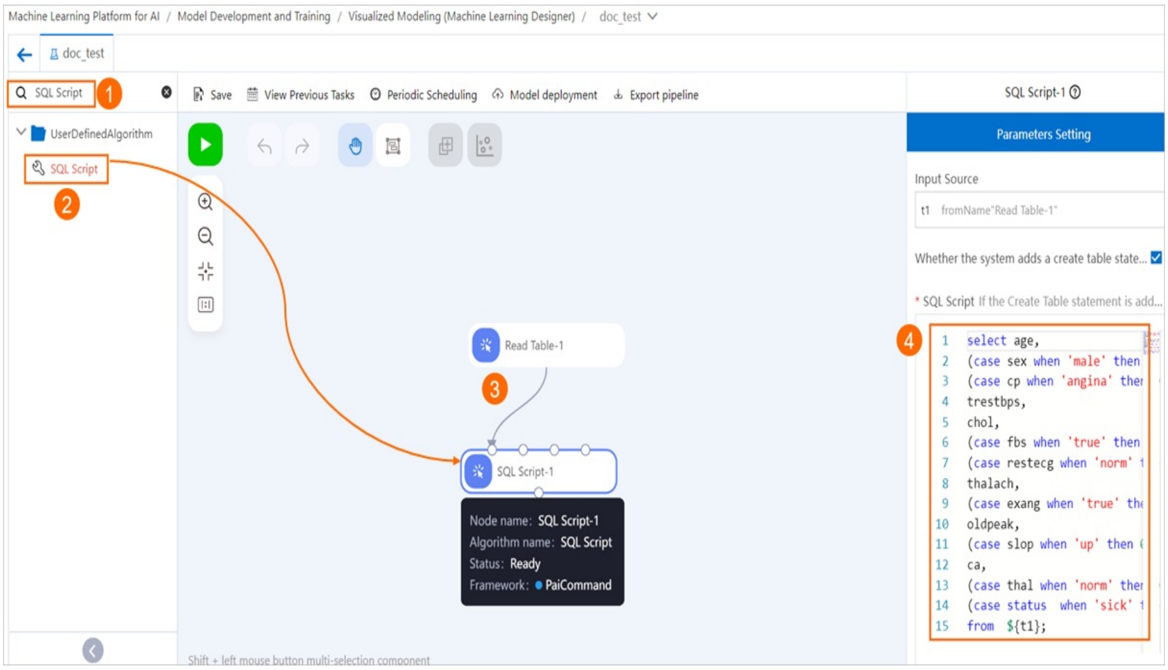
Step 3: Preprocess data

In this example, the public data on heart disease cases is used as raw data, and all field values of the raw data are normalized during preprocessing. To normalize the field values, perform the following steps:

1. Convert all non-numeric fields in the raw data to numeric fields by using an SQL statement. This ensures that all fields are numeric fields after preprocessing.
2. Convert all fields to the DOUBLE type. This ensures that the data type of all fields meets the requirement of normalization.
3. Normalize the values of all fields in the table.

The following section describes the detailed operations.

1. Convert non-numeric fields to numeric fields.



Node name: SQL Script-1
Algorithm name: SQL Script
Status: Ready
Framework: ● PaiCommand

Input Source
t1 fromName"Read Table-1"

Whether the system adds a create table state... ☑

* SQL Script If the Create Table statement is add...

```
select age,
(case sex when 'male' then
(case cp when 'angina' ther
trestbps,
chol,
(case fbs when 'true' then
(case restecg when 'norm' 1
thalach,
(case exang when 'true' the
oldpeak,
(case slop when 'up' then (
ca,
(case thal when 'norm' ther
(case status  when 'sick' 1
from  ${t1};
```

   i. In the left-side component list, enter a keyword in the search box to search for the **SQL Script** component.

   ii. Drag the **SQL Script** component to the canvas on the right. A pipeline node named **SQL Script-1** is automatically generated.

   iii. Draw a line from the **Read Table-1** node to the **SQL Script-1** node. This way, the SQL Script-1 node becomes the downstream node of the Read Table-1 node.

