Machine Learning before the hands-on lab setup guide

Requirements

1. Microsoft Azure subscription must be pay-as-you-go or MSDN
   - Trial subscriptions will not work. You will run into issues with Azure resource quota limits.
   - Subscriptions with access limited to a single resource group will not work. You will need the ability to deploy multiple resource groups.

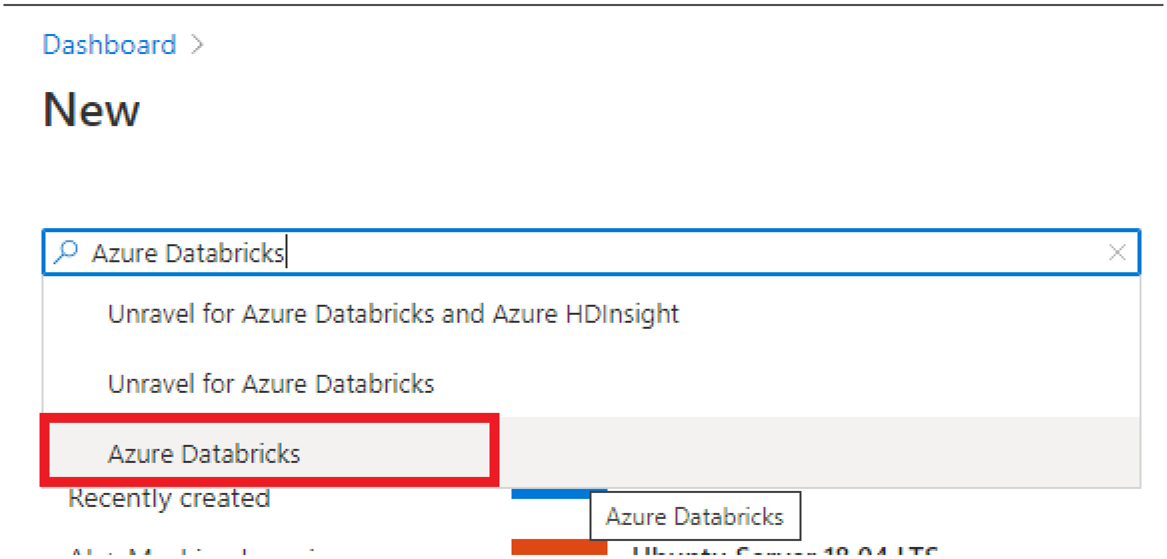
Before the hands-on lab

Duration: 30 minutes

Task 1: Create your Azure Databricks Account

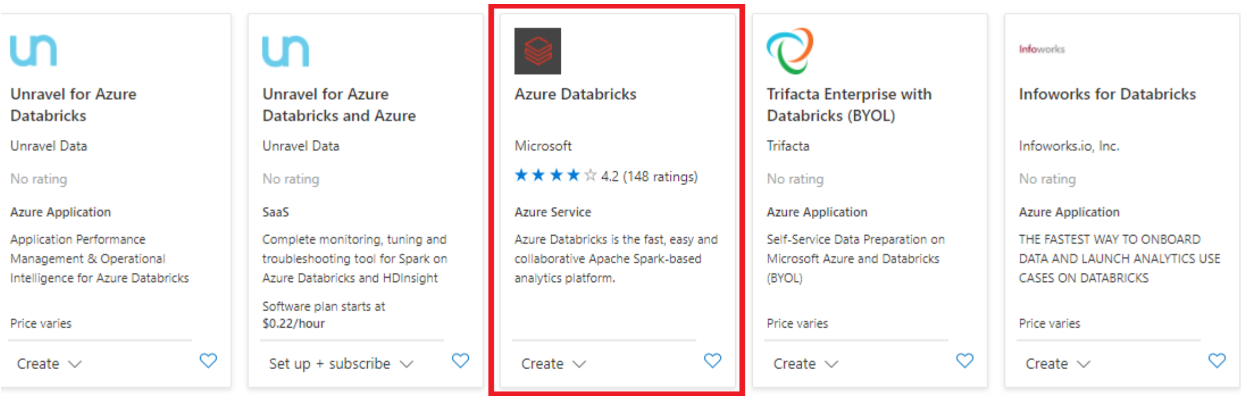
Azure Databricks is an Apache Spark-based analytics platform optimized for Azure that supports data engineering and machine learning and deep learning workloads.

1. In the Azure Portal, select **+ Create a resource**, then type "Azure Databricks" into the seach bar. Select Azure Databricks from the results.



Dashboard >
New
Unravel for Azure Databricks and Azure HDInsight
Unravel for Azure Databricks
Azure Databricks
Recently created
Azure Databricks

Azure Databricks



Unravel for Azure Databricks
Unravel Data
No rating
Azure Application
Application Performance Management & Operational Intelligence for Azure Databricks
Price varies
Create

Unravel for Azure Databricks and Azure
Unravel Data
No rating
SaaS
Complete monitoring, tuning and troubleshooting tool for Spark on Azure Databricks and HDInsight
Software plan starts at $0.22/hour
Set up + subscribe

Azure Databricks
Microsoft
★★★★☆ 4.2 (148 ratings)
Azure Service
Azure Databricks is the fast, easy and collaborative Apache Spark-based analytics platform.
Create

Trifacta Enterprise with Databricks (BYOL)
Trifacta
No rating
Azure Application
Self-Service Data Preparation on Microsoft Azure and Databricks (BYOL)
Price varies
Create

