Tutorial for Machine Learning Pipeline

**OEA Curated**

Date Published: December, 2021

This document’s content is governed by [Microsoft’s Legal Notices](https://github.com/microsoft/OpenEduAnalytics/blob/main/README.md#legal-notices) for open source contributions.

Table of Content

[I. Run a sample pipeline 4](#_Toc90843887)

[1. Preliminaries 4](#_Toc90843888)

[2. Setup Azure environment 4](#_Toc90843889)

[2.1 Create Azure Machine Learning (AML) workspace 4](#_Toc90843890)

[2.2 Configure access control in Azure Machine Learning 5](#_Toc90843891)

[2.3 Create a linked service in Azure Synapse Analytics 8](#_Toc90843892)

[3. Setup Pipeline Template 10](#_Toc90843893)

[3.1 Upload pipeline template 10](#_Toc90843894)

[3.2 Update configuration 11](#_Toc90843895)

[3.3 Run pipeline 11](#_Toc90843896)

[4. Check Dataset in Synapse 12](#_Toc90843897)

[4.1 Data (stage1p) 12](#_Toc90843898)

[4.2 Data (stage2p & 2np) 12](#_Toc90843899)

[4.3 Data (stage2p/Processed) 12](#_Toc90843900)

[4.4 Data (stage3p) 13](#_Toc90843901)

[5. Check training result in AML 14](#_Toc90843902)

[5.1 Data (AML) 14](#_Toc90843903)

[5.2 Experiment 14](#_Toc90843904)

[5.3 Source Code 16](#_Toc90843905)

[5.4 Model 16](#_Toc90843906)

[6. Run error analysis 16](#_Toc90843907)

[II. Run your own experiment 18](#_Toc90843908)

[1. Explore data 18](#_Toc90843909)

[2. Engineer features 18](#_Toc90843910)

[2.1 Create featurization notebook 18](#_Toc90843911)

[2.2 Include notebook into pipeline 18](#_Toc90843912)

[3. Re-create main dataset 18](#_Toc90843913)

[4. Change dataset for training 18](#_Toc90843914)

[5. Submit an experiment to AML 19](#_Toc90843915)

[6. Run AutoML 19](#_Toc90843916)

[7. Add a new model to AML 20](#_Toc90843917)

[7.1 Setup development environment 20](#_Toc90843918)

[7.2 Submit experiment with default configuration 20](#_Toc90843919)

[7.3 Implement model 21](#_Toc90843920)

[7.4 Train the new model from Azure Synapse 24](#_Toc90843921)

# Run a sample pipeline

### Preliminaries

* Environment with OpenEduAnalytics
  + To setup an environment, follow the steps in [OpenEduAnalytics/README.md at main · microsoft/OpenEduAnalytics (github.com)](https://github.com/microsoft/OpenEduAnalytics/blob/main/README.md)

### Setup Azure environment

#### Create Azure Machine Learning (AML) workspace

* + 1. On Azure Portal, select “Machine learning”

Graphical user interface, text, application, email

Description automatically generated

* + 1. Click “+ Create”

Graphical user interface, text, application, chat or text message

Description automatically generated

* + 1. Enter “Workspace name” and “Region” as you prefer, chose “Key vault” and “Application insights” under your resource group, create a new container registry, then click “Review + create”

Graphical user interface, application

Description automatically generated

* + 1. Open Azure Machine Learning Studio. Click “Compute” tab > “+New” and create a compute with the name you like (This information will be necessary later)

Graphical user interface, text, application, email

Description automatically generated

#### Configure access control in Azure Machine Learning

* + 1. On Azure portal, open AML workspace created in previous step and click “Access control (IAM)” > “+ Add” > “Add role assignment”

Graphical user interface, text, application, email

Description automatically generated

* + 1. Choose “Contributor” and click “Next”

Graphical user interface, text, application, email

Description automatically generated

* + 1. On “Members” tab,
    - Select “Managed identity”
    - Click “+ Select members”

On “Select managed identities” pane

* + - Select “Synapse workspace” as “Managed Identity”
    - Enter the name of your Synapse workspace and search for it