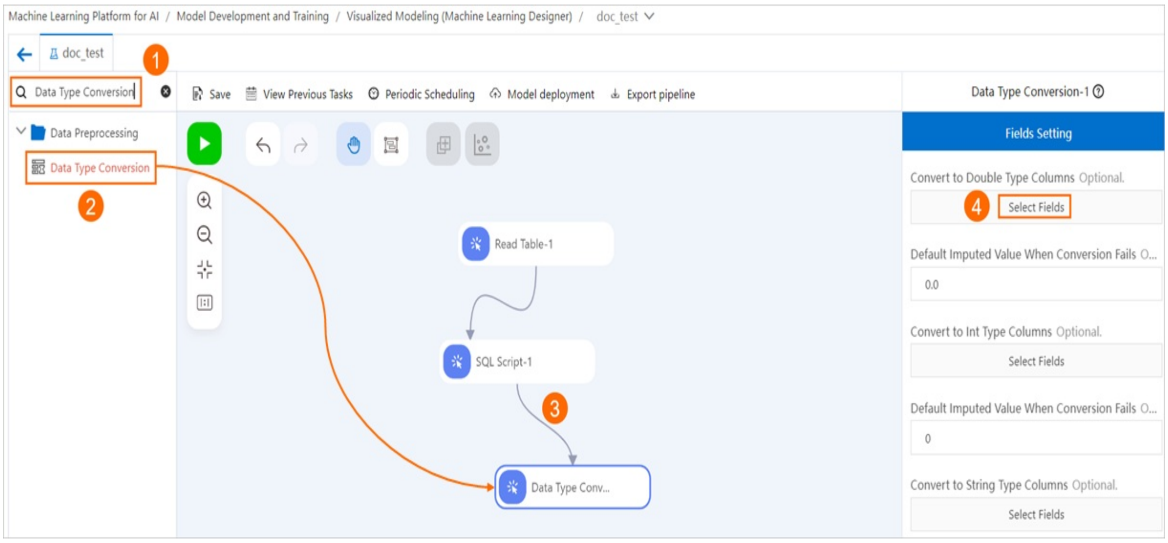
iv. Click the **SQL Script-1** node. On the Parameters Setting tab in the right-side pane, enter the following SQL statement in the SQL Script field.

```
select age,
(case sex when 'male' then 1 else 0 end) as sex,
(case cp when 'angina' then 0  when 'notang' then 1 else 2 end) as cp,
trestbps,
chol,
(case fbs when 'true' then 1 else 0 end) as fbs,
(case restecg when 'norm' then 0  when 'abn' then 1 else 2 end) as restecg,
thalach,
(case exang when 'true' then 1 else 0 end) as exang,
oldpeak,
(case slop when 'up' then 0  when 'flat' then 1 else 2 end) as slop,
ca,
(case thal when 'norm' then 0  when 'fix' then 1 else 2 end) as thal,
(case status  when 'sick' then 1 else 0 end) as status
from  ${t1};
```

> **Note** In the preceding SQL statement, ${*t1*} indicates the name of the source table of the **SQL Script-1** node. Each SQL script node can have four input sources. When you draw a line to an SQL script node, the input port that you select determines the name of the source table. The name of the source table can be t1, t2, t3, or t4. Modify the preceding SQL statement based on the input port that you select.

v. Click the ▶ icon in the upper part of the canvas. The **Read Table-1** and **SQL Script-1** nodes are run in sequence.

2. Convert all fields to the DOUBLE type.



Convert to Double Type Columns Optional.
Default Imputed Value When Conversion Fails O... 0.0
Convert to Int Type Columns Optional.
Default Imputed Value When Conversion Fails O... 0
Convert to String Type Columns Optional.