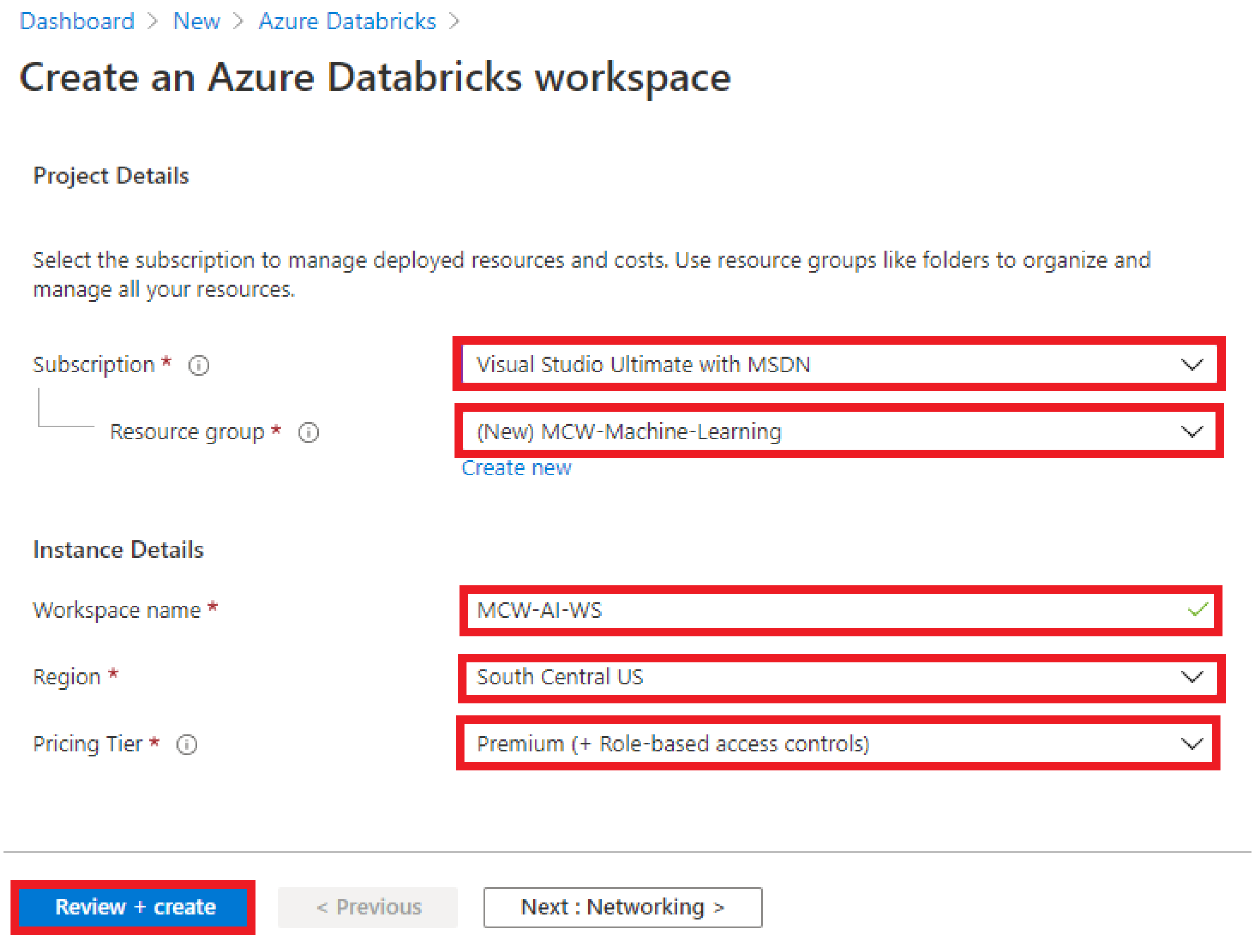
Infoworks for Databricks
Infoworks.io, Inc.
No rating
Azure Application
THE FASTEST WAY TO ONBOARD DATA AND LAUNCH ANALYTICS USE CASES ON DATABRICKS
Price varies
Create

2. Select **Create** on the bottom of the blade that follows.

3. Set the following configuration on the Azure Databricks Service creation forms:

   - **Workspace name** : Enter a unique name as indicated by a green checkmark.
   - **Subscription**: Select the subscription you are using for this hands-on lab.
   - **Resource Group**: Select **Create new** and provide the name `MCW-Machine-Learning`
   - **Region**: Select a region close to you. ***(If you are using Azure Pass, select South Central US.)***
   - **Pricing**: Select Premium

4. Select **Review + Create** and then select **Create** when the form values passes



Dashboard > New > Azure Databricks >

Create an Azure Databricks workspace

Project Details

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

Subscription *	Visual Studio Ultimate with MSDN
Resource group *	(New) MCW-Machine-Learning Create new

Instance Details

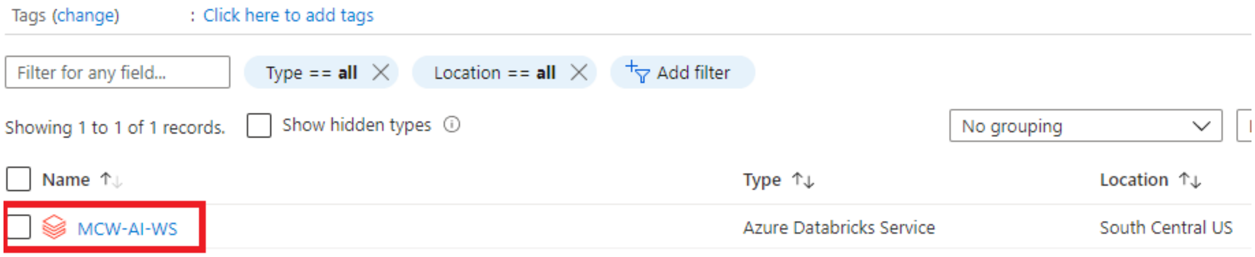
Workspace name *	MCW-AI-WS ✓
Region *	South Central US
Pricing Tier *	Premium (+ Role-based access controls)

Task 2: Create an Azure Databricks cluster

You have provisioned an Azure Databricks workspace, and now you need to create a new cluster within the workspace.