(An icon of your Synapse workspace will be displayed)

* + - Click the icon of your Synapse workspace
    - Click “Select” button

Graphical user interface, text, application, email

Description automatically generated

* + 1. Click “Review + assign” on “Members” tab and then on “Review + assign” tab

Graphical user interface, text, application, email

Description automatically generatedGraphical user interface, text, application, email

Description automatically generated

#### Create a linked service in Azure Synapse Analytics

* + 1. On Synapse Studio, click ”Manage” tab > “Linked Services” > “+ New”

Graphical user interface, text, application, email

Description automatically generated

* + 1. Enter “Machine Learning” as a search key, select the icon, and click “Continue”

Graphical user interface, application

Description automatically generated

* + 1. Do the following and click “Commit”
    - Enter “Name” as you prefer
    - Select “Managed Identity” as “Authentication method”
    - Select your “Azure subscription”
    - Select your “Azure Machine Learning workspace name” as you have created in 2.1(3)
    - Click “Test connection” and make sure you see “Connection successful”
    - Click “Commit” button

Graphical user interface, text, application, email

Description automatically generated

### Setup Pipeline Template

#### Upload pipeline template

* + 1. Click “Integrate” tab > “+” > “Import resources from support files” and upload “Data\_ML\_pipeline.zip”

Graphical user interface, text, application, email

Description automatically generated

* + 1. Click “Publish all” and

Graphical user interface, text, application, website

Description automatically generated

* + 1. Make sure

Graphical user interface

Description automatically generated

#### Update configuration

* + 1. Open “ml\_config” notebook and modify the following values
  + LINKED\_SERVICE\_NAME Specify the name of the linked service you created in 2.3
  + COMPUTE Specify the name of the compute created in 2.1(4)

Text

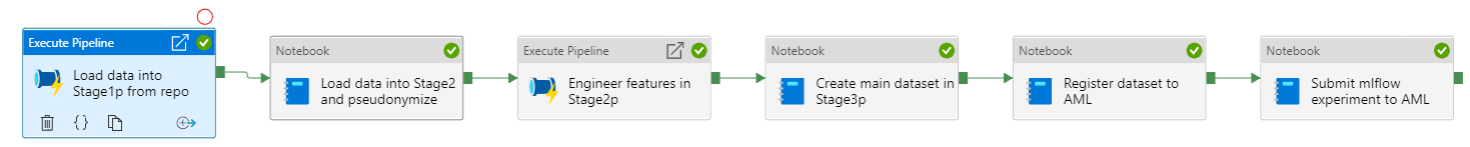
Description automatically generated

#### Run pipeline

* + 1. Click “Commit all” and “Publish”
    2. Click “Debug” and run the pipeline and make sure all activities are completed without errors as shown below

Graphical user interface, text

Description automatically generated



### Check Dataset in Synapse

#### Data (stage1p)

First activity in the pipeline (“Load data into Stage1p - copy test data from URL”) imports data from Github repo to stage1np.

In Synapse, click “Data” tab and check the imported data in stage1np/contoso\_sis/[datetime stamp] and stage1np/m365/[datetime stamp] as shown below. Csv files are downloaded from [Github repo](https://github.com/microsoft/OpenEduAnalytics/tree/main/modules/Contoso_SIS/test_data/batch1) and saved in those folders.

Graphical user interface, text, application, email

Description automatically generated

#### Data (stage2p & 2np)

Second activity in the pipeline (“Load data into Stage2 and pseudonymize”) imports the data in stage1np to stage2p and 2np.

Check “Delta” data imported from stage in stage2p/contoso\_sis/(also stage2np/contoso\_sis/) and stage2p/m365/ as shown below.

Graphical user interface, text, application, email

Description automatically generated

#### Data (stage2p/Processed)

3rd activity in the pipeline (“Engineer features”) creates features for each student and save it as “Delta” data in stage2p/processed/ML\_pipeline/.

Graphical user interface, text, application, email

Description automatically generated

#### Data (stage3p)

4th activity in the pipeline (“Create main dataset in Stage3p”) joins the datasets created in the previous step and saves it as “Delta” in stage3p/ML\_pipeline/MainData. This dataset has all the possible features, key, and labels for machine learning.