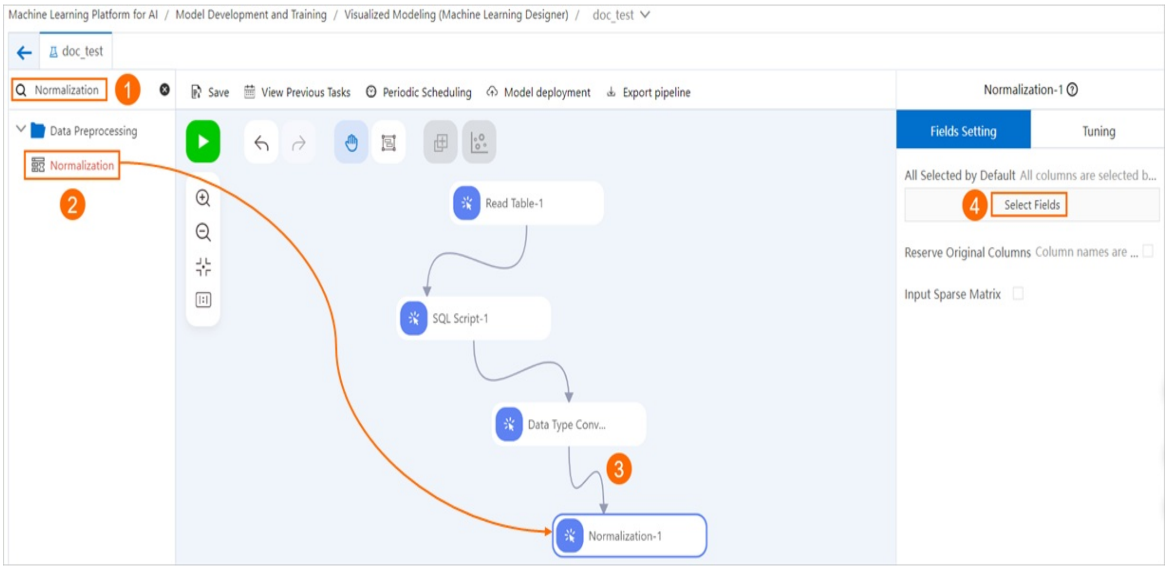
i. In the left-side component list, enter a keyword in the search box to search for the **Data Type Conversion** component.

ii. Drag the **Data Type Conversion** component to the canvas on the right. A pipeline node named **Data Type Conversion-1** is automatically generated.

iii. Draw a line from the **SQL Script-1** node to the **Data Type Conversion-1** node.

iv. Click the **Data Type Conversion-1** component on the canvas. On the **Fields Setting** tab in the right-side pane, click **Select Fields** in the **Convert to Double Type Columns** section and select all fields to convert them to the DOUBLE type.

3. Normalize field values.



i. In the left-side component list, enter a keyword in the search box to search for the **Normalization** component.

ii. Drag the **Normalization** component to the canvas on the right. A pipeline node named **Normalization-1** is automatically generated.

iii. Draw a line from the **Data Type Conversion-1** node to the **Normalization-1** node.

iv. Click the **Normalization-1** component on the canvas. On the **Fields Setting** tab in the right-side pane, click Select Fields and select all fields in the dialog box that appears.