1. From within Azure portal navigate to your resource group name (e.g., `MCW-Machine-Learning`).

2. Next, select your Azure Databricks service from the



Tags (change) : Click here to add tags

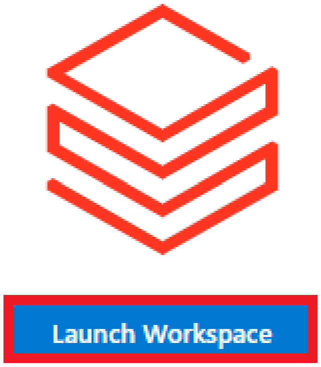
Filter for any field... | Type == **all** ✕ | Location == **all** ✕ | Add filter

Showing 1 to 1 of 1 records. □ Show hidden types ⓘ

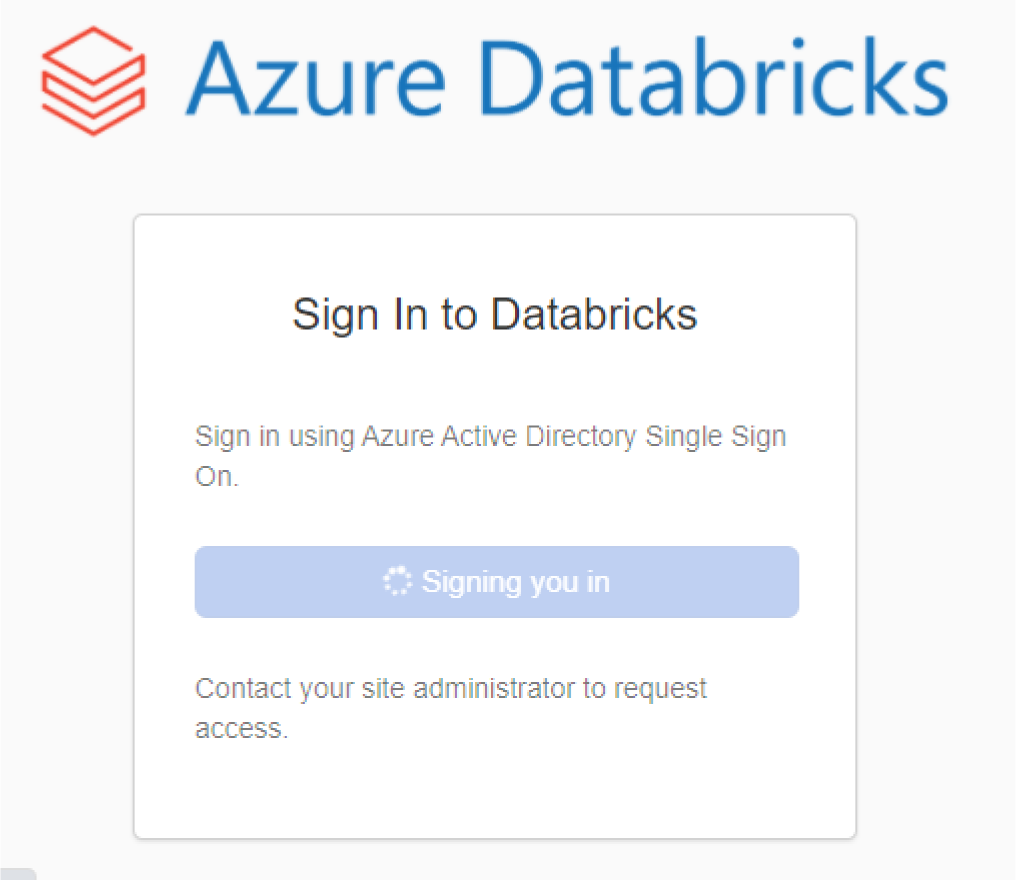
No grouping

□ Name ↑↓	Type ↑↓	Location ↑↓
□ MCW-AI-WS	Azure Databricks Service	South Central US

3. In the Overview pane of the Azure Databricks service, **Launch Workspace**.



Azure Databricks will automatically log you in using Azure Active Directory Single



Azure Databricks

Sign In to Databricks

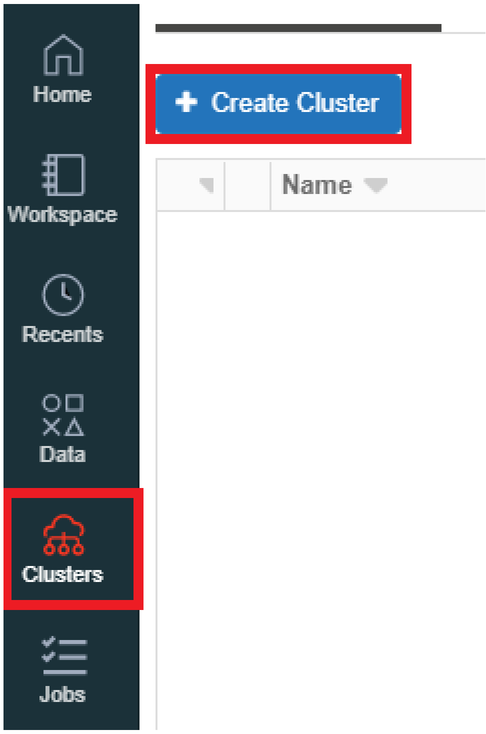
Sign in using Azure Active Directory Single Sign On.