Graphical user interface, text, application, email

Description automatically generated

A corresponding datatable is also created in Lake database named “s3p”.

Graphical user interface, text, application

Description automatically generated

### Check training result in AML

#### Data (AML)

* + 1. 5th activity in the pipeline (“Register dataset to AML”) registers 3 datasets to AML. These are the subsets of main dataset created previously.

In AML studio, click “Datasets” tab. Then you see 3 datasets registered as shown as below. These are “features”, “label”, and “sensitive\_feature”. Graphical user interface, text, application, email

Description automatically generated

#### Experiment

* + 1. Click “Experiments” tab > “experiment-ML\_pipeline”>”[random display name as shown below]”. This is the experiment the last activity in Synapse pipeline has submitted.

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface

Description automatically generated

* + 1. Check evaluation results by clicking “Metrics”, ”Explanations”, and ”Fairness”

Graphical user interface, application

Description automatically generated

You can find details in the webpages below

* + Explanations [Use Python to interpret & explain models (preview) - Azure Machine Learning | Microsoft Docs](https://docs.microsoft.com/en-us/azure/machine-learning/how-to-machine-learning-interpretability-aml#visualizations)
  + Fairness [Assess ML models' fairness in Python (preview) - Azure Machine Learning | Microsoft Docs](https://docs.microsoft.com/en-us/azure/machine-learning/how-to-machine-learning-fairness-aml#upload-fairness-insights-for-multiple-models)

#### Source Code

* + 1. Click “Experiments” tab > “experiment-dataconfigA”>”[random display name as shown below]”> “Snapshot” to see the source code downloaded from Github repository. (You can also find the source code by opening the link to the repository in “ml\_config” notebook.)

Graphical user interface, text, application

Description automatically generated

#### Model

* + 1. Click “Models” > “model-ML-pipeline” for the trained model

Graphical user interface, text, application

Description automatically generated

### Run error analysis

* + 1. Upload [error\_analysis.ipynb](https://github.com/ContosoISD3/cisdggimpl5/blob/main_aml/aml_projects/mlflow/template_project/error_analysis.ipynb) to AML and run it

Graphical user interface, application

Description automatically generated

* + 1. Check the analysis result

A picture containing table

Description automatically generatedGraphical user interface, text, application

Description automatically generated

# Run your own experiment

The pipeline imported in the previous section is just template you can use to create your own machine learning pipeline. This section guides several ways to customize the template.

### Explore data

Once new dataset is imported, you will need to manually explore data to find out features that could be useful for your model. The data created in the process is recommended to be saved in XXX.

### Engineer features

#### Create featurization notebook

With the features found in the previous step, you can create a notebook to create intermediate dataset that contains features for each key. For example, if you need to create a machine learning model to identify vulnerable students, the intermediate dataset should have features associated with each student, such as the number of absences with Student ID. You can find sample notebook activities in the pipeline named “Engineer features in Stage2p”.

#### Include notebook into pipeline

Open the third activity named “Engineer features in Stage2p” in the pipeline, look at another pipeline named “FeatureEngineering”, and add a new “notebook” activity that executes the notebook you implemented in the previous step.

### Re-create main dataset

You can include the intermediate dataset you created in the previous step into “Main” dataset by modifying Config section of “create\_main\_dataset” notebook in the pipeline activity named “Create main dataset in Stage3p”. This notebook joins datasets that are created in the previous step and saves it in Stage3p. You can make changes as described in the comments.

A picture containing scatter chart

Description automatically generated

### Change dataset for training

Training datasets needs to be registered to AML before submitting an experiment. This can be done by editing “ml\_config” notebook and running “register\_dataset\_to\_aml” notbook. As described in 5.1, 3 datasets (“feature”, “label”, and “sensitive\_feature”) are necessary to train a ML model and evaluate a trained ML model.

They are the subsets of main dataset created in the previous step. Names of columns needs to be listed as “key”, “feature” (or “exclude feature”), “sensitive\_feature”, and “label”. “sensitive\_feature” is used to evaluate fairness after training a model.

