

Hafizah Ab Rahim

Microsoft Azure Machine Learning

No Code Binary Classification Model

August 10, 2020

Introduction

Azure Machine Learning is a **cloud-based platform** for building and operating machine learning solutions in Azure. It includes a wide range of features and capabilities that help data scientists **prepare data, train models, publish predictive services**, and **monitor their usage**.

One of these features is a **visual interface** called **Designer** that you can use to train, test, and deploy machine learning models **without writing any code**.

Objective

- Use Microsoft Azure to **create a binary classification model** using a home loan dataset without any code.
- Create a model that determines **who qualifies for a home loan** based on different historical features.

What is Binary Classification?

- Classification is a type of **supervised** machine learning technique. It predicts which class or category a feature belongs to.
- Classification is considered binary when there are **two class labels**.
- The dataset used has two class labels – **Yes (1) and No (0)**, where Yes means an applicant qualifies for a home loan and No means they do not.

Create Azure ML Workspace

To use Azure Machine Learning, you create a *workspace* in your Azure subscription. You can then use this workspace to manage data, compute resources, code, models, and other artifacts related to your machine learning workloads.

Steps

1. Go to "portal.azure.com"

2. Click "Create a resource"

Previous resources created will be shown here

The screenshot shows the Azure portal homepage. An orange arrow points from the first step to the browser address bar. Another orange arrow points from the second step to the 'Create a resource' button, which is highlighted with a dashed blue box. A third orange arrow points from the 'Previous resources created will be shown here' text to the 'Recent resources' table.

portal.azure.com

Search resources, services, and docs (G+)

Azure services

[Create a resource](#)

Virtual machines App Services Storage accounts SQL databases Azure Database for PostgreSQL Azure Cosmos DB Kubernetes services Function App [More services](#)

Recent resources

Name	Type	Last Viewed
Azure subscription 1	Subscription	a minute ago
Workspace_new	Machine Learning	14 hours ago

Navigate

Subscriptions Resource groups All resources Dashboard

Tools

[Microsoft Learn](#) Learn Azure with free online training from Microsoft

[Azure Monitor](#) Monitor your apps and infrastructure

[Security Center](#) Secure your apps and infrastructure

[Cost Management](#) Analyze and optimize your cloud spend for free

Useful links

[Technical Documentation](#) [Azure Migration Tools](#)

[Azure Services](#) [Find an Azure expert](#)

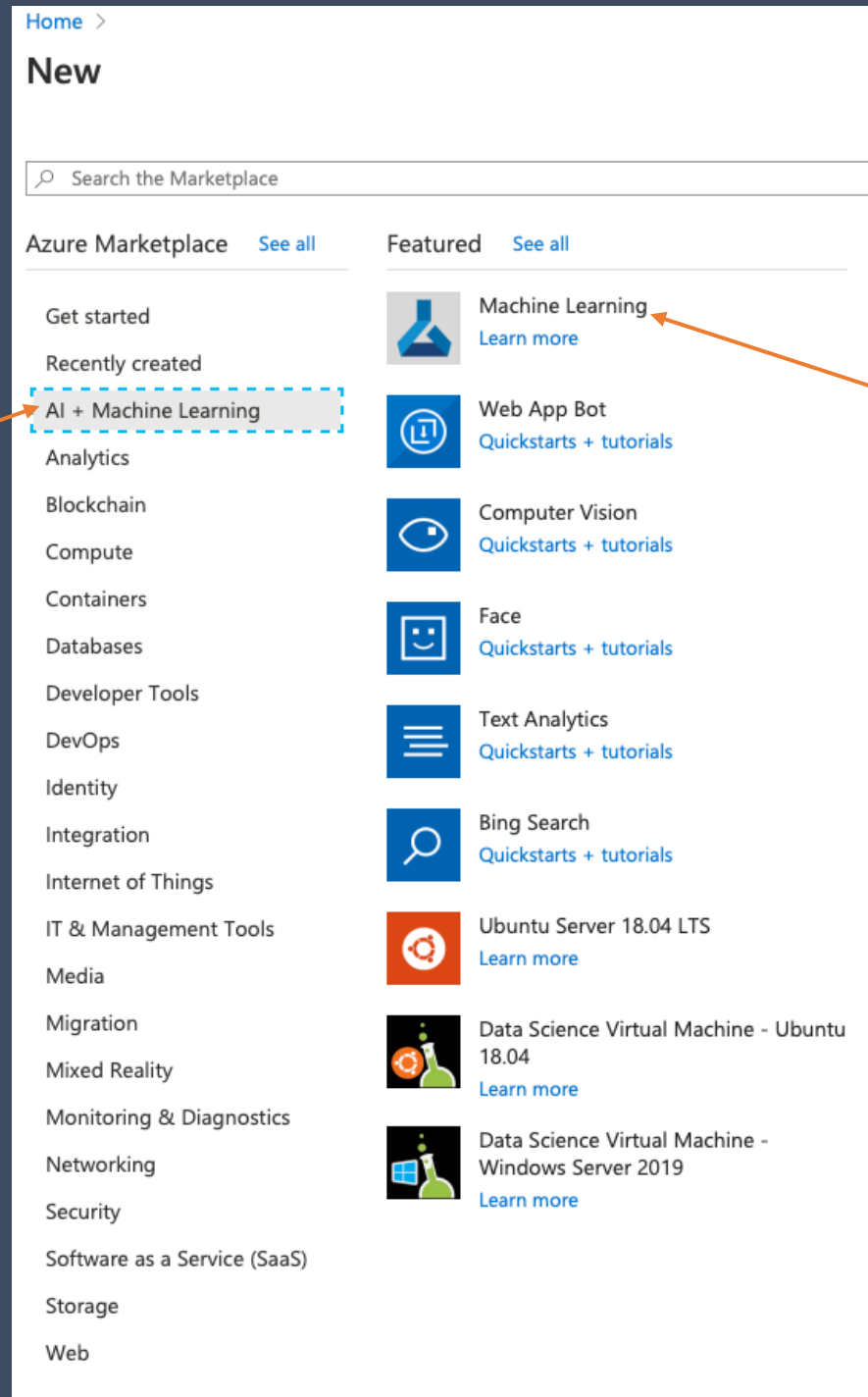
[Recent Azure Updates](#) [Quickstart Center](#)

Azure mobile app

Download on the App Store GET IT ON Google Play

Steps

3. Click "AI + Machine Learning"



4. Click "Machine Learning"

Steps

5. Create a new resource group & name your workspace

6. Choose "Enterprise" & then click "Review + create"

7. Click "Go to resource" after you receive green check symbol.

Home > Microsoft.MachineLearningServices | Overview

Deployment

Search (Cmd+ /)

Overview

Inputs

Outputs

Template

Your deployment is complete

Deployment name: Microsoft.MachineLearningServices
Subscription: [Azure subscription 1](#)
Resource group: [Finance](#)

Deployment details (Download)

Next steps

[Go to resource](#)

Machine Learning

Create a machine learning workspace

Validation passed

Basics Tags Review + create

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *

Azure subscription 1

Resource group *

(New) Finance-RG

[Create new](#)

Workspace details

Specify the name, region, and edition for the workspace.

Workspace name *

Finance-WS

Region *

East US

Workspace edition *

Enterprise

For your convenience, these resources are added automatically to the workspace, if regionally available: [Azure Storage](#), [Azure Application Insights](#), [Azure Key Vault](#)

[Review + create](#)

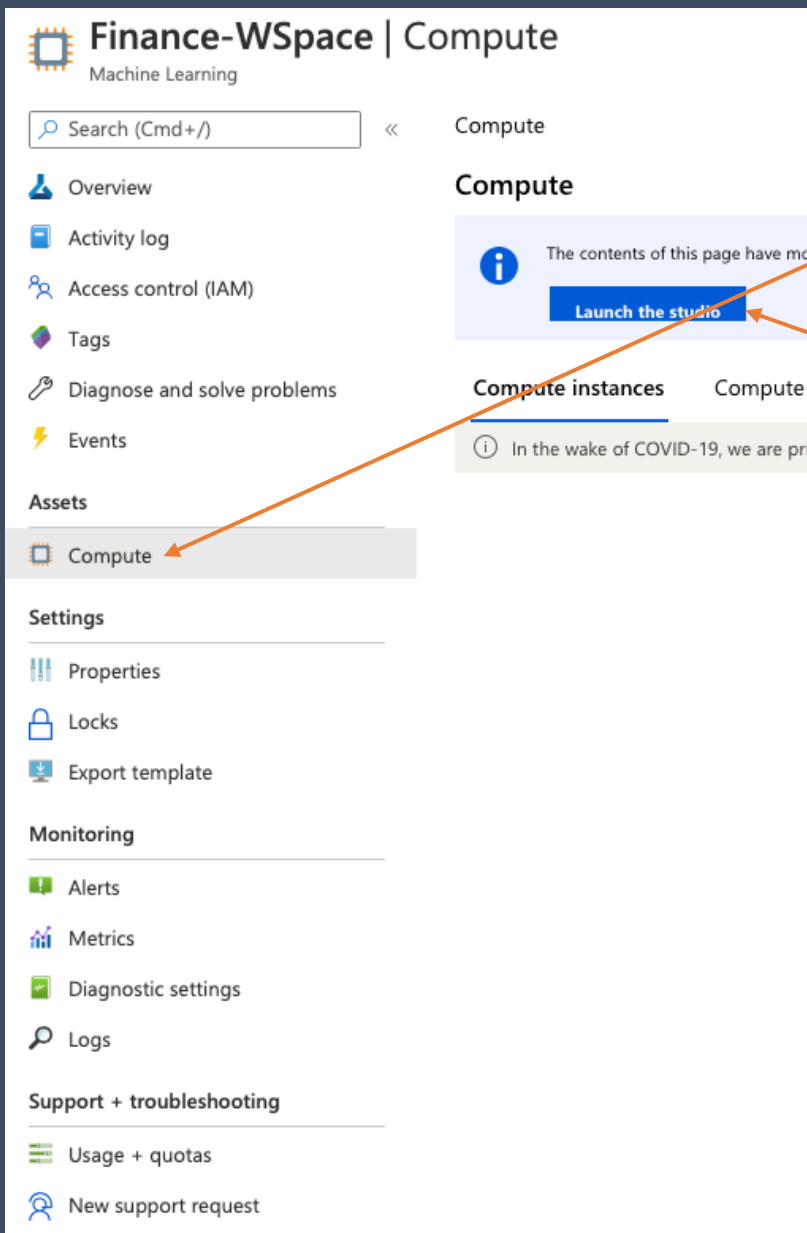
[< Previous](#)

[Next : Tags](#)

Create Compute Resources

To train and deploy models using Azure Machine Learning designer, you need compute on which to run the training process, test the model, and host the model in a deployed service.