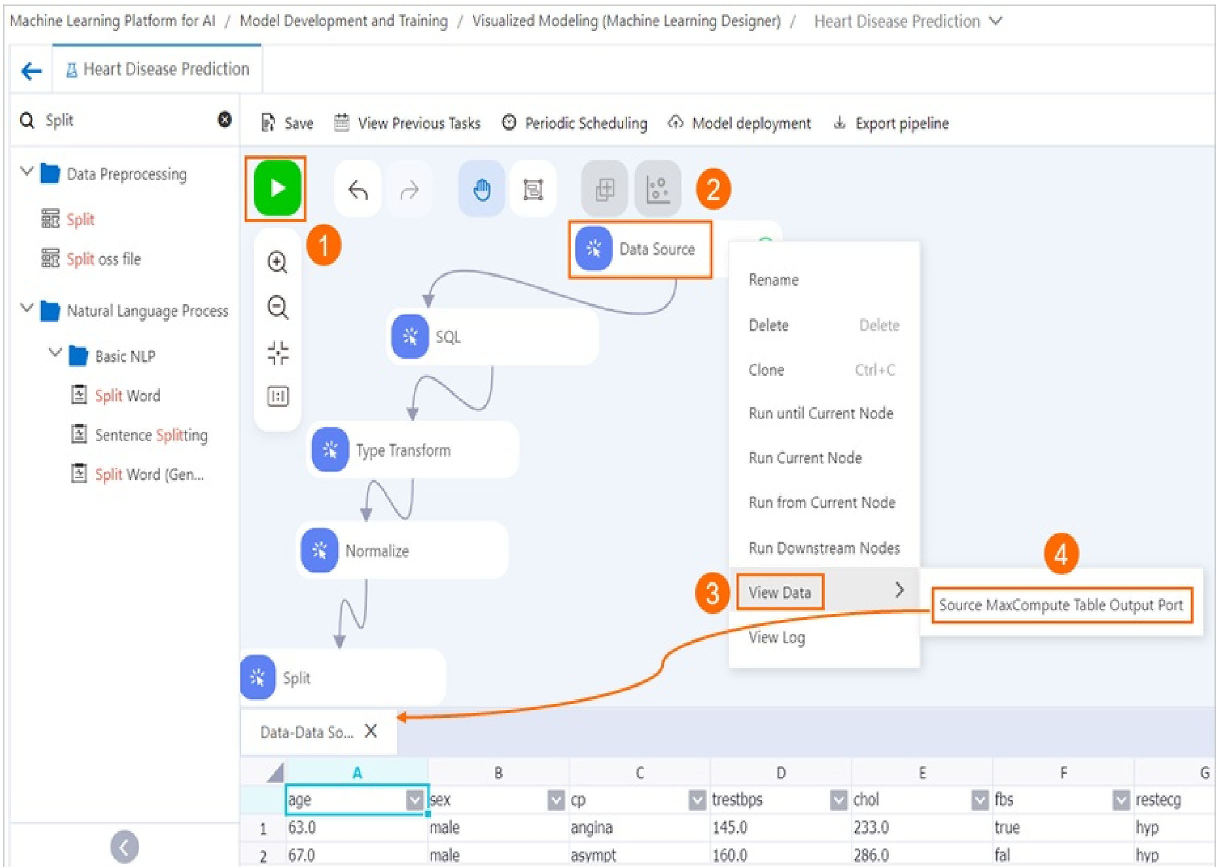
4. In the left-side component list, enter a keyword in the search box to search for the **Split** component. Drag the **Split** component to the canvas on the right. Draw a line from the **Normalization-1** node to the Split-1 node that is generated.

   By default, the **Split** component splits the raw data into a model training set and a model prediction set at a ratio of 4:1. To change the ratio, you can click the **Split** component and set the **Splitting Fraction** parameter on the **Parameters Setting** tab.

5. In the top toolbar of the canvas, click **Save**.

Debug and run the pipeline

In the upper part of the canvas, click the ▶ icon.



	A	B	C	D	E	F	G
	age	sex	cp	trestbps	chol	fbs	restecg
1	63.0	male	angina	145.0	233.0	true	hyp
2	67.0	male	asympt	160.0	286.0	fal	hyp

After you click the Run icon, all nodes are run in sequence. After a node is successfully run, the node is marked with a ✓ icon in the node box. You can right-click a successful node and select **View Data** to check whether the output data is correct.

> **Note** If the pipeline is complex, you can save and run the pipeline every time after you add a node to the pipeline. If a node fails to run, you can right-click the node and select View Log to troubleshoot the failure.

What's next

After the data is preprocessed, you need to visualize the data. For more information, see Data visualization.

2.3. Data visualization

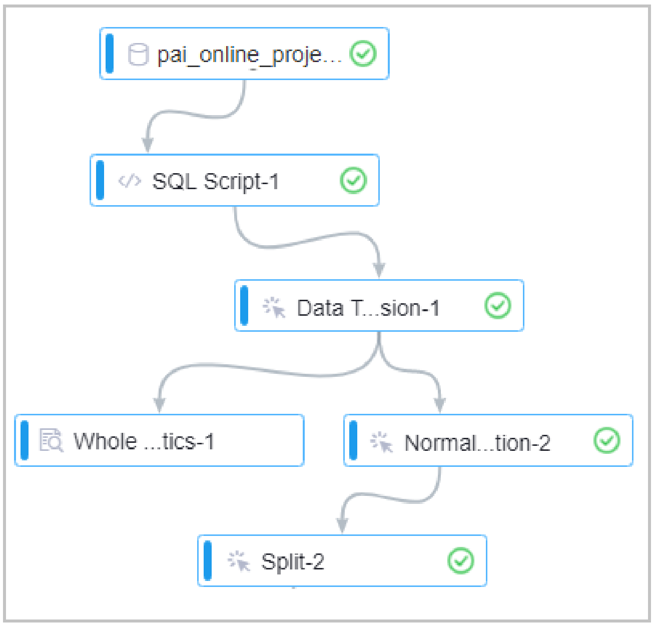
In this topic, the whole table statistics component is used as an example to describe how to visualize data.