For details of config elements, you can check the comments in “ml\_config” notebook.

Text

Description automatically generated

### Submit an experiment to AML

Once training dataset is created, an experiment can be submitted to AML. In the example above, a linear regression model is used as a machine learning model. But there are several other models you can use without coding as shown in the class diagram included in “7.3 Implement model”. You can specify one of these as “model” in the config notebook above.

### Run AutoML

Instead of running a model in the scripts saved in You can also run AutoML by clicking on “Data” tab > “data\_ml\_pipeline\_maindataset” in s3p” > “Machine Learning” > “Train a new model”.

Graphical user interface, text, application, email

Description automatically generated

You can find detailed documents below.

* [Tutorial: Train a model by using automated machine learning - Azure Synapse Analytics | Microsoft Docs](https://docs.microsoft.com/en-us/azure/synapse-analytics/machine-learning/tutorial-automl)

### Add a new model to AML

As described in “5.Submit an experiment to AML”, there are models already ready to use without coding. If it is necessary to use a model not on the list, it is also possible to create a new model by coding. By implementing a model with certain interface defined in the framework, you can check “Metrics”, ”Explanations”, and ”Fairness” in AML as you see in 5.2(2).

#### Setup development environment

* + 1. You need Python development environment with [Anaconda](https://www.anaconda.com/).
    2. Create virtual environment

Open CUI(\*), go to “aml\_code” folder, run the following command.

|  |
| --- |
| conda env create -f dev-env.yaml |

(\*) e.g. windows: Anaconda Prompt, mac: Terminal

* + 1. Activate virtual environment

Run the following command.

|  |
| --- |
| conda activate dev-oea-modeling |

#### Submit experiment with default configuration

* + 1. Download the code from [OEA Github repository](https://github.com/microsoft/OpenEduAnalytics)
    2. Copy “sample\_config.json” as “config.json” and modify it

You can find your environment variables in Azure portal as shown below. “compute” is the one created in 2.1(4).

* + Azure portal > AML Instance

Graphical user interface, application

Description automatically generated

* + Azure Portal > Azure Active Directory

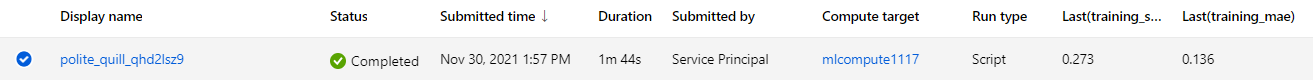
Graphical user interface, text, application, email