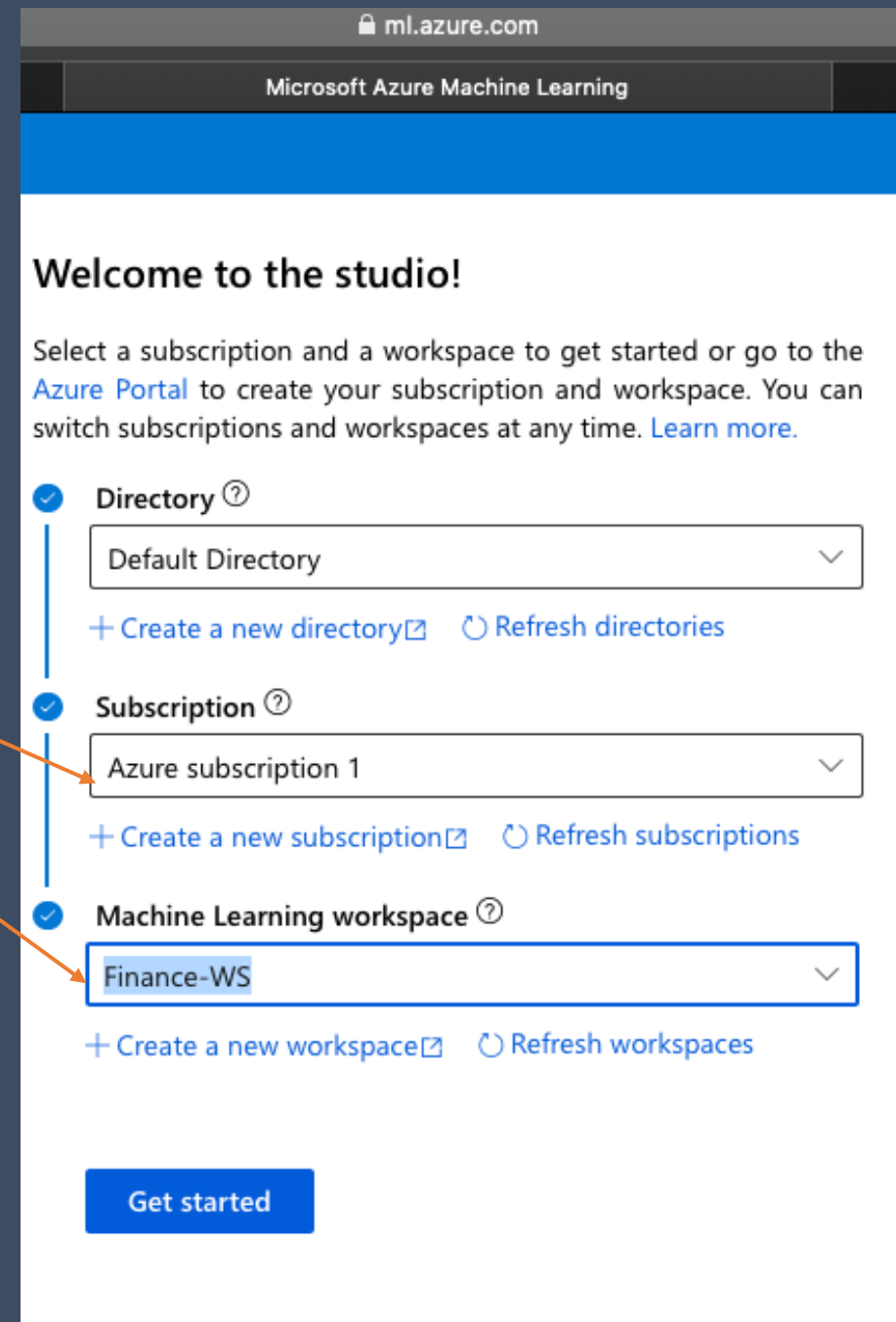
Steps



8. Click "Compute"

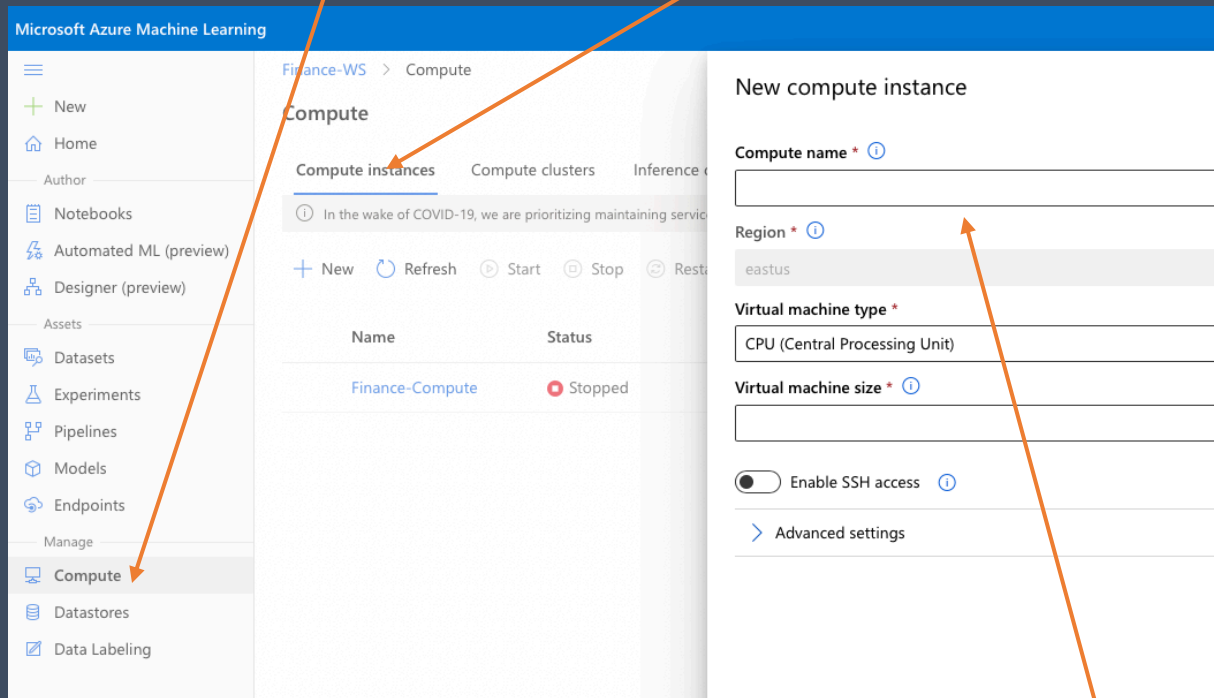
9. Click "Launch the studio"

10. Once directed to the new site, choose your subscription & choose the workspace you have just created.

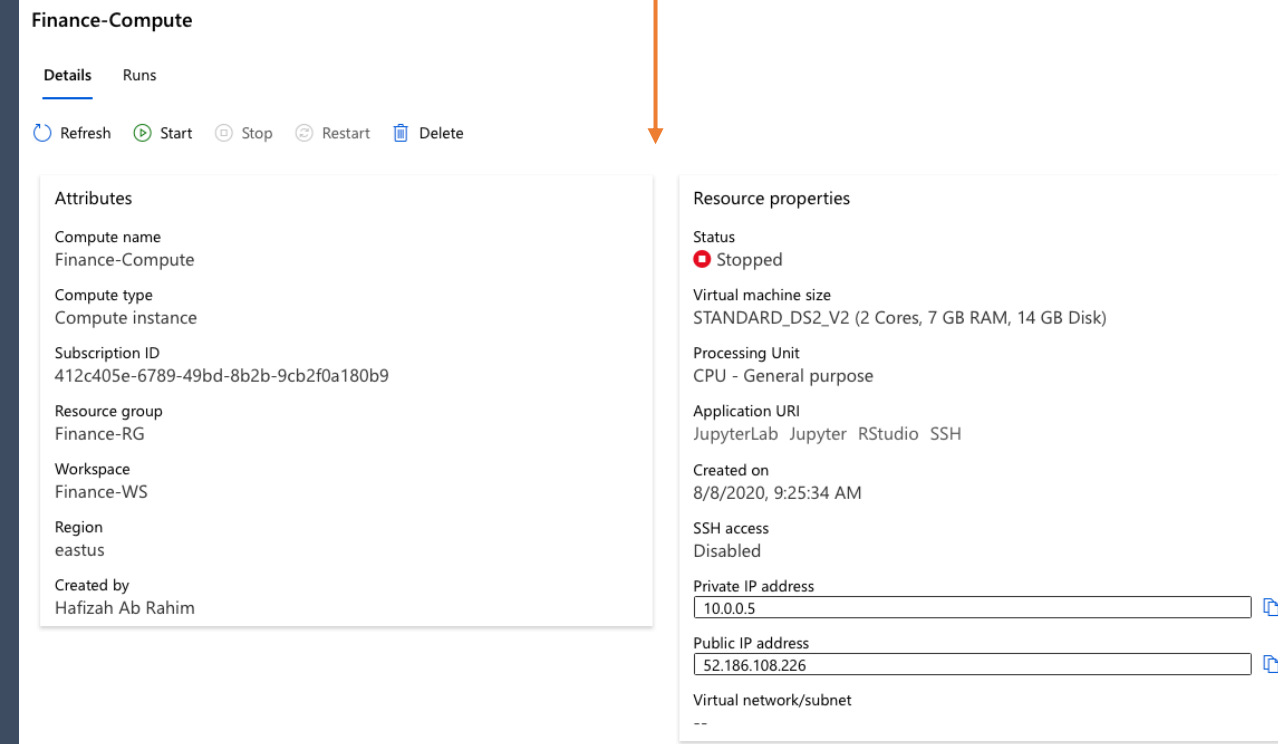


Steps

11. Click "Compute" then "Compute instances"



Once details are entered, the page below will display



On the **Compute Instances** tab, add a new compute instance with the following settings. You'll use this to test your model:

- **Compute name:** enter a unique name
- **Virtual Machine type:** CPU
- **Virtual Machine size:** Standard_DS2_v2

12. Complete the following using this information.

Steps

13. Click "Compute" then "Compute clusters"

14. Complete the following using this information.

The screenshot shows the Microsoft Azure Machine Learning interface. On the left is a navigation pane with a 'Compute' section highlighted. The main area shows the 'Compute clusters' tab, with a table listing a cluster named 'Finance-CompClus' in a 'Succeeded' state. An overlay form titled 'New compute cluster' is open on the right, with orange arrows pointing from the text instructions to specific fields: 'Compute clusters' in the navigation pane, the 'Compute clusters' tab, and the 'Compute name' field.

Name	Provisioning state
Finance-CompClus	Succeeded (0 nodes)

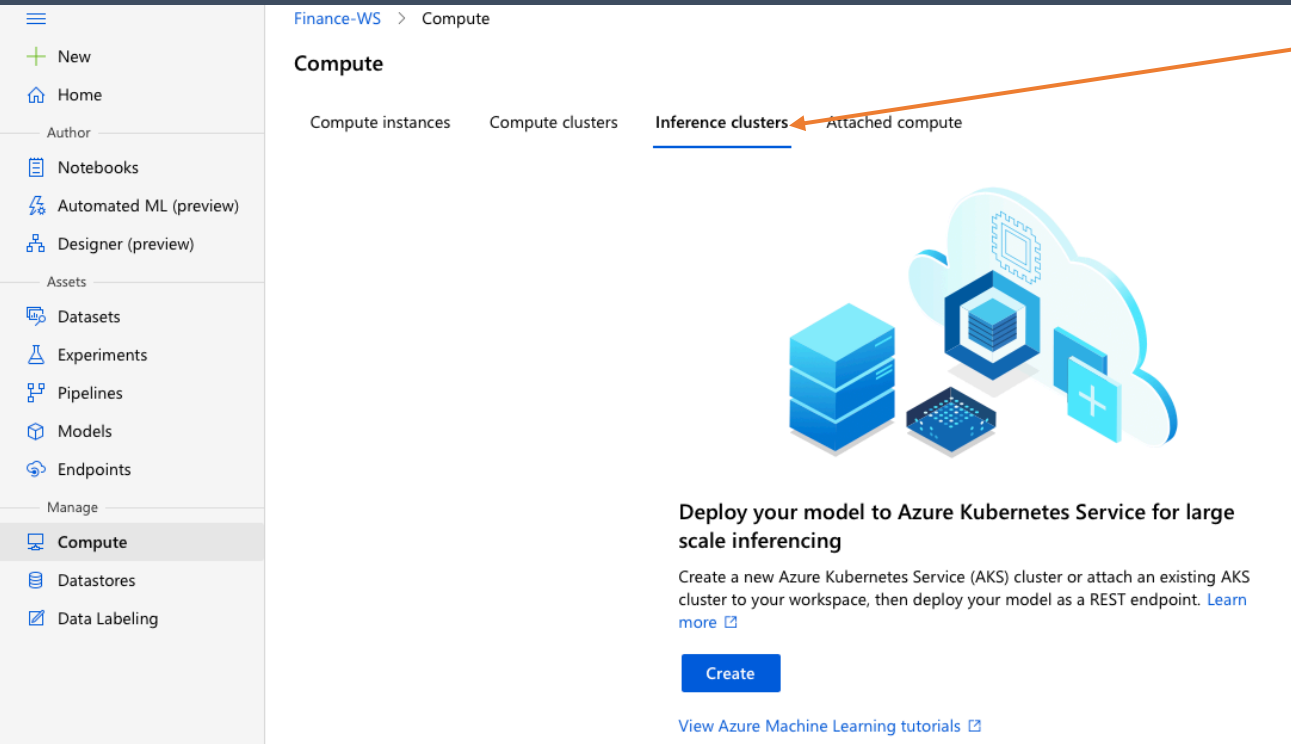
New compute cluster

- Compute name *
- Region *
- Virtual machine type *
- Virtual machine priority * ☒ Dedicated ☐ Low priority
- Virtual machine size *
- Minimum number of nodes *
- Maximum number of nodes *
- Idle seconds before scale down *
- [Advanced settings](#)

While the compute instance is being created, switch to the **Compute Clusters** tab, and add a new compute cluster with the following settings. You'll use this to train a machine learning model:

- **Compute name:** enter a unique name
- **Virtual Machine size:** Standard_DS2_v2
- **Virtual Machine priority:** Dedicated
- **Minimum number of nodes:** 2
- **Maximum number of nodes:** 2
- **Idle seconds before scale down:** 120

Steps



15. Click "Compute" then "Inference clusters"

16. Complete the following using this information.

New inference cluster

Compute name * ⓘ

InfCLUS

Kubernetes Service

[Create new](#) [Use existing](#)

Region *

East US

Virtual machine size * ⓘ

Standard_DS2_v2 2 Cores, 7 GB (RAM), 14 GB (Disk)

Cluster purpose

☐ Production ☒ Dev-test

Number of nodes * ⓘ

2

Network configuration ⓘ

[Basic](#) [Advanced](#)

☐ Enable SSL configuration ⓘ

While the compute cluster is being created, switch to the **Inference Clusters** tab, and add a new cluster with the following settings. You'll use this to deploy your model as a service.