Prerequisites

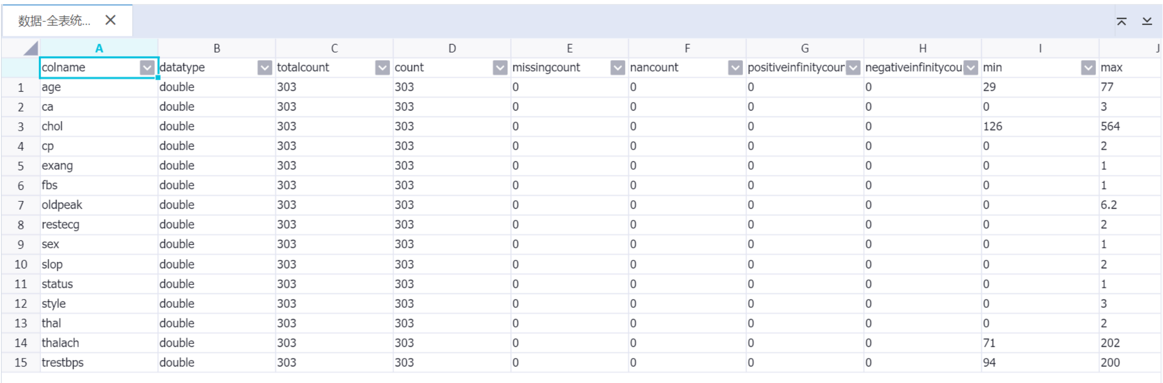
Data preprocessing is complete. For more information, see Prepare and preprocess data.

Procedure

1.
2.
3.
4. In the left-side navigation pane, click **Components**.
5. In the Components list, click **Statistical Analysis**. Then, drag and drop the **Whole Table Statistics** component onto the canvas, and connect it to the components prepared during Prepare and preprocess data. The following figure shows how to connect the components.



6. Right-click the **Whole Table Statistics** component on the canvas. In the menu that appears, click **Run This Node**.
7. After the experiment stops running, right-click the **Whole Table Statistics** component. In the menu that appears, click **View Data** to view statistics of the whole table.



	colname	datatype	totalcount	count	missingcount	nancount	positiveinfinitycour	negativeinfinitycou	min	max
1	age	double	303	303	0	0	0	0	29	77
2	ca	double	303	303	0	0	0	0	0	3
3	chol	double	303	303	0	0	0	0	126	564
4	cp	double	303	303	0	0	0	0	0	2
5	exang	double	303	303	0	0	0	0	0	1
6	fbs	double	303	303	0	0	0	0	0	1
7	oldpeak	double	303	303	0	0	0	0	0	6.2
8	restecg	double	303	303	0	0	0	0	0	2
9	sex	double	303	303	0	0	0	0	0	1
10	slop	double	303	303	0	0	0	0	0	2
11	status	double	303	303	0	0	0	0	0	1
12	style	double	303	303	0	0	0	0	0	3
13	thal	double	303	303	0	0	0	0	0	2
14	thalach	double	303	303	0	0	0	0	71	202
15	trestbps	double	303	303	0	0	0	0	94	200

What's next

After data visualization is complete, you can start algorithm modeling. For more information, see Algorithm modeling.