Signing you in

Contact your site administrator to request access.

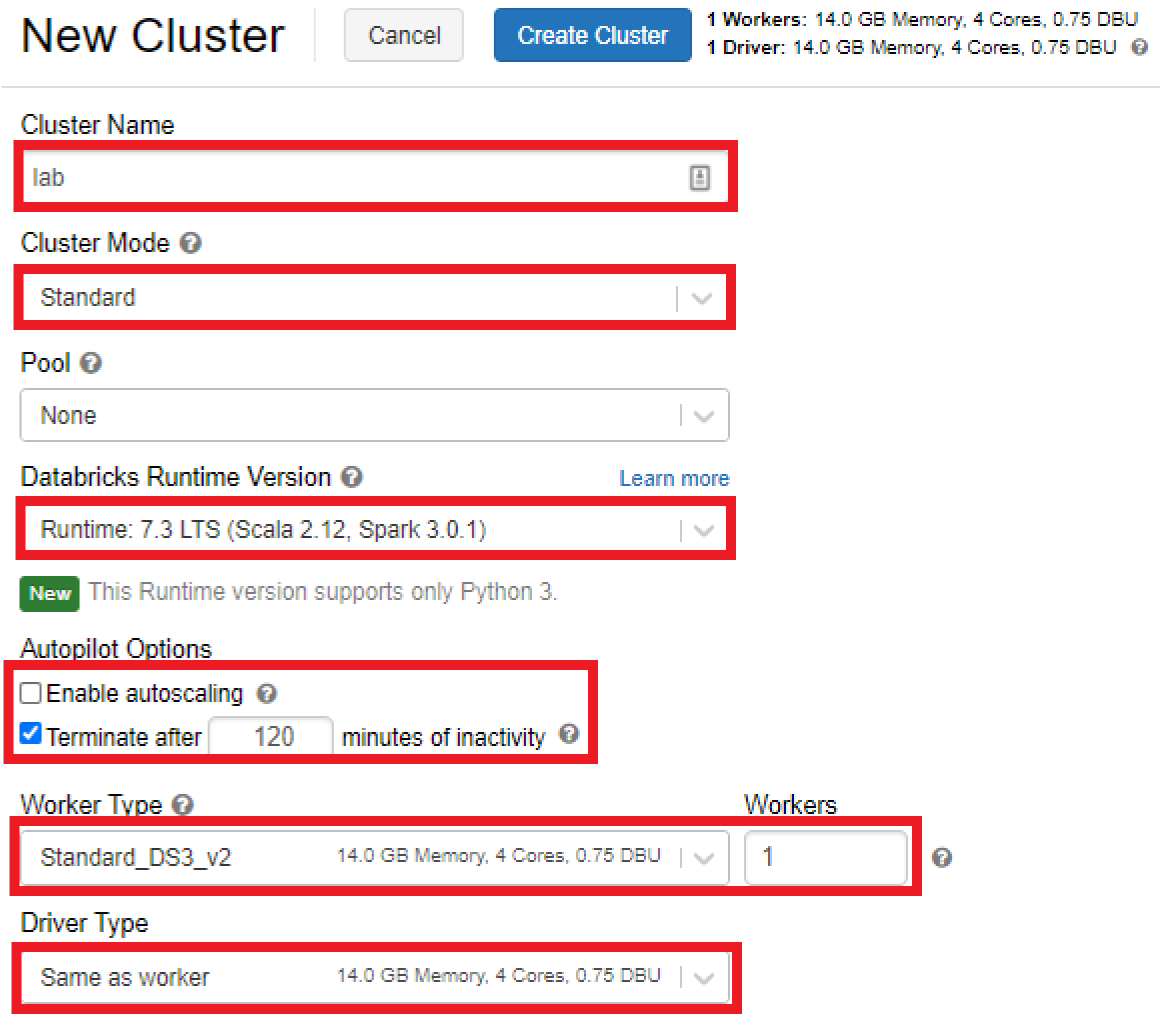
Sign On.

4. Select **Clusters** (1) from the menu, then select **Create Cluster** (2).



5. On the Create New Cluster form, provide the following:

   - **Cluster Name**: `lab`
   - **Cluster Mode**: **Standard**
   - **Databricks Runtime Version**: **Runtime: 7.3 LTS (Scala 2.12, Spark 3.0.1)**
   - **Enable autoscaling**: **Uncheck** this option.
   - **Terminate after ___ minutes of inactivity**: Leave **checked** and in the text box enter `120`.
   - **Worker Type**: **Standard_DS3_v2**
   - **Workers**: `1`
   - **Driver Type**: **Same as worker**



New Cluster

1 Workers: 14.0 GB Memory, 4 Cores, 0.75 DBU
1 Driver: 14.0 GB Memory, 4 Cores, 0.75 DBU

Cluster Name: lab

Cluster Mode: Standard

Pool: None

Databricks Runtime Version: Runtime: 7.3 LTS (Scala 2.12, Spark 3.0.1)

New This Runtime version supports only Python 3.

Autopilot Options

□ Enable autoscaling

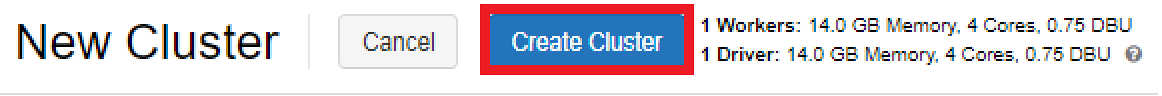
☑ Terminate after 120 minutes of inactivity

Worker Type: Standard_DS3_v2 14.0 GB Memory, 4 Cores, 0.75 DBU

Workers: 1

Driver Type: Same as worker 14.0 GB Memory, 4 Cores, 0.75 DBU

6. Select **Create Cluster**. It will take few minutes to create the cluster. Please ensure that the cluster state is running before proceeding further.