- **Compute name:** enter a unique name
- **Kubernetes Service:** Create new
- **Region:** Select a different region than the one used for your workspace
- **Virtual Machine size:** Standard_DS2_v2 (Use the filter to find this in the list)
- **Cluster purpose:** Dev-test
- **Number of nodes:** 2
- **Network configuration:** Basic
- **Enable SSL configuration:** Unselected

Compute

Compute instances Compute clusters Inference clusters Attached compute

ⓘ In the wake of COVID-19, we are prioritizing maintaining service availability for first responders, health and emergency management services, critical government infrastructure, and existing paid customers.

+ New Refresh Start Stop Restart Delete ☒ Show created by me only

Name	Status	Application URI	Virtual machine size	Created on ↓
Finance-Compute	Running	JupyterLab Jupyter RStudio SSH	STANDARD_DS2_V2	Aug 8, 2020 9:25 AM

Make sure all the three types of "Compute" are ready!

On the right shows the state when "compute" is ready.

Compute

Compute instances **Compute clusters** Inference clusters Attached compute

ⓘ In the wake of COVID-19, we are prioritizing maintaining service availability for first responders, health and emergency management services, critical government infrastructure, and existing paid customers.

+ New Refresh Delete

Name	Provisioning state	Virtual machine size	Created on ↓
Finance-CompClus	Succeeded (2 nodes)	STANDARD_DS2_V2	Aug 8, 2020 9:48 AM

Compute

Compute instances Compute clusters **Inference clusters** Attached compute

+ New Refresh Delete Detach

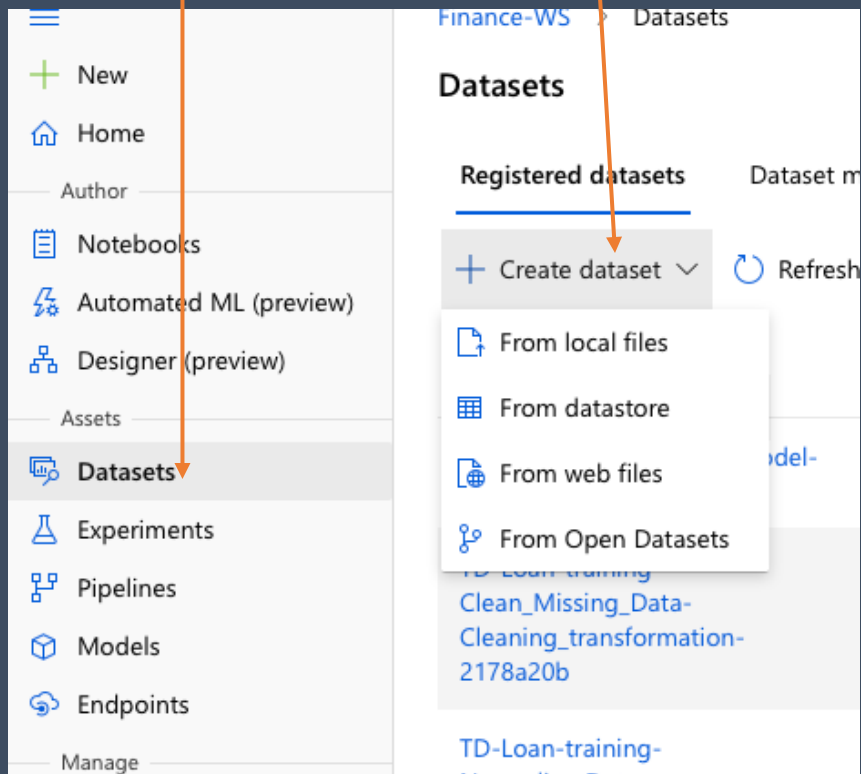
Name	Type	Created/Attached	Provisioning state	Created on ↓
InfCLUS	Kubernetes service	Created	Succeeded	Aug 9, 2020 10:03 AM

Explore Data

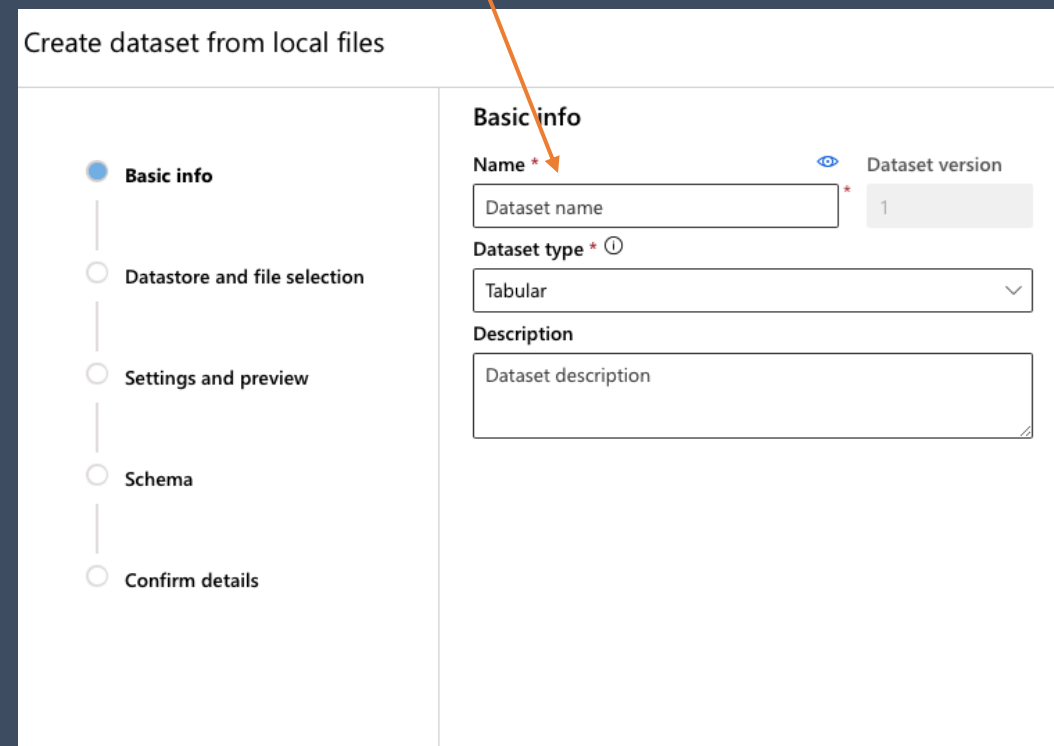
To train a classification model, you need a dataset that includes historical *features* (characteristics of the entity for which you want to make a prediction) and known *label* values (the class indicator that you want to train a model to predict).

Steps

17. Click “Datasets” then “Create dataset”. There are 4 ways for you to upload your data



18. Name your dataset.



Steps

19. Follow these steps. Fill up the information as shown.

Create dataset from local files

Basic info

Datastore and file selection

Settings and preview

Schema

Confirm details

Datastore and file selection

Select or create a datastore *

Currently selected datastore: workspaceblobstore (Azure Blob Storage) (Default)

Previously created datastore

Create new datastore

Select files for your dataset *

After dataset creation, these files will be uploaded to your default Blob storage and made available in your workspace. Supported file types include: delimited (i.e. csv, tsv), Parquet, JSON Lines, and plain text.

Browse

1 files selected. Total size 0.03625 MiB. 0/1 files uploaded

File name	Size (MiB)	Upload %	Status
Loan_Train.csv	0.03625		

Upload path

UI Files will be uploaded to '\$(Upload path)/08-09-2020_020717.UTC'

☐ Skip data validation ⓘ

Create dataset from local files

Basic info

Datastore and file selection

Settings and preview

Schema

Confirm details

Settings and preview

These settings were automatically detected. Please verify that the selections were made correctly or update

File format

Delimited

Delimiter

Comma

Example

Field1,Field2,Field3

Encoding

UTF-8

Column headers

Use headers from the first file

Skip rows

None

Id	Loan_ID	Gender	Married	Dependent
1	LP001002	Male	false	0
2	LP001003	Male	true	1
3	LP001005	Male	true	0
4	LP001006	Male	true	0
5	LP001008	Male	false	0
6	LP001011	Male	true	2
7	LP001013	Male	true	0
8	LP001014	Male	true	3+
9	LP001018	Male	true	2

Steps

20. Choose columns that you want to be included in the table.
Check to make sure the all data type are correct.

Basic info

Datastore and file selection

Settings and preview

Schema

Confirm details

Include	Column name	Properties	Type	Format settings and example
<input checked="" type="checkbox"/>	Gender	Not applicable to selected type	String	Male, Male, Male
<input checked="" type="checkbox"/>	Married	Not applicable to selected type	Boolean	false, true, true
<input checked="" type="checkbox"/>	Dependents	Not applicable to selected type	String	0, 1, 0
<input checked="" type="checkbox"/>	Education	Not applicable to selected type	String	Graduate, Graduate, Graduate
<input checked="" type="checkbox"/>	Self_Employed	Not applicable to selected type	Boolean	false, false, true
<input checked="" type="checkbox"/>	ApplicantIncome	Not applicable to selected type	Integer	5849, 4583, 3000
<input checked="" type="checkbox"/>	CoapplicantIncome	Not applicable to selected type	Integer	0, 1508, 0

Steps

21. Check all the details. Click “Create”.

The screenshot shows a multi-step wizard for creating a dataset. On the left, a vertical navigation pane lists five steps: 'Basic info', 'Datastore and file selection', 'Settings and preview', 'Schema', and 'Confirm details'. The 'Confirm details' step is currently selected and highlighted with a blue circle. The main content area is titled 'Confirm details' and is divided into three sections. The 'Basic info' section contains fields for 'Name' (Loan-Train), 'Dataset version' (1), and 'Dataset type' (Tabular). The 'Datastore and file selection' section contains fields for 'Datastore' (workspaceblobstore), 'Selected files (1)' (Loan_Train.csv), and 'Path' (UI/08-09-2020_020717.UTC/Loan_Train.csv). The 'File settings' section contains fields for 'File format' (Delimited), 'Delimiter' (Comma), 'Encoding' (UTF-8), 'Column headers' (Use headers from the first file), and 'Skip rows' (None). At the bottom of the main content area, there is a checkbox labeled 'Profile this dataset after creation' which is currently unchecked. At the bottom of the wizard, there are three buttons: 'Back', 'Create' (highlighted in blue), and 'Cancel'.