2.4. Algorithm modeling

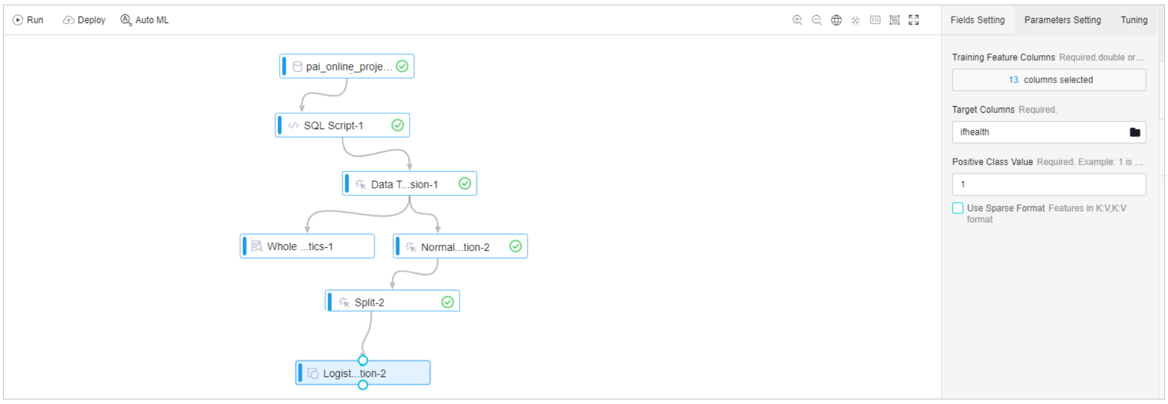
In this topic, the logistic regression for binary classification algorithm is used as an example to describe how to generate models in Machine Learning Platform for AI.

Prerequisites

Data visualization is complete. For more information, see Data visualization.

Procedure

1.
2.
3.
4. In the left-side navigation pane, click **Components**.
5. In the Components list, choose **Machine Learning > Binary Classification**. Then, drag and drop the **Logistic Regression for Binary Classification** component onto the canvas, and connect it to the components prepared during Data visualization.
6. Click the **Logistic Regression for Binary Classification** component on the canvas. On the uicontrol **Fields Setting** tab on the right side, set **Target Columns** to **ifhealth**. In the **Training Feature Columns** section, select all columns except for the **Target Columns**, as shown in the following figure.



Training Feature Columns Required.double or...

13 columns selected

Target Columns Required.

ifhealth

Positive Class Value Required. Example: 1 is ...

1

□ Use Sparse Format Features in K:V,K:V format

7. Right-click the **Logistic Regression for Binary Classification** component. In the menu that appears, click **Run This Node**.

What's next

After algorithm modeling is complete. You can evaluate the model. For more information, see Model evaluation.

2.5. Evaluate the model

This topic describes how to evaluate models by using a binary classification model.

Prerequisites

Algorithm modeling is complete. For more information, see Algorithm modeling.