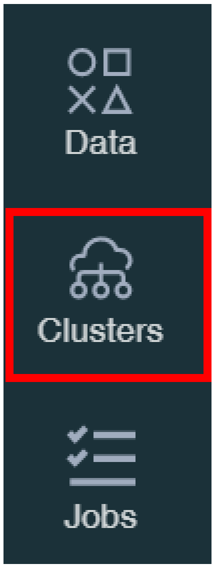
New Cluster

1 Workers: 14.0 GB Memory, 4 Cores, 0.75 DBU
1 Driver: 14.0 GB Memory, 4 Cores, 0.75 DBU

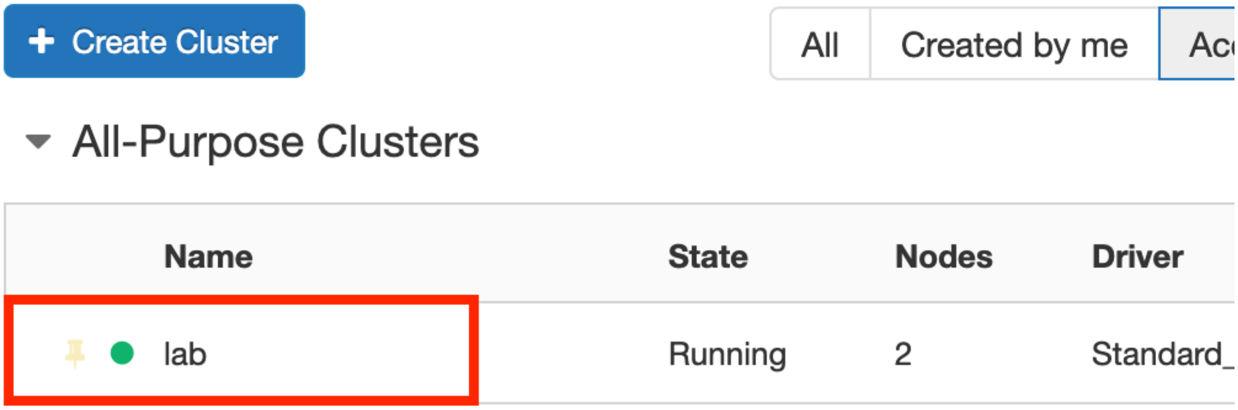
Task 3: Install libraries on the Azure Databricks Cluster

The notebooks you will run depends on certain Python libraries that will need to be installed in your cluster. The following steps walk you through adding these dependencies.

1. From the left-hand menu in your Workspace, select **Clusters**.



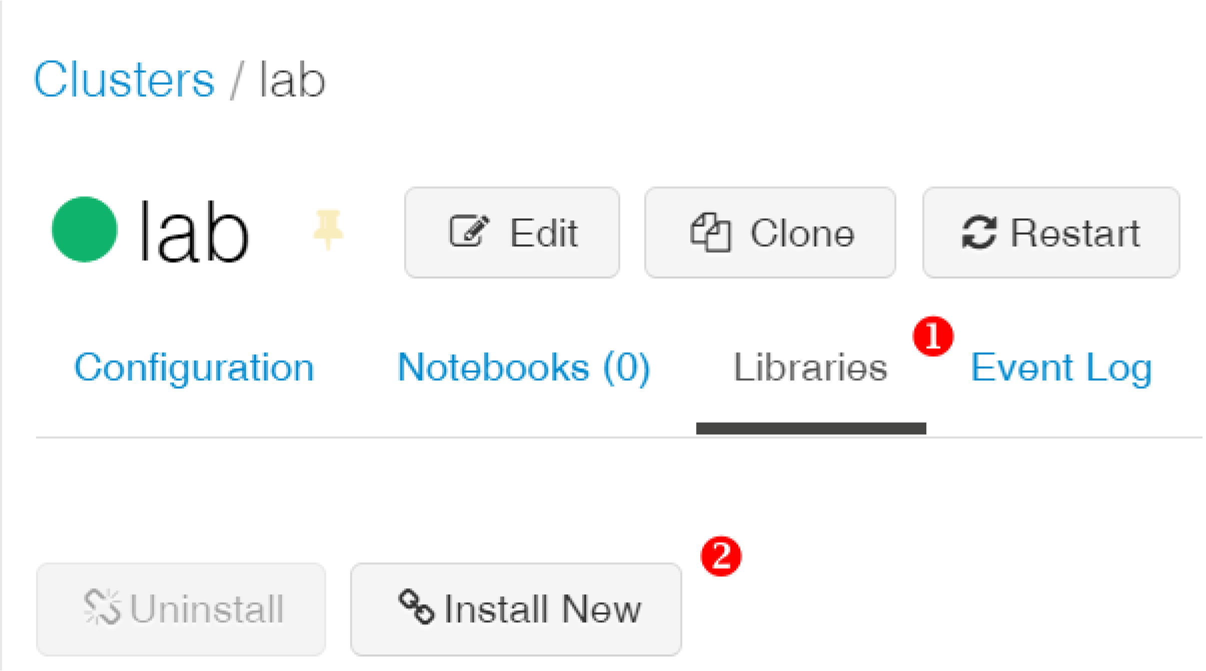
2. In the list of clusters, select your cluster. Make sure the state of the cluster is `Running`.