Confirm details

Basic info

Name
Loan-Train

Dataset version
1

Dataset type
Tabular

Datastore and file selection

Datastore
workspaceblobstore

Selected files (1)
Loan_Train.csv

Path
UI/08-09-2020_020717.UTC/Loan_Train.csv

File settings

File format
Delimited

Delimiter
Comma

Encoding
UTF-8

Column headers
Use headers from the first file

Skip rows
None

☐ Profile this dataset after creation

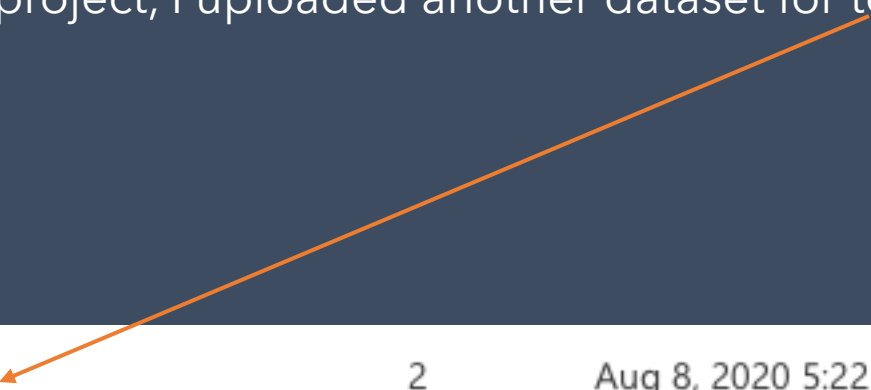
Navigation:

- Basic info
- Datastore and file selection
- Settings and preview
- Schema
- Confirm details**

Buttons: Back, Create, Cancel

Steps

22. Repeat steps 17- 21 when you want to upload another dataset.
In this project, I uploaded another dataset for testing.



Loan-Test	2	Aug 8, 2020 5:22 PM	Aug 8, 2020 7:58 PM	Tabular	Hafizah Ab Rahim
Loan	2	Aug 8, 2020 5:20 PM	Aug 8, 2020 5:58 PM	Tabular	Hafizah Ab Rahim

Creating a Pipeline

To get started with Azure Machine Learning designer, first you must create a pipeline and add the dataset you want to work with.

Steps

23. Click "Designer (preview)" and then "+"

Microsoft Azure Machine Learning

Finance-WS > Designer (preview)

Designer (preview)

New pipeline

Easy-to-use prebuilt modules ?

Sample 1: Regression - Automobile Price Prediction... ?

Sample 2: Classification - Automobile Price Prediction... ?

Assets

- Datasets
- Experiments
- Pipelines

Steps

24. Rename your training pipeline and choose your compute target.

The screenshot displays the Microsoft Azure Machine Learning Designer interface. The left sidebar shows the navigation menu with options like New, Home, Author, Notebooks, Automated ML (preview), Designer (preview), Assets, Datasets, Experiments, Pipelines, Models, Endpoints, Manage, Compute, Datastores, and Data Labeling. The main workspace is titled 'Finance-WS > Designer (preview) > Authoring' and shows a pipeline named 'HomeLoan-Train'. The 'Modules' panel on the left lists various categories such as Data Input and Output, Data Transformation, Feature Selection, Statistical Functions, Machine Learning Algorithms, Model Training, Model Scoring & Evaluation, Python Language, R Language, Text Analytics, Computer Vision, Recommendation, Anomaly Detection, and Web Service. The 'Canvas' area is highlighted with a yellow border and contains the text 'Canvas'. The 'Settings' panel on the right shows the 'Default compute target' set to 'Finance-CompClus' and the 'Draft name' set to 'HomeLoan-Train'. The 'Draft details' section shows the draft was created on August 9, 2020, at 10:21 AM by Hafizah Ab Rahim.

Microsoft Azure Machine Learning

Finance-WS > Designer (preview) > Authoring

Search by name, tags and description

HomeLoan-Train

Submit Publish

Autosave on

100%

Search in canvas

Draft autosaved on 8/9/2020, 10:25:29 AM

Modules

- Data Input and Output (3)
- Data Transformation (19)
- Feature Selection (2)
- Statistical Functions (1)
- Machine Learning Algorithms (18)
- Model Training (4)
- Model Scoring & Evaluation (6)
- Python Language (2)
- R Language (1)
- Text Analytics (7)
- Computer Vision (6)
- Recommendation (5)
- Anomaly Detection (2)
- Web Service (2)

Canvas

Settings

- Default compute target**
Finance-CompClus
[Select compute target](#)
- Pipeline parameters**
No parameters selected
- Default output settings**
[Select default datastore](#)
- Draft details**
Draft name: HomeLoan-Train
Draft description (optional): Pipeline created on 20200809
Created on: August 9, 2020 10:21 AM
Created by: Hafizah Ab Rahim
Last edit time: August 9, 2020 10:22 AM
Last edited by: Hafizah Ab Rahim

Steps

25. Click "Datasets" and choose your dataset. Drag your dataset onto the canvas.

The screenshot displays the Databricks workspace interface. On the left sidebar, the 'Datasets' option is highlighted under the 'Assets' section. The main panel is titled 'Datasets (8)' and lists several datasets. The dataset 'Loan' (Version 2 with correct data type) is highlighted with a yellow border. An orange arrow points from this dataset to the canvas area, where it is being dragged. The canvas area shows a 'Loan' dataset box. The right sidebar displays the details for the 'Loan' dataset, including its ID, name, source type, and path.

Finance-WS > Designer (preview) > Authoring

Search by name and description

HomeLoan-Train ⚙️

Submit Publish ...

Autosave on

100%

Draft autosaved on 8/9/2020, 10:28:08 AM

Datasets

Datasets (8)

- TD-Loan-training-Convert_to_Indicator_V...
This is a dataset promoted by inference graph generation automatically on 08/09/2020 01:10...
8/8/2020
- TD-Loan-training-Clean_Missing_Data-Cle...
This is a dataset promoted by inference graph generation automatically on 08/09/2020 01:11...
8/8/2020
- TD-Loan-training-Normalize_Data-Transf...
This is a dataset promoted by inference graph generation automatically on 08/09/2020 01:10...
8/8/2020
- TD-Loan-training-Convert_to_Indicator_V...
This is a dataset promoted by inference graph generation automatically on 08/09/2020 01:10...
8/8/2020
- Loan-Test
8/8/2020
- Loan**
Version 2 with correct data type
8/8/2020

Loan

Parameters Outputs

ID
10d2d5b5-5760-4a21-8836-f21d8f723f39

Dataset name
Loan

Datasource type
AmlDataset

Description
Version 2 with correct data type

Data type
DataFrameDirectory

Datastore name
workspaceblobstore

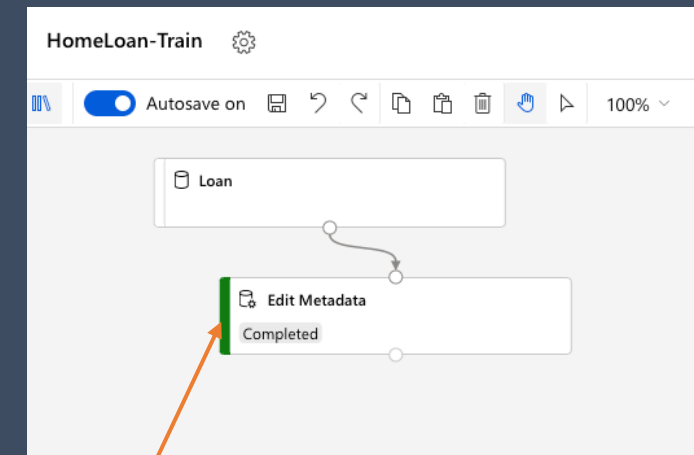
Relative path
UI/08-08-2020_095558.UTC/Loan_Train.csv

Steps

26. Click "Modules" and then, click "Data Transformation". Find "Edit Metadata" and drag it to the canvas .

The screenshot shows the 'HomeLoan-Train' interface. On the left, the 'Modules' panel is open, displaying a list of modules including 'Convert to Dataset', 'Convert to Indicator Values', 'Edit Metadata', 'Group Data into Bins', 'Join Data', and 'Normalize Data'. The 'Edit Metadata' module is highlighted with a blue border. An orange arrow points from the 'Edit Metadata' module in the list to the 'Edit Metadata' module on the canvas. Another orange arrow points from the 'Edit Metadata' module on the canvas to the 'Edit Metadata' settings panel on the right. The settings panel includes fields for 'Column names', 'Data type', 'Categorical', 'Fields', and 'New column names'. The 'Fields' field is set to 'Features'.

Don't forget to connect the boxes using arrows!



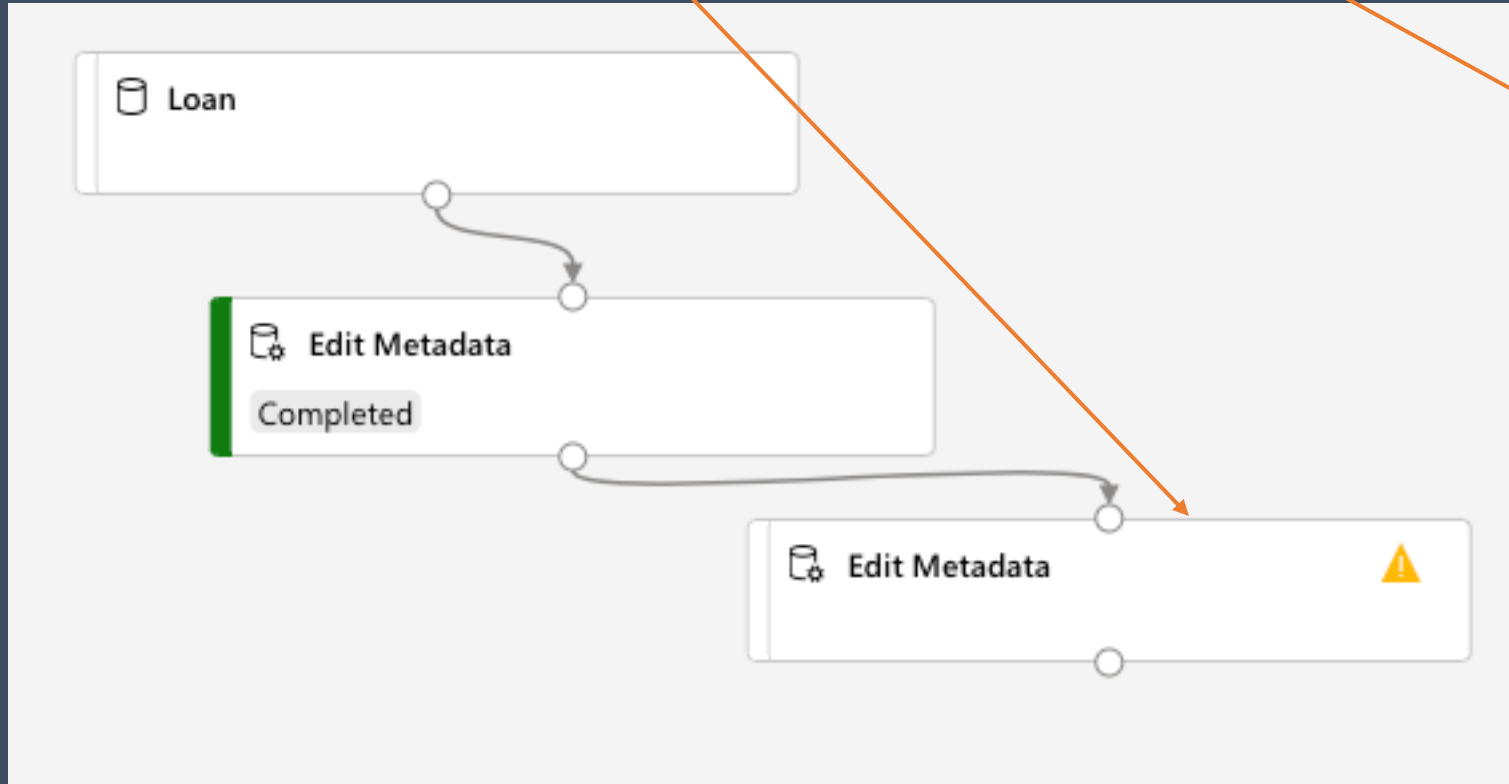
A green highlighted box shows that you have successfully run the pipeline

27. Choose columns that are categorical and label "Fields" as "Features". If you miss this step, you won't be able to "Score" your model.

The screenshot shows the 'Column' settings panel. The 'Select columns' section has 'With rules' selected. The 'Include' section shows a list of columns: 'Education', 'Self_Employed', and 'Property_Area'. A tooltip message is displayed: 'Please run the modules before this one if you want to have auto-complete when typing columns.' The 'Save' button is highlighted with a blue border.