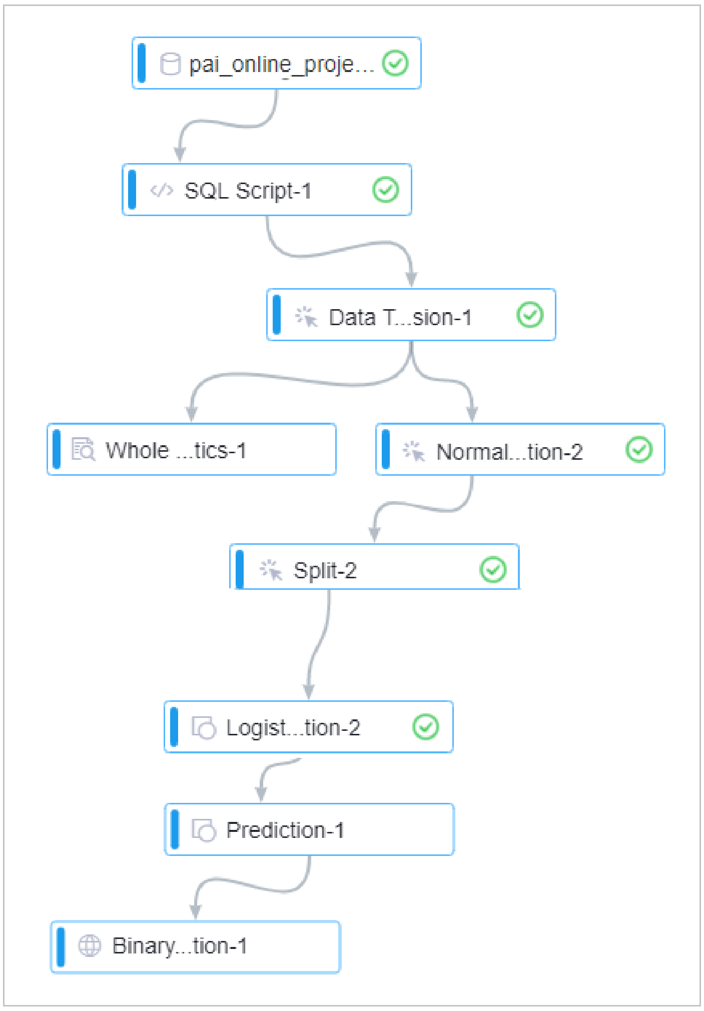
Procedure

1.
2. Drag components to the canvas to create an experiment.

i. In the left-side navigation pane, click **Components**.

ii. In the Components pane, choose **Machine Learning > Recommendation**. Then, drag the **Prediction** component to the canvas.

iii. In the Components pane, choose **Machine Learning > Evaluation**. Then, drag the **Binary Classification Evaluation** component to the canvas and connect it to the components prepared during algorithm modeling. For more information, see Algorithm modeling. The following figure shows how to connect the components.



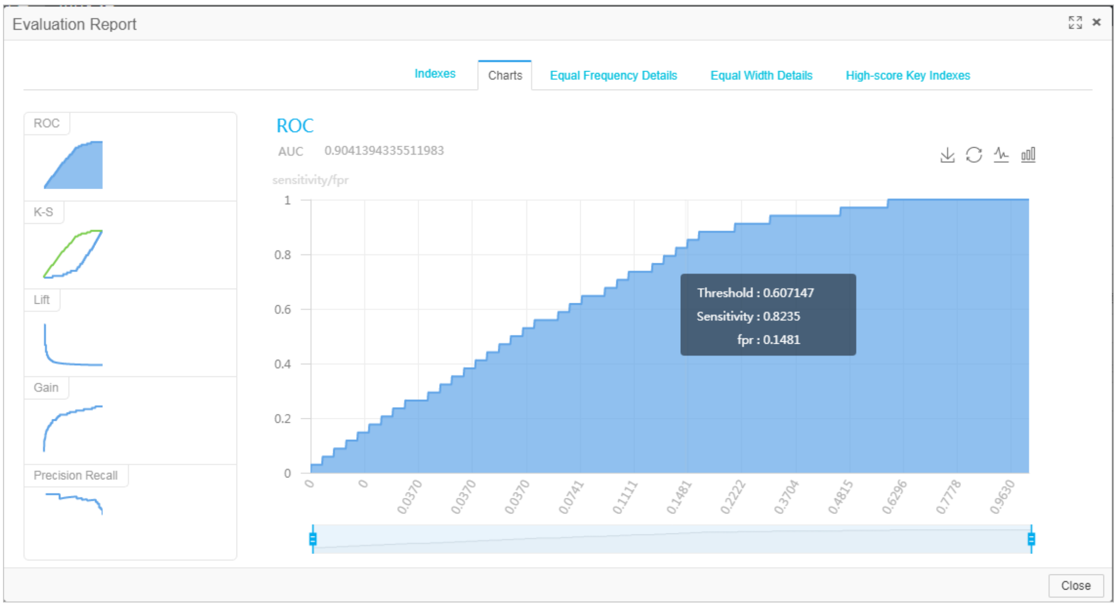
iv. Click the **Binary Classification Evaluation** component on the canvas. On the **Fields Setting** tab on the right side, set the **Original Label Column** parameter to **ifhealth**.

3. In the upper part of the canvas, click **Run**.

4. View the model evaluation report.

i. After the experiment stops running, right-click the **Binary Classification Evaluation** component. In the shortcut menu that appears, click **View Evaluation Report**.

ii. Click the **Charts** tab, and view the receiver operating characteristic curve (ROC) of the binary classification model with different parameters.