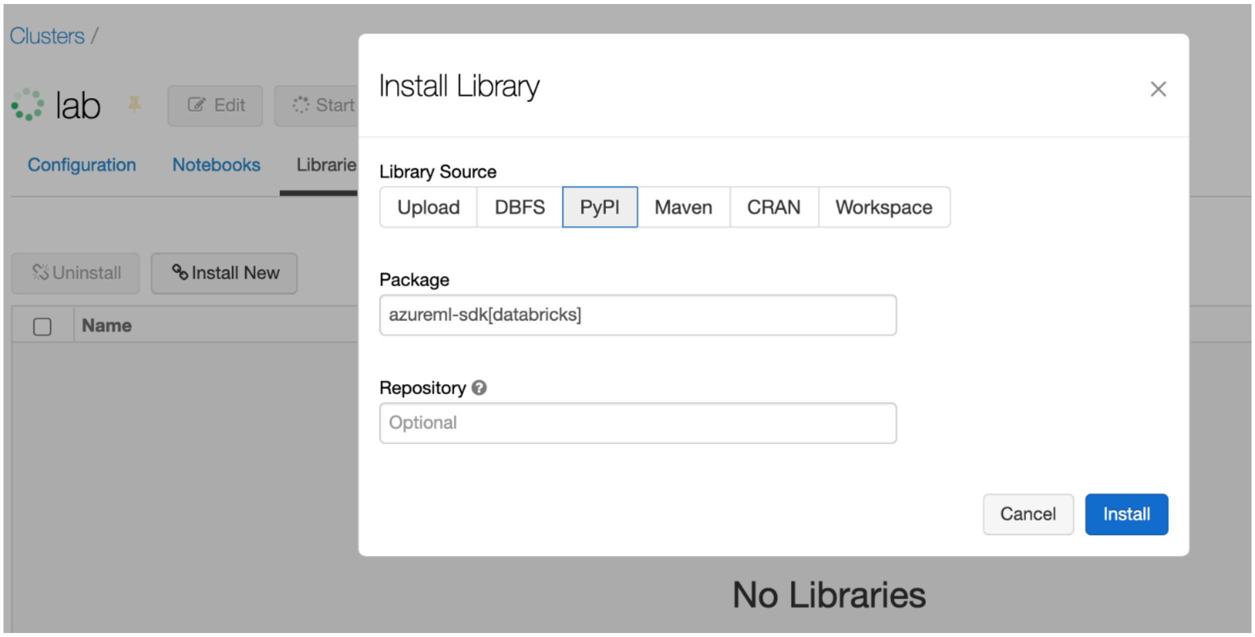
All-Purpose Clusters

Name	State	Nodes	Driver
● lab	Running	2	Standard_

3. Select the **Libraries** link and then select **Install New**.



4. In the Library Source, select **PyPi** and in the Package text box type `azureml-sdk[databricks]` and select **Install**.



Library Source: PyPI

Package: azureml-sdk[databricks]

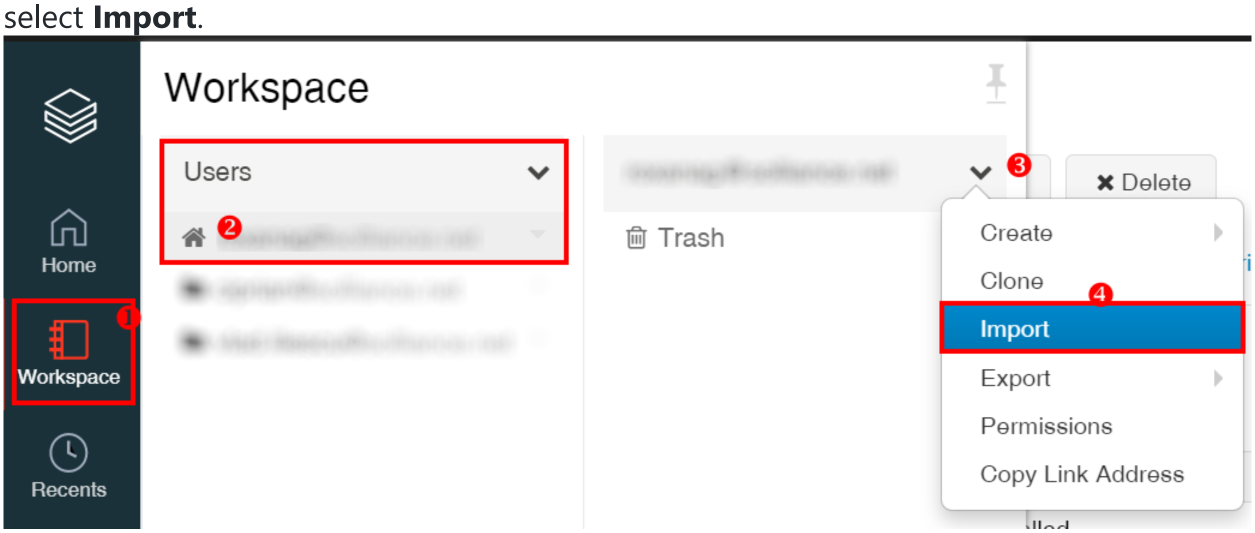
Repository: Optional

5. An entry for azureml-sdk will appear in the list. The install will progress first with a status of installing followed by the status of installed.

Task 4: Upload the Databricks notebook archive

1. Within the Azure Databricks Workspace, using the command bar on the left, select Workspace, Users and select your username (the entry with house icon).