Steps

28. Add another "Edit Metadata". Add a column that acts as a label.



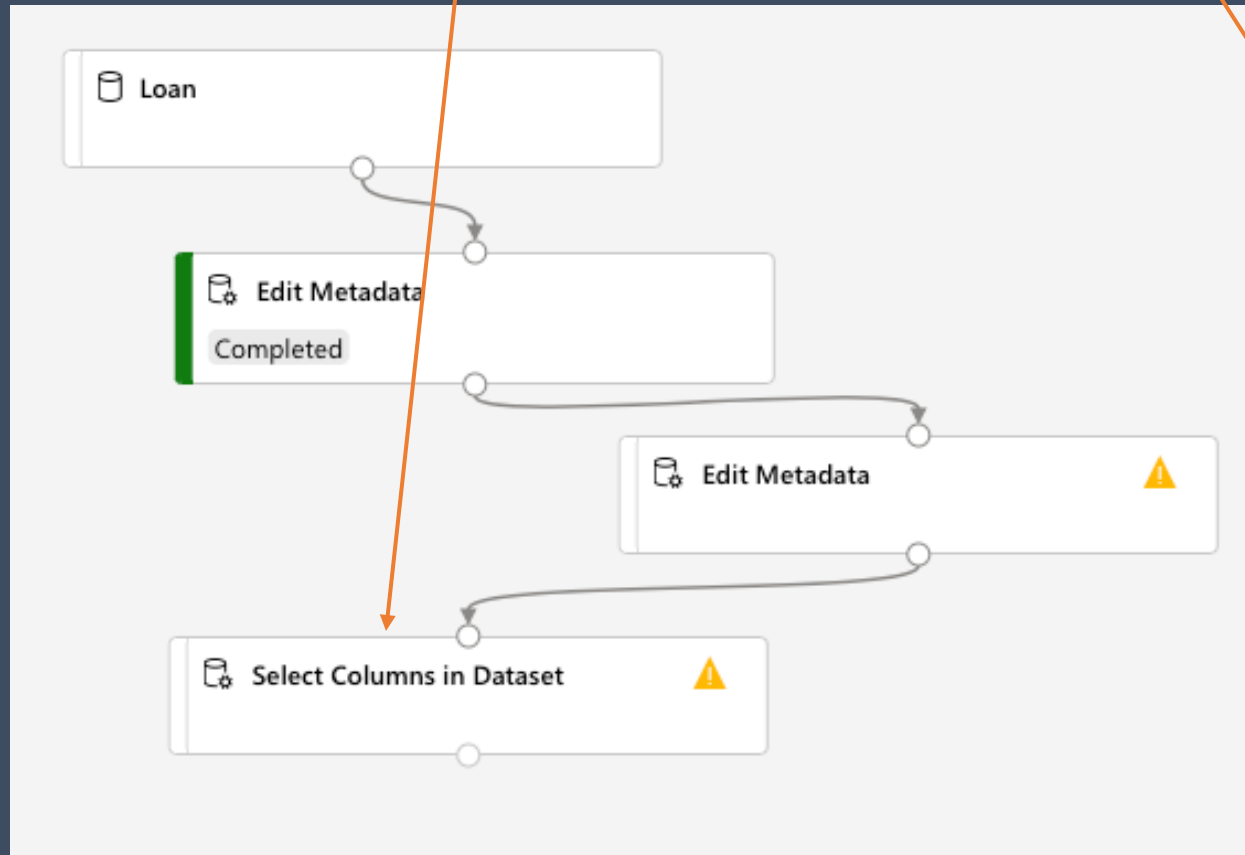
Edit Metadata

Parameters	Outputs + logs	Details	Metrics	Child
Column ② *				
Column names: Loan_Status				
Data type ② *				
Unchanged				
Categorical ② *				
Categorical				
Fields ② *				
Labels				
New column names ②				
Output settings				
Run settings				
Comment				
Module information				

This is to ensure that we can easily eliminate this part when running the pipeline using the test dataset.
(**Testing process does not require label column**)

Steps

29. Add "Select Columns in Dataset" from the "Data Transformation" module and drag it to the canvas and choose columns that you want to include and exclude for analysis.



Select Columns in Dataset

Parameters Outputs + logs Details Metrics Child runs

Select columns ⓘ * [Edit column](#)

All columns
Exclude column names: Loan_ID,CoapplicantIncome,Gender,Married

Output settings >

Run settings >

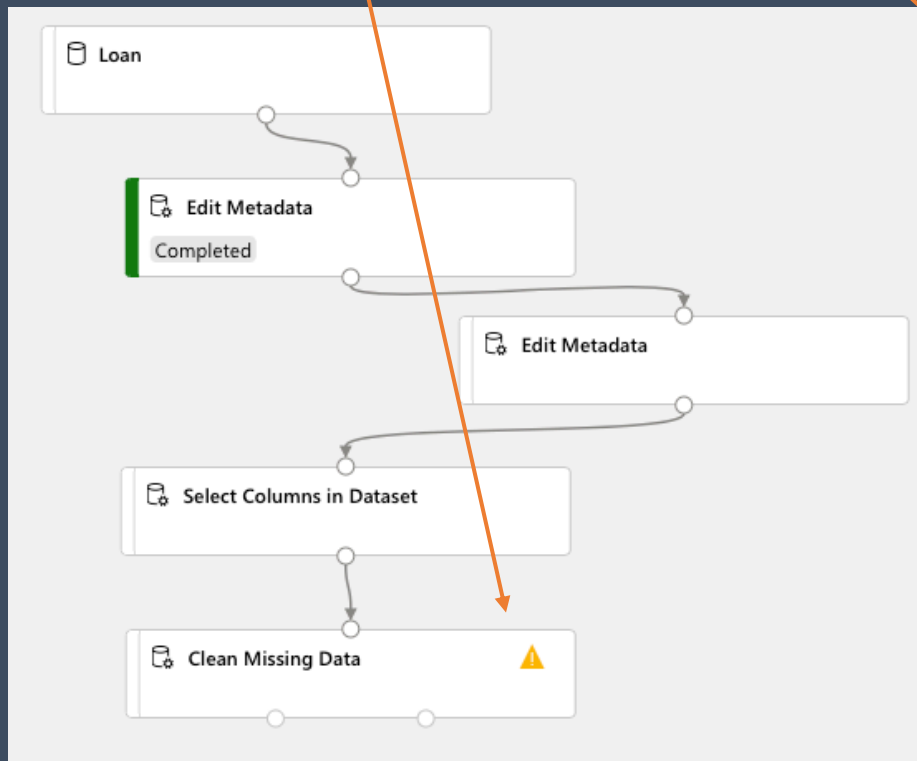
Comment >

Module information >

Steps

30. Add "Clean Missing Data" from the "Data Transformation" module and drag it to the canvas .

Boxes on the right show statistics before and after cleaning the missing data



Clean Missing Data

Parameters Outputs + logs Details Metrics Child runs

Columns to be cleaned ? *

Column names:
Dependents, Education, Self_Employed, ApplicantIncome, LoanAmount, Loan_Amount_Term, Credit_History, Property_Area

Minimum missing value ratio ? *

0.0

Maximum missing value ratio ? *

1.0

Cleaning mode ? *

Remove entire row

Self_Employed	
Statistics	
Mean	-
Median	-
Min	-
Max	-
Standard deviation	-
Unique values	2
Missing values	32
Feature type	Categorical Feature

Self_Employed	
Statistics	
Mean	-
Median	-
Min	-
Max	-
Standard deviation	-
Unique values	2
Missing values	0
Feature type	Categorical Feature

Steps

31. Add "Normalize Data" from the "Data Transformation" module and drag it to the canvas .

Boxes below show statistics before and after data normalization.



Normalize Data

Parameters Outputs + logs Details Metrics Child runs

Transformation method ⓘ *

MinMax

☒ Use 0 for constant columns when checked

Columns to transform ⓘ *

Column names: ApplicantIncome, LoanAmount, Loan_Amount_Term

Edit column

ApplicantIncome	LoanAmount
4583	128
3000	66
2583	120
6000	141
5417	267
2333	95

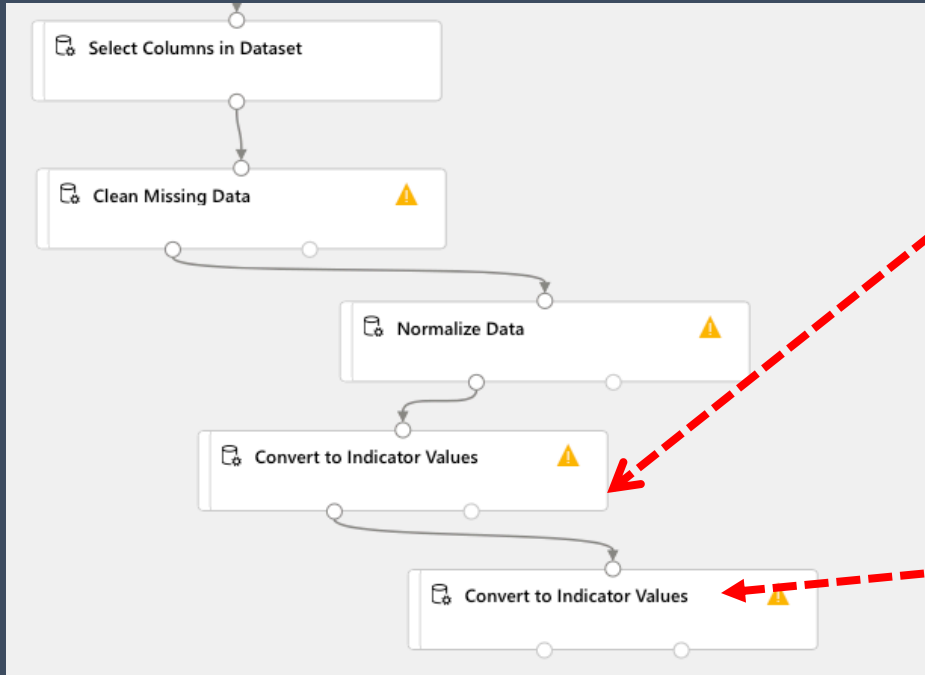
ApplicantIncome	LoanAmount
0.05483	0.172214
0.03525	0.082489
0.030093	0.160637
0.072356	0.191027
0.065145	0.373372
0.027001	0.124457

Steps

32. Add "Convert to Indicator Values" from the "Data Transformation" module and drag it to the canvas .

Click the box to overwrite the original column

Boxes below show statistics before and after changing to Boolean values.