AUC 0.9041394335511983

Threshold : 0.607147
Sensitivity : 0.8235
fpr : 0.1481

3.Create experiments based on templates

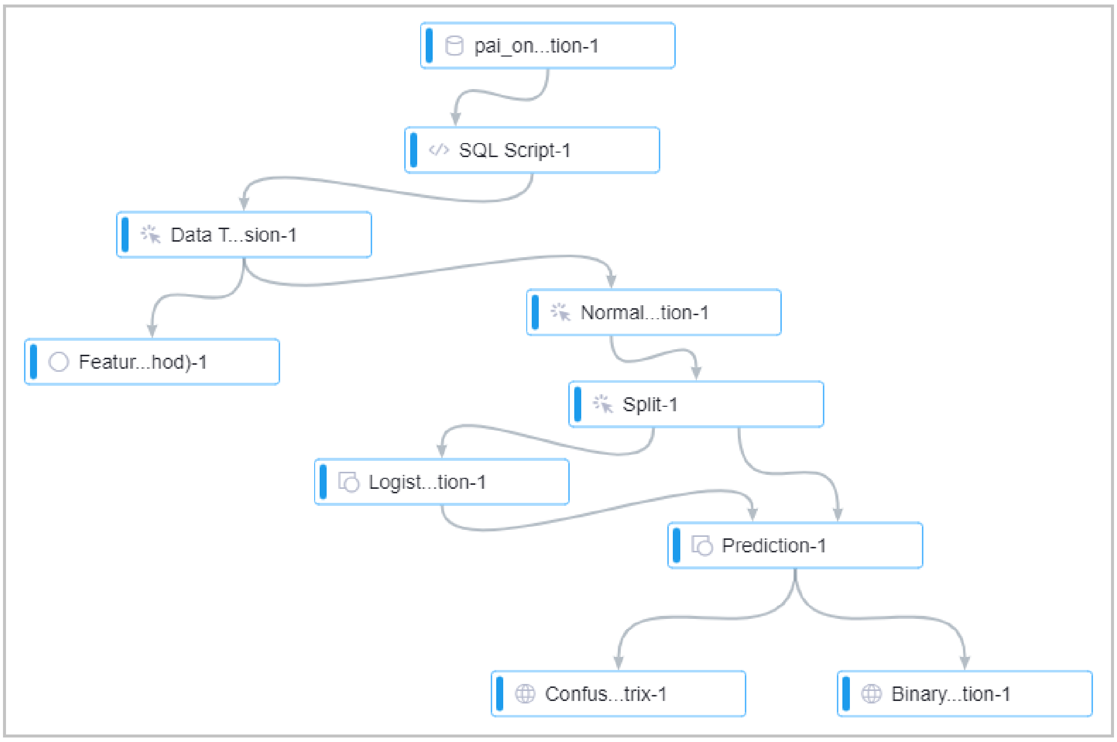
This topic describes how to create experiments and deploy experiment models by using the heart disease prediction template.

Prerequisites

- Machine Learning Platform for AI is activated. For more information, see Purchase.
- A project is created. For more information, see Create a project.

Procedure

1.
2. Create and run an experiment.
   i. In the left-side navigation pane, click **Home**.
   ii. In the **Templates** section, click **Create** below **Heart Disease Prediction**.
   iii. In the **New Experiment** dialog box, set the **Name** parameter and use the default values for other parameters.
   iv. Click **OK**. Wait about 10 seconds for the canvas of the experiment to appear. The following figure shows the canvas.



   v. In the upper part of the canvas, click **Run**. When the experiment is running, you can right-click the components to view their output information.
3. Deploy the model.
   i. After the experiment stops running, move the pointer over **Deploy** and select **Online Model Service**.

ii. Click **Next**.

iii. In the **Resources And Models** panel, set the **Custom Model Name** parameter and use the default values for other parameters.

iv. Click **Deploy**.

v. When the status of the model changes from **Creating** to **Running** in the **State** column, the model is deployed.

> ? **Note** When you do not use the model, click **Stop** in the **Operating** column. This saves costs.