2. In the blade that appears, select the downwards pointing chevron next to your name, and
   select **Import**.

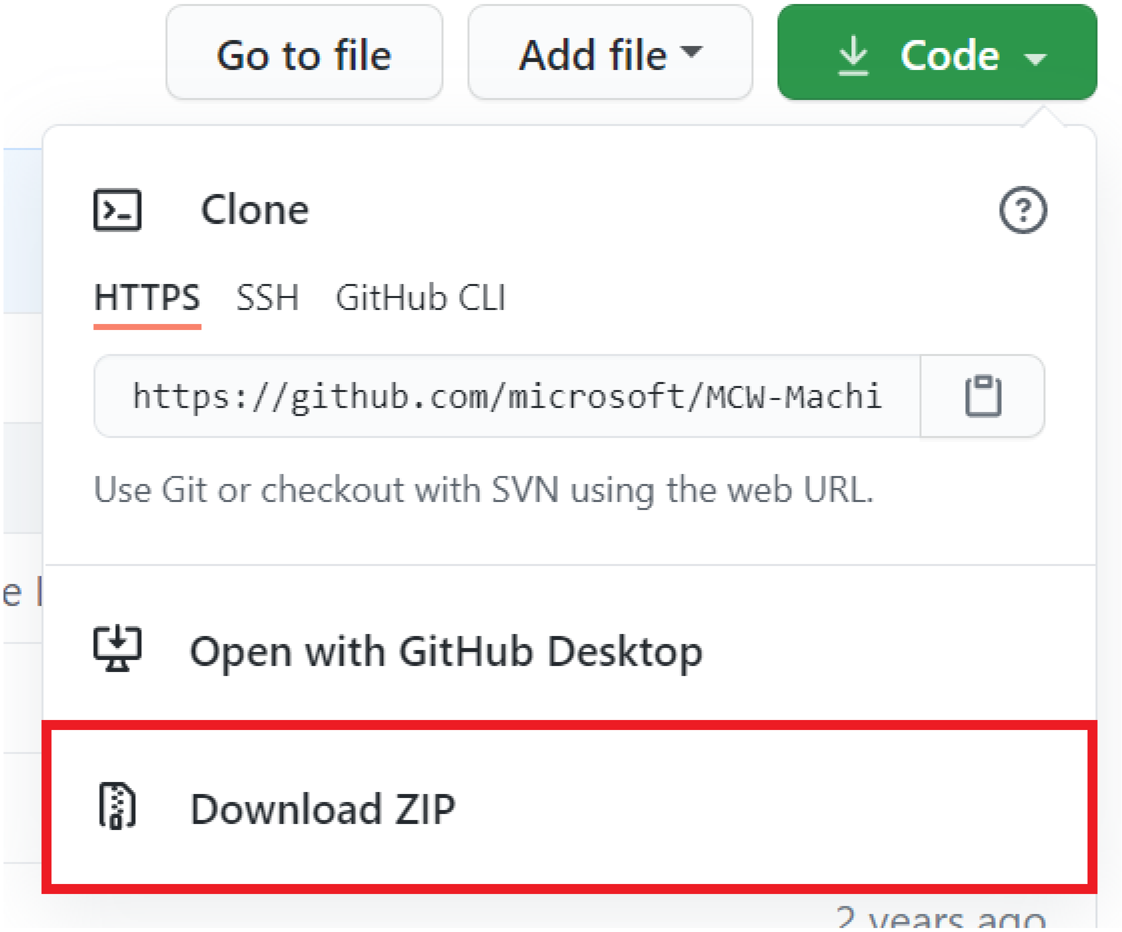


3. On the Import Notebooks dialog, select **URL** and paste in the following URL:

```
https://github.com/microsoft/MCW-Machine-Learning/blob/master/Hands-on%20lab/notebooks/AI%20with%20Databricks%20and%20AML.dbc?raw=true
```

4. Select **Import**

   Note: If you are facing difficulty with this URL, you can also consider cloning this repository locally, and uploading the file from your computer to databricks. To do this, download the zip archive of this repository, and extract it to a known location. Then, from the repository root, navigate to `Hands-on lab/notebooks/AI with Databricks and AML.dbc`.



https://github.com/microsoft/MCW-Machi

Use Git or checkout with SVN using the web URL.