Convert to Indicator Values

Parameters Outputs + logs Details Metrics Child runs

Categorical columns to convert * [Edit column](#)

Column names: Education,Self_Employed,Property_Area

☒ Overwrite categorical columns

Convert to Indicator Values

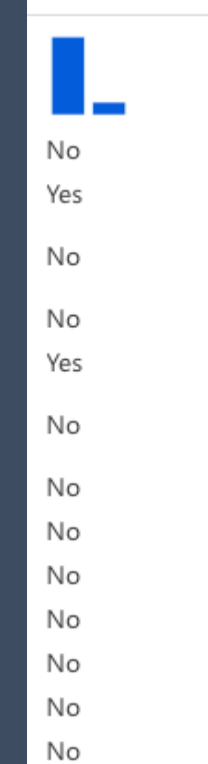
Parameters Outputs + logs Details Metrics Child runs

Categorical columns to convert * [Edit column](#)

Column names: Loan_Status

☐ Overwrite categorical columns

Self_Employed

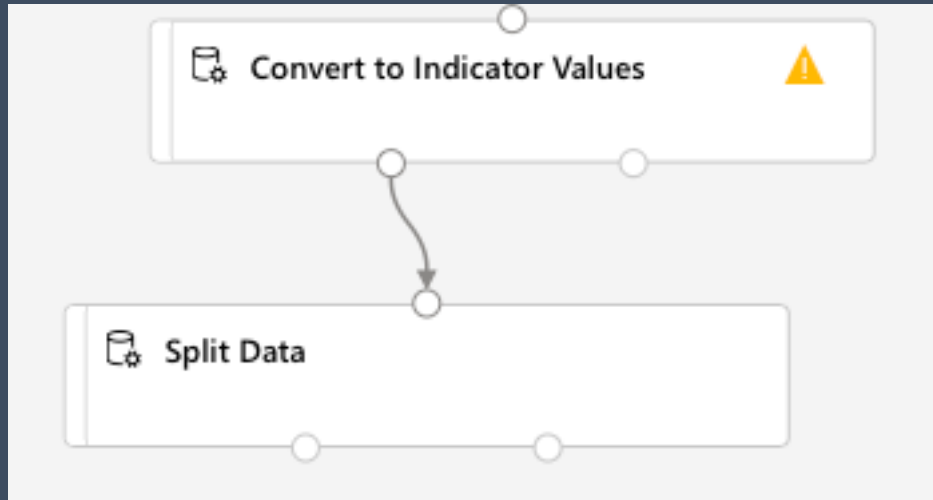


	Self_Employed- No	Self_Employed- Yes
No	1	0
Yes	0	1

This is to ensure that we can easily eliminate this part when running the pipeline using the test dataset.
(**Testing process do not require label column**)

Steps

33. Add "Split Data" from the "Data Transformation" module and drag it to the canvas .



Split Data

Parameters Outputs + logs Details Metrics Child runs

Splitting mode ? *

Split Rows

Fraction of rows in the first output dataset ? *

0.7

☒ Randomized split

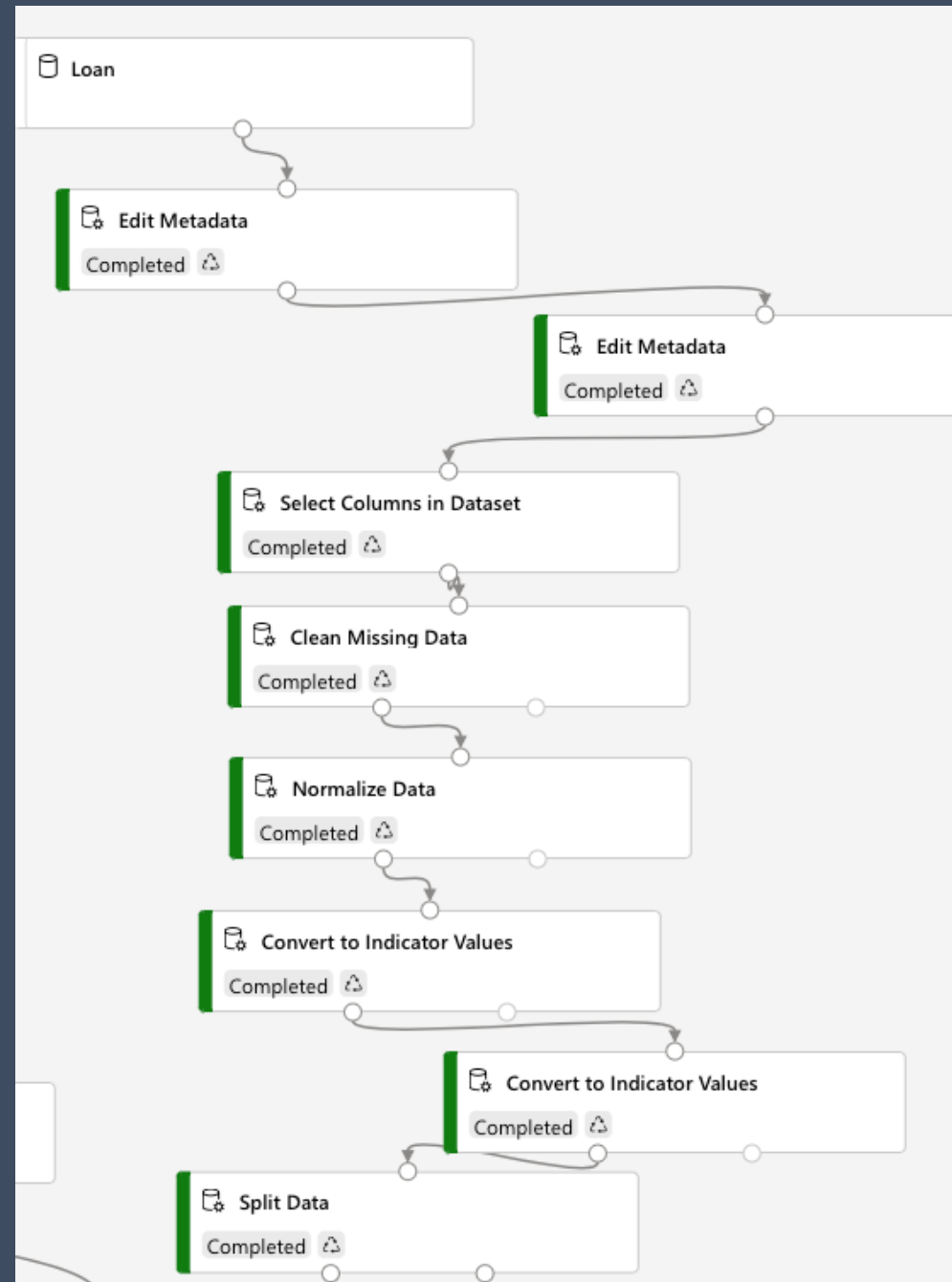
Random seed ? *

123

Stratified split ? *

False

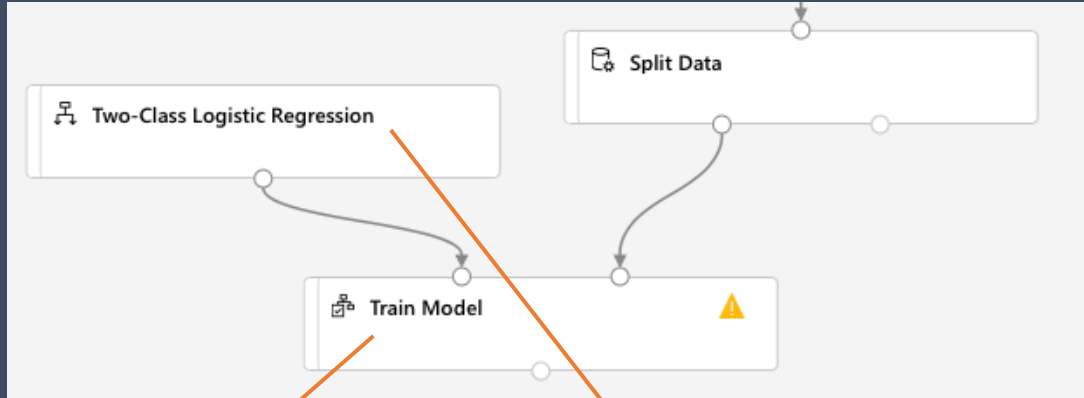
Run the pipeline. Make sure all boxes are marked "completed" before proceeding to the step.



Steps

34. Add "Train Model" from the "Model Training" module and drag it to the canvas.

35. Add "Two-Class Logistic Regression" from the "Machine Learning Algorithm" module and drag it to the canvas.



Train Model

Parameters Outputs + logs Details Metrics Child runs

Label column ? *

Column names: Loan_Status-Y

Output settings >

Run settings >

Comment >

Module information >

Two-Class Logistic Regression

Parameters Outputs + logs Details Metrics Child runs

Create trainer mode ? *

SingleParameter

Optimization tolerance ? *

1e-07

L2 regularization weight ? *

1.0

Random number seed ?

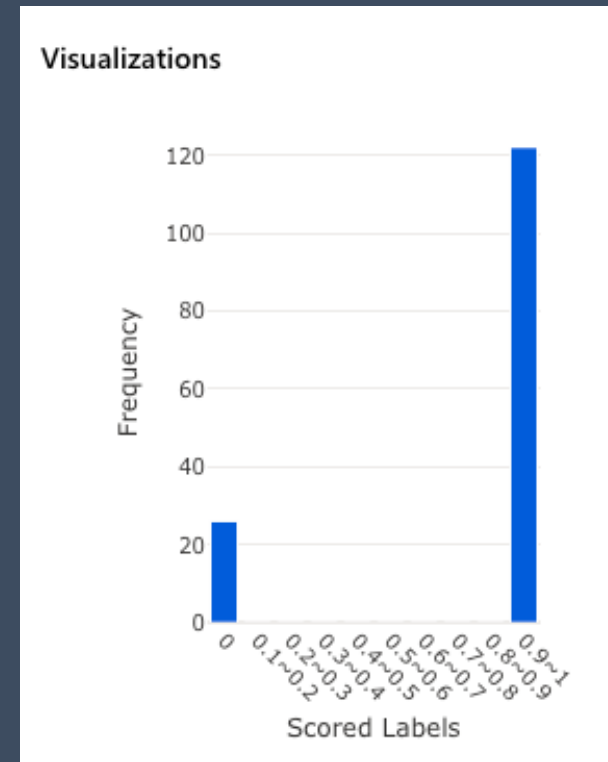
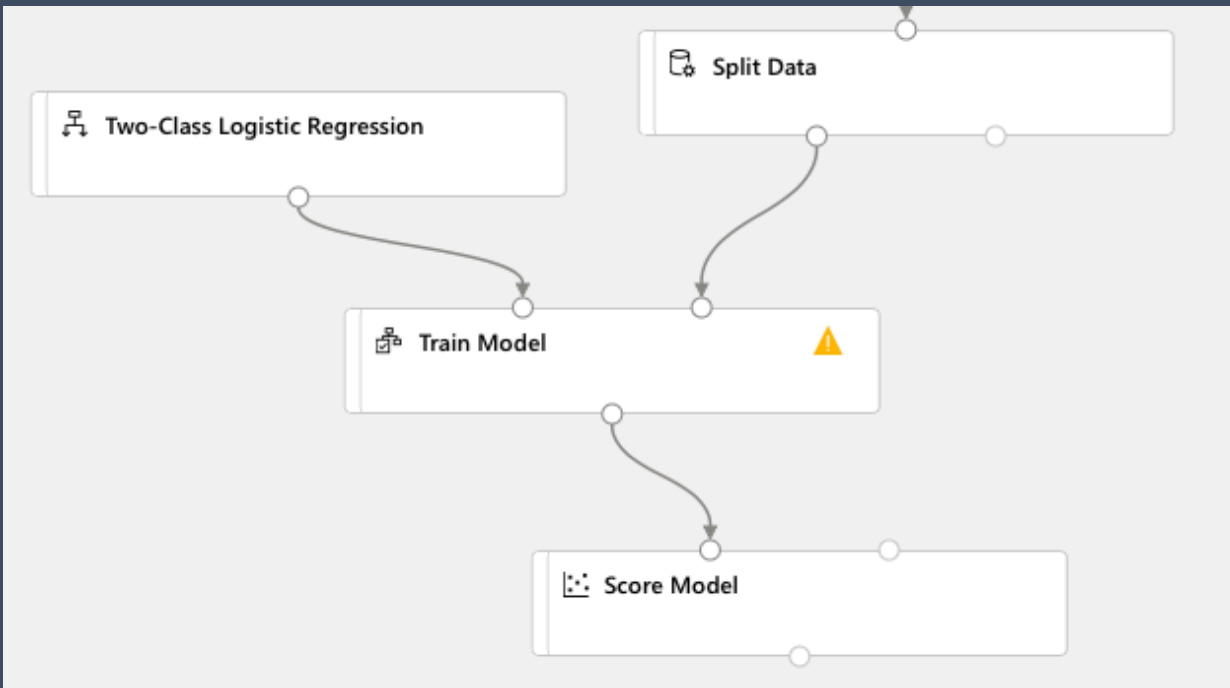
The "Loan_Status-Y" column best represents the "Loan_Status" column where N=0 and Y=1.

Loan_Status-N	Loan_Status-Y	Loan_Status
1	0	N
0	1	Y
0	1	Y
0	1	Y
0	1	Y
0	1	Y
1	0	N
0	1	Y
1	0	N
0	1	Y
0	1	Y
1	0	N
0	1	Y

Steps

36. Add "Score Model" from the "Model Scoring and Evaluation" module and drag it to the canvas.

These are the results after scoring process is complete.