Description automatically generated

* + 1. Move to the one with “submit\_pipelines.py” and run

|  |
| --- |
| python submit\_pipeline.py |

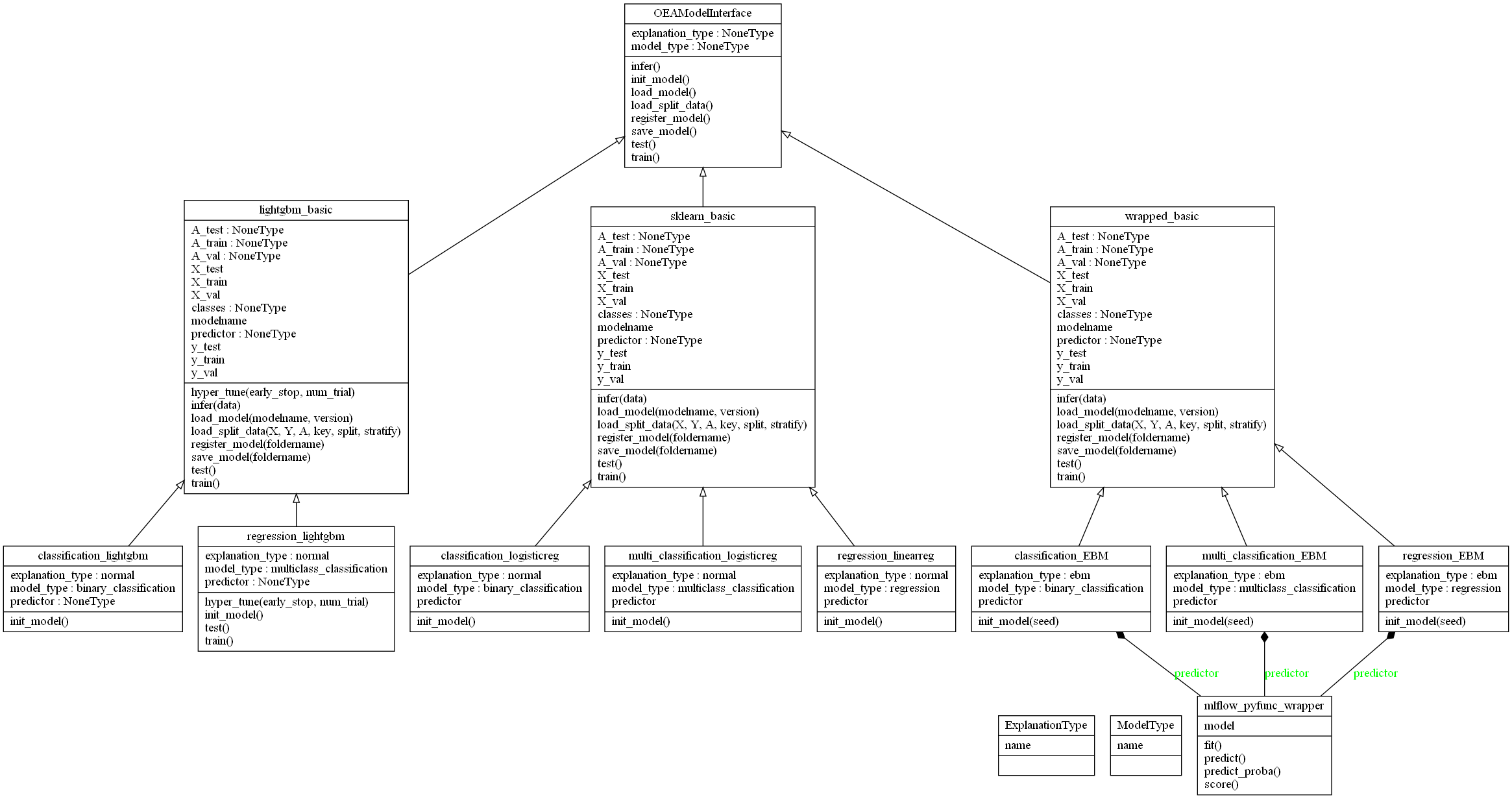
* + 1. Click “Experiments” tab > “template\_test” on AML and make sure that there is one submitted by you.



#### Implement model

* + 1. Create a new python file in “aml\_code/modeling” folder
    2. Define a new model class in the new python file. This class needs to implement methods and specify class variables (“model\_type” and “explanation\_type”) that are defined in OEA\_model.py > “OEAModelInterface” class.

You can check several model classes included in the repository as a reference. The following class diagram shows those classes and their relationships.



* + 1. Modify the following variables in “submit\_pipelines.py”. “model” needs to be changed to the name of the model class you created. If you have registered new datasets to AML in “4. Change dataset for training”, it is also necessary to change “feature\_dataset\_name”, “label\_dataset\_name”, “sensitive\_dataset\_name”, and “key”.

Graphical user interface, text

Description automatically generated

* + 1. Submit an experiment by running the same command used in 7.2(3). Click “Experiments” tab > “template\_test” on AML and make sure that there is one submitted by you.

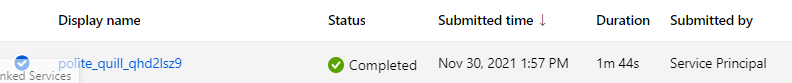
#### Train the new model from Azure Synapse

* + 1. In Synapse, change “model” in “ml\_config” notebook to the name of the model class created in the previous step.

Text

Description automatically generated

* + 1. Run “submit\_mlflow\_experiment\_to\_aml” notebook. Go to AML, click “Experiments” tab > “template\_test”, and make sure that there is one submitted by “Service Principal” and completed.



* + 1. Run the whole pipeline and make sure everything works fine