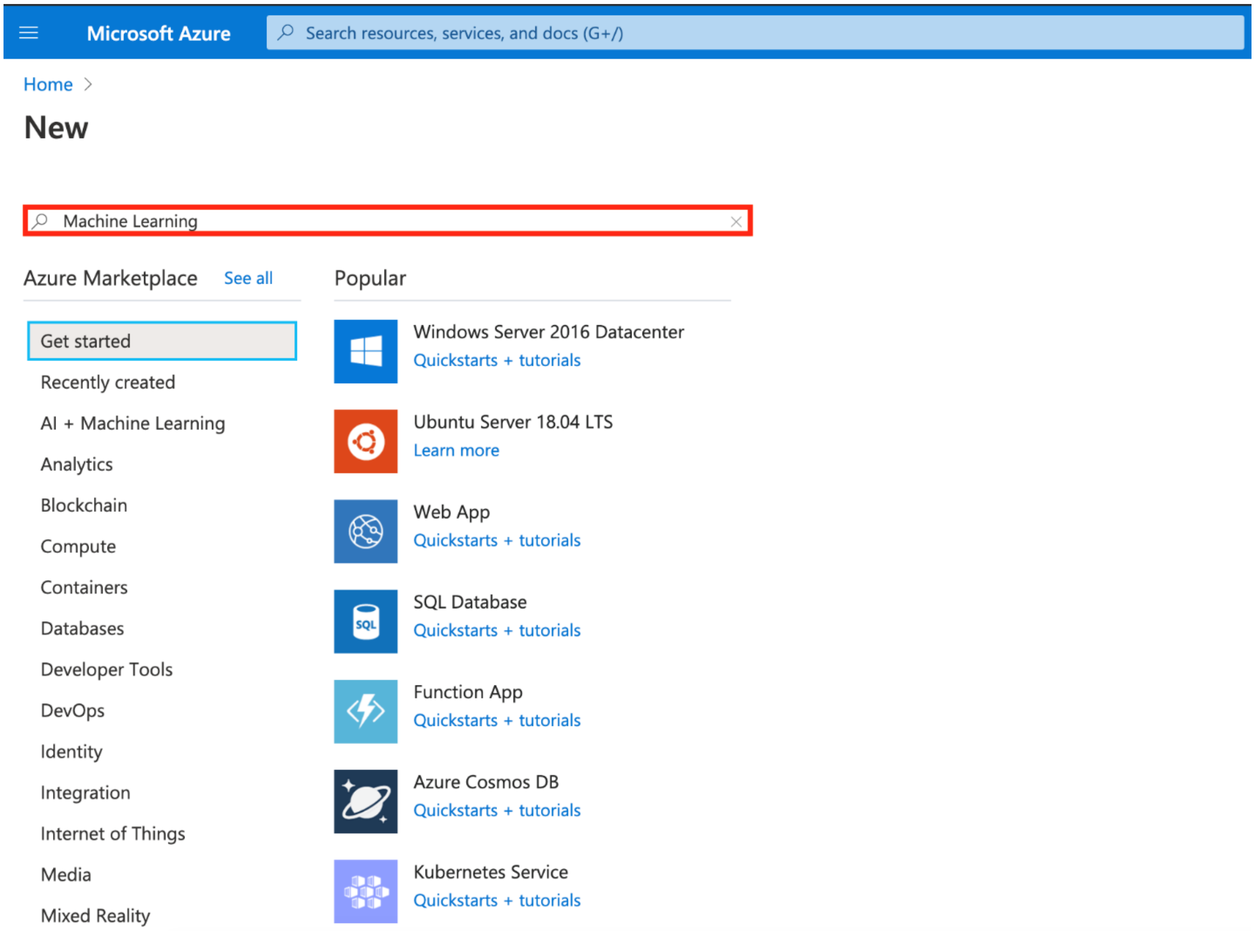
5. A folder named after the archive should appear. Select that folder.

6. The folder will contain one or more notebooks. These are the notebooks you will use in the completing this lab.

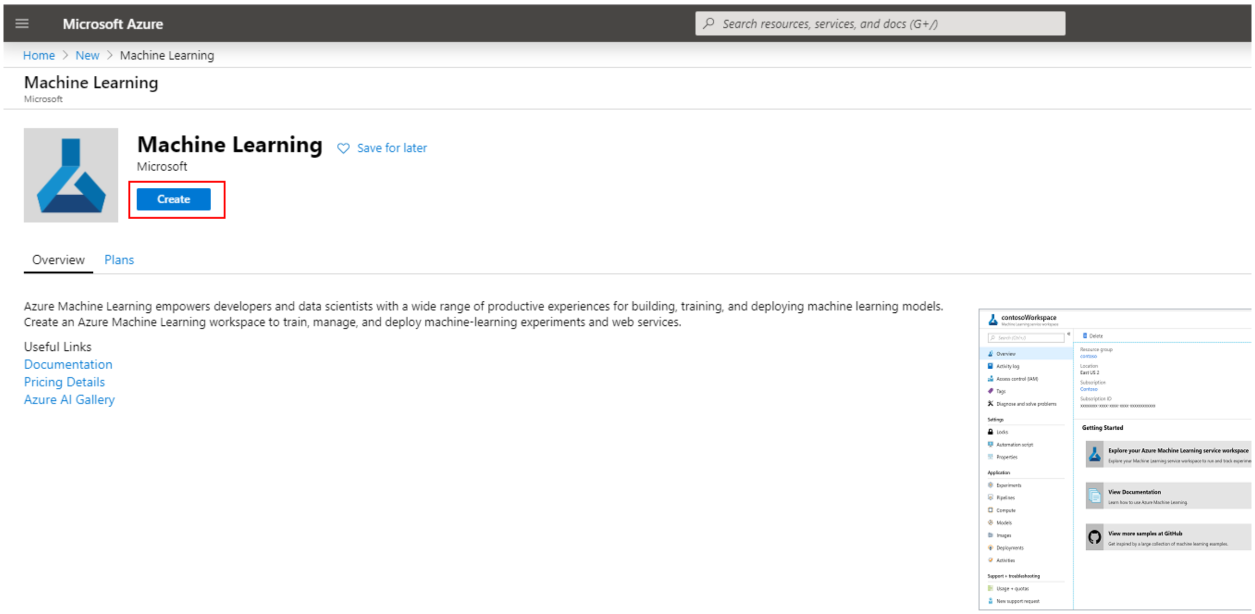
Task 5: Create your Azure Machine Learning Workspace

1. In the Azure Portal, select **+ Create a resource**, then type `Machine Learning` into the search

bar.



Microsoft Azure
Search resources, services, and docs (G+/)
Home >
New
Machine Learning
Azure Marketplace
See all
Popular
Get started
Recently created
AI + Machine Learning
Analytics
Blockchain
Compute
Containers
Databases
Developer Tools
DevOps
Identity
Integration
Internet of Things
Media
Mixed Reality
Windows Server 2016 Datacenter
Quickstarts + tutorials
Ubuntu Server 18.04 LTS
Learn more
Web App
Quickstarts + tutorials
SQL Database
Quickstarts + tutorials
Function App
Quickstarts + tutorials
Azure Cosmos DB
Quickstarts + tutorials
Kubernetes Service
Quickstarts + tutorials

2. 

Microsoft Azure
Search resources, services, and docs (G+/)
Home > New > Machine Learning

Machine Learning
Microsoft

Machine Learning ♡ Save for later
Microsoft

Create

Overview Plans