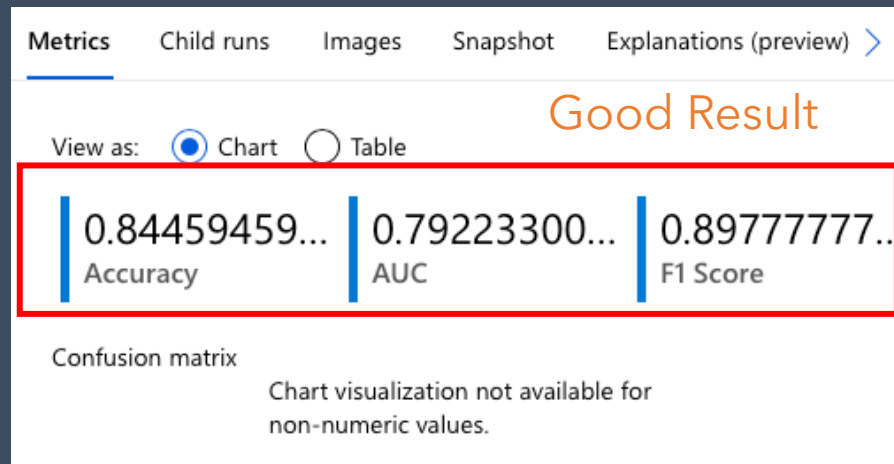
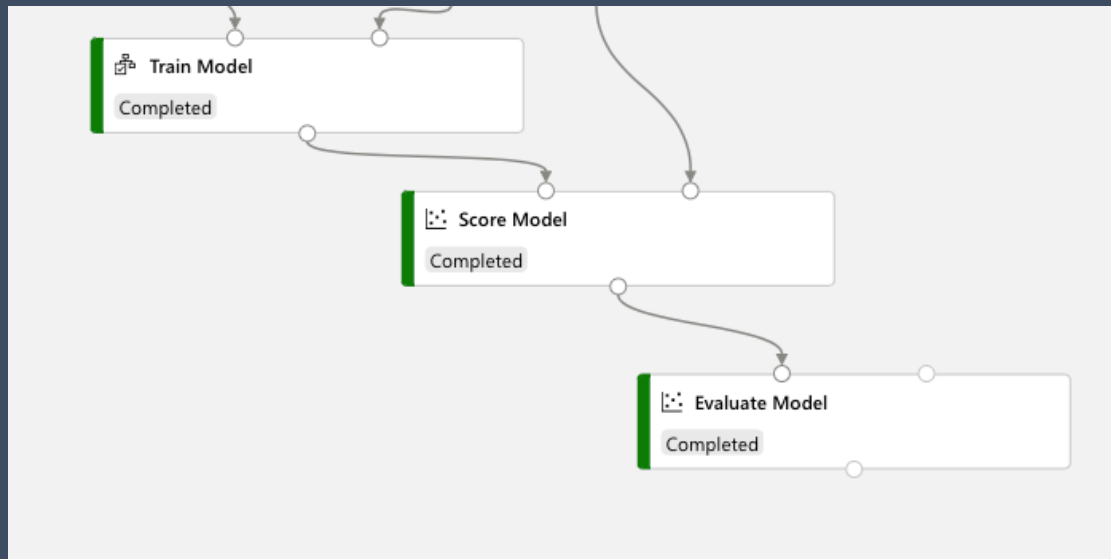


Scored Labels	Scored Probabilities
1	0.644631
1	0.903914
1	0.691953
1	0.64878
1	0.714284
1	0.744122
1	0.755227
1	0.801475
1	0.90392

Steps

37. Add "Evaluate Model" from the "Model Scoring and Evaluation" module and drag it to the canvas.

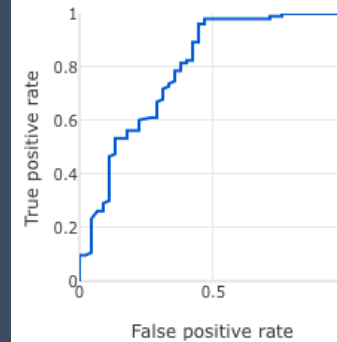
These are the results after the evaluation process is complete.



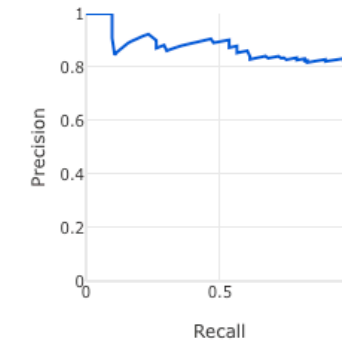
Evaluate Model result visualization

● Scored dataset (left port)

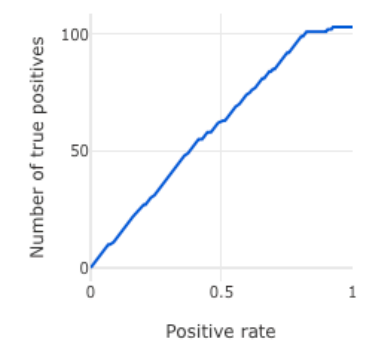
ROC curve



Precision-recall curve



Lift curve



Threshold 0.5

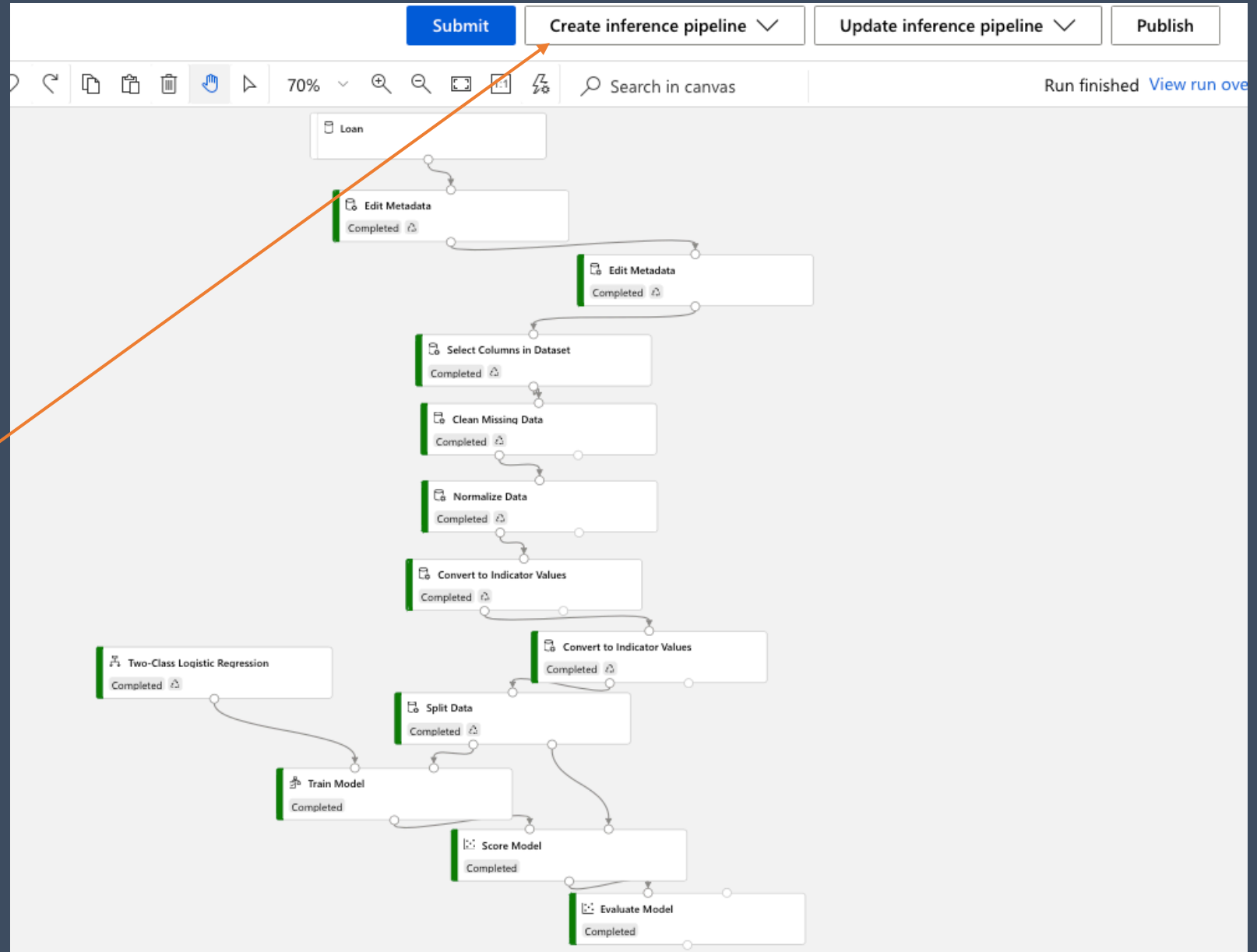
Accuracy 0.845
Precision 0.828
Recall 0.981
F1 Score 0.898
AUC 0.792

		Actual	
		1	0
Predicted	1	101	21
	0	2	24

Steps

Submit the pipeline. Make sure all boxes are marked "completed" before proceeding to the next step.

38. Click "Create inference pipeline"

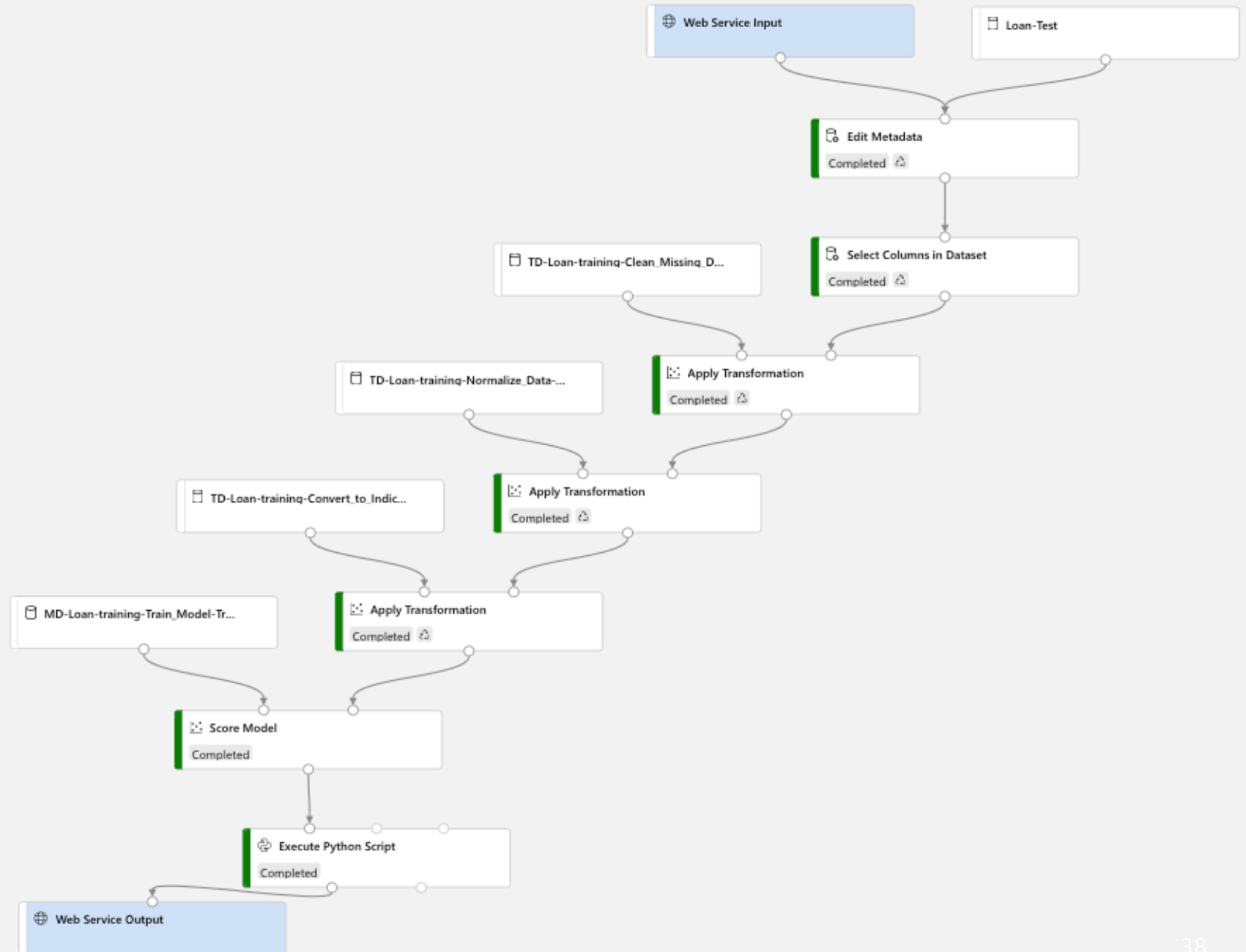


Create an Inference Pipeline

After creating and running a pipeline to train the model, you need a second pipeline that performs the same data transformations for new data, and then uses the trained model to *inference* (in other words, predict) label values based on its features. This pipeline will form the basis for a predictive service that you can publish for applications to use.

This is a final version of the inference pipeline that can then be deployed.

Steps taken will be explained in the next slide.



Steps

39.

Replace the "Loan" dataset with the testing dataset (Loan-Test).

Delete the second "Edit Metadata" box. The label column is not required here.

Delete the second "Convert to Indicator Values" box. The label column is not required here.

Delete the "Evaluate Model" box.

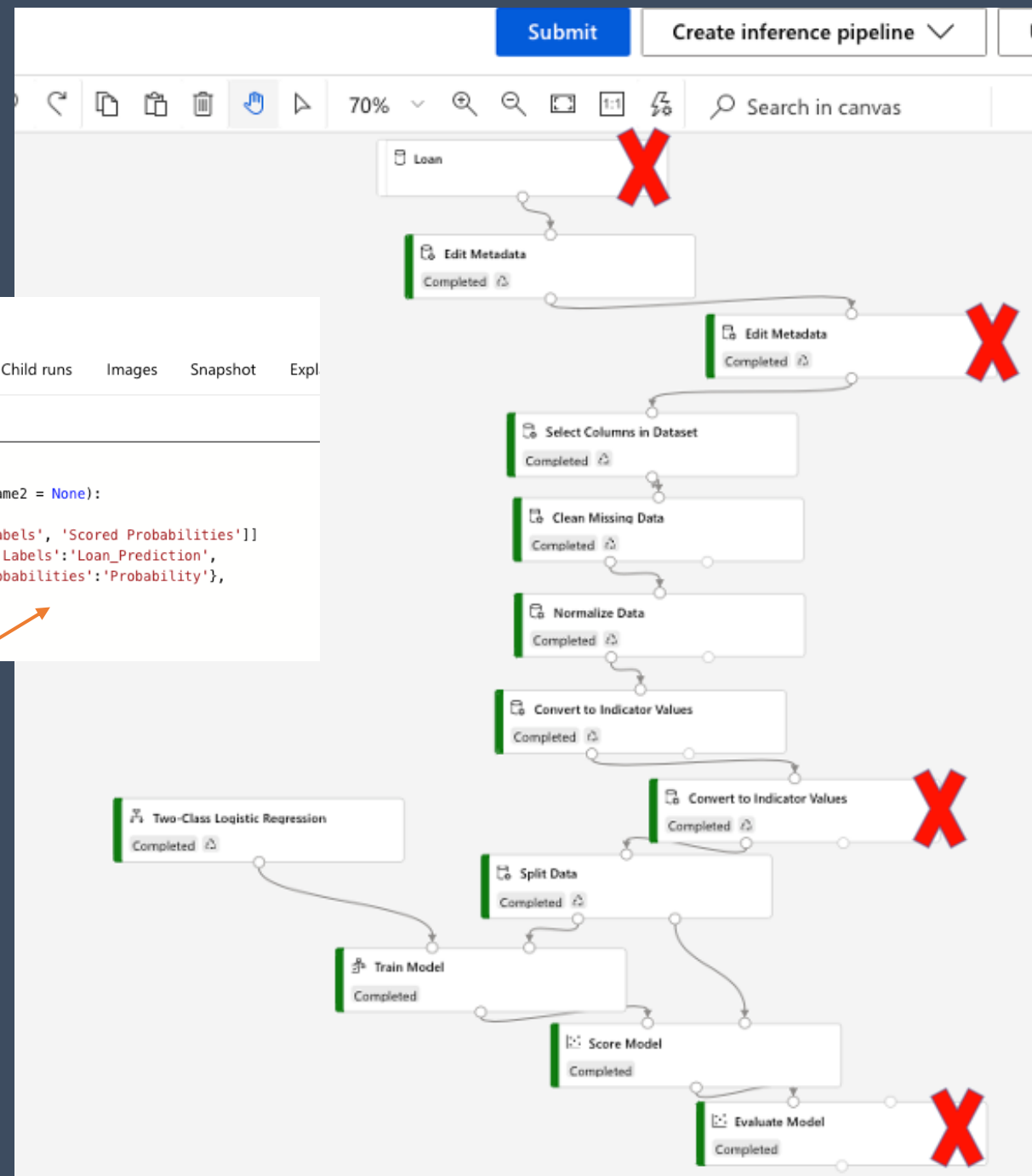
Insert the "Execute Python Script" from the "Python Language" module and drag it to the canvas.

```
Execute Python Script

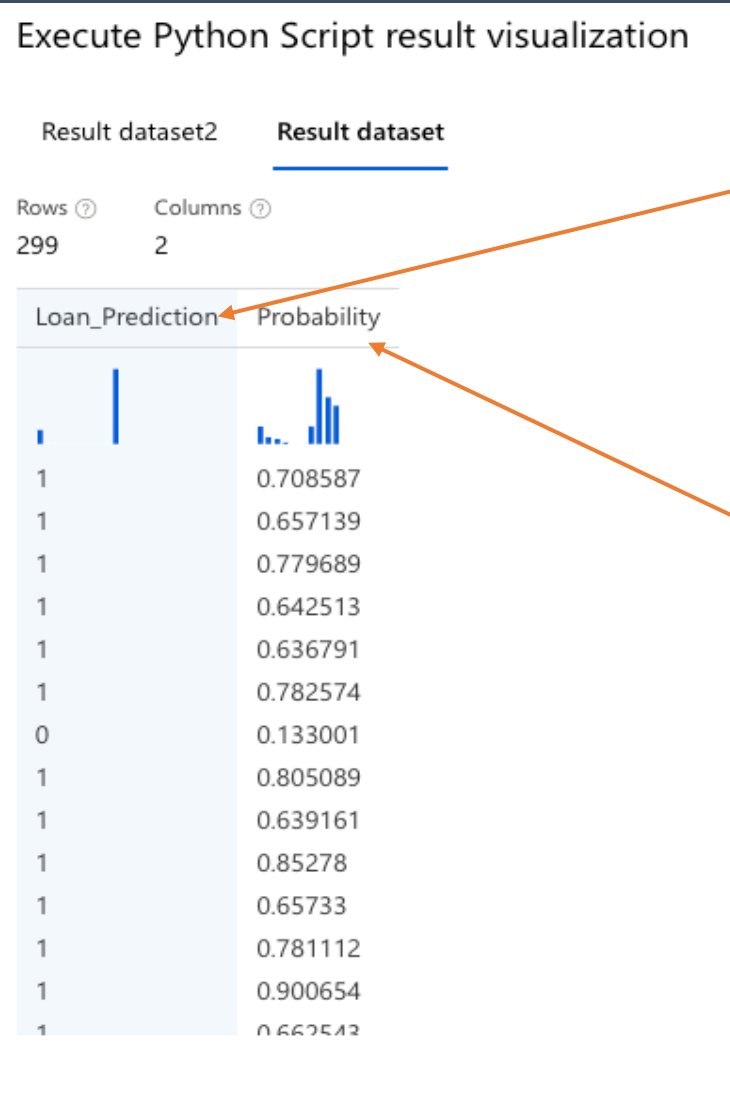
Parameters  Outputs + logs  Details  Metrics  Child runs  Images  Snapshot  Expl

Python script ⓘ *

1 import pandas as pd
2
3 def azureml_main(dataframe1 = None, dataframe2 = None):
4
5     scored_results = dataframe1[['Scored Labels', 'Scored Probabilities']]
6     scored_results.rename(columns={'Scored Labels': 'Loan_Prediction',
7                                   'Scored Probabilities': 'Probability'},
8                           inplace=True)
9     return scored_results
```



Below are the results after the Python script has been executed.



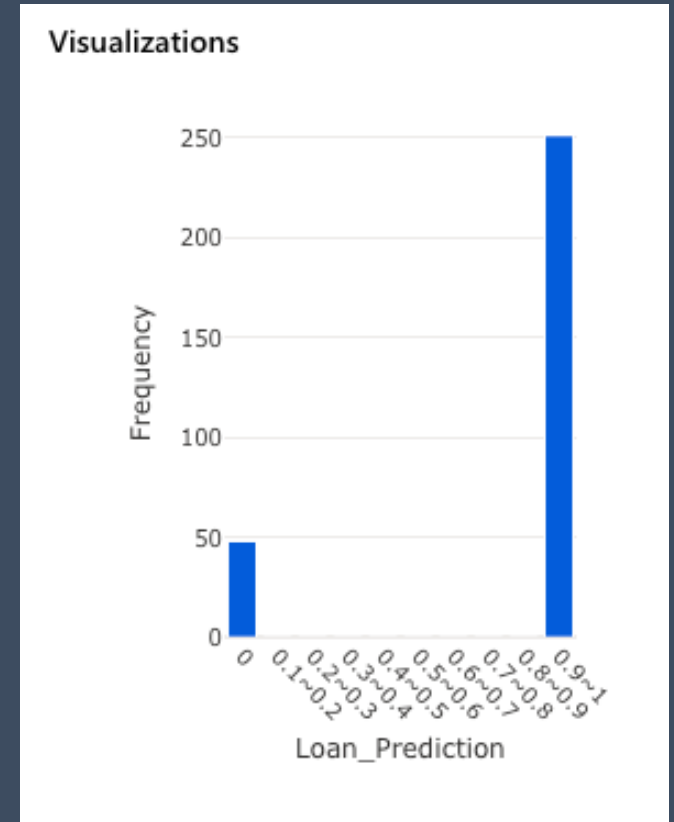
The class label shows binary outcomes of 0 and 1 where 0 signifies a successfully qualified home loan application and 0 signifies a denied application.

The outcomes of 0 or 1 depends on the values in the probability column.

A probability of less than 0.5 gives an outcome of 0.

A probability of more than 0.5 gives an outcome of 1.

The bar chart below shows that the number of qualified home loan applications is 5 times more than denied applications.



Ways to Improve Model Accuracy

1. Use **SMOTE** from the "Data Transformation" module to increase the number of underrepresented cases in the dataset. This would balance out the dataset.
2. Try different machine learning **algorithms** such as Support Vector Machine and Random Forest.
3. Use **feature importance** to do feature selection by choosing only the top 20% of features.

SMOTE

This visualization shows an imbalanced dataset where class "N" has less data than class "Y".

*This visual was coded in Python using Jupyter Notebook.



SMOTE stands for Synthetic Minority Oversampling Technique.

SMOTE takes the entire dataset as an input, but it increases the percentage of only the minority cases.

SMOTE does not change the number of majority cases.

Knowledge check

✓ 200 XP

2 minutes

1. You are using Azure Machine Learning designer to create a training pipeline for a binary classification model. You have added a dataset containing features and labels, a Two-Class Decision Forest module, and a Train Model module. You plan to use Score Model and Evaluate Model modules to test the trained model with a subset of the dataset that was not used for training. Which additional kind of module should you add?

☐ Join Data

☒ Split Data ✓

That is correct. Use a Split Data module to randomly split a dataset into test and validation subsets.

☐ Select Columns in Dataset

2. You use an Azure Machine Learning designer pipeline to train and test a binary classification model. You review the model's performance metrics in an Evaluate Model module, and note that it has an AUC score of 0.3. What can you conclude about the model?

☐ The model can explain 30% of the variance between true and predicted labels.

☐ The model predicts accurately for 70% of test cases.

☒ The model performs worse than random guessing. ✓

That is correct. An AUC of 0.5 is what you'd expect with random prediction of a binary model.

3. You use Azure Machine Learning designer to create a training pipeline for a classification model. What must you do before deploying the model as a service?

☒ Create an inference pipeline from the training pipeline ✓

That is correct. You must create an inference pipeline to deploy as a service.

☐ Add an Evaluate Model module to the training pipeline

☐ Clone the training pipeline with a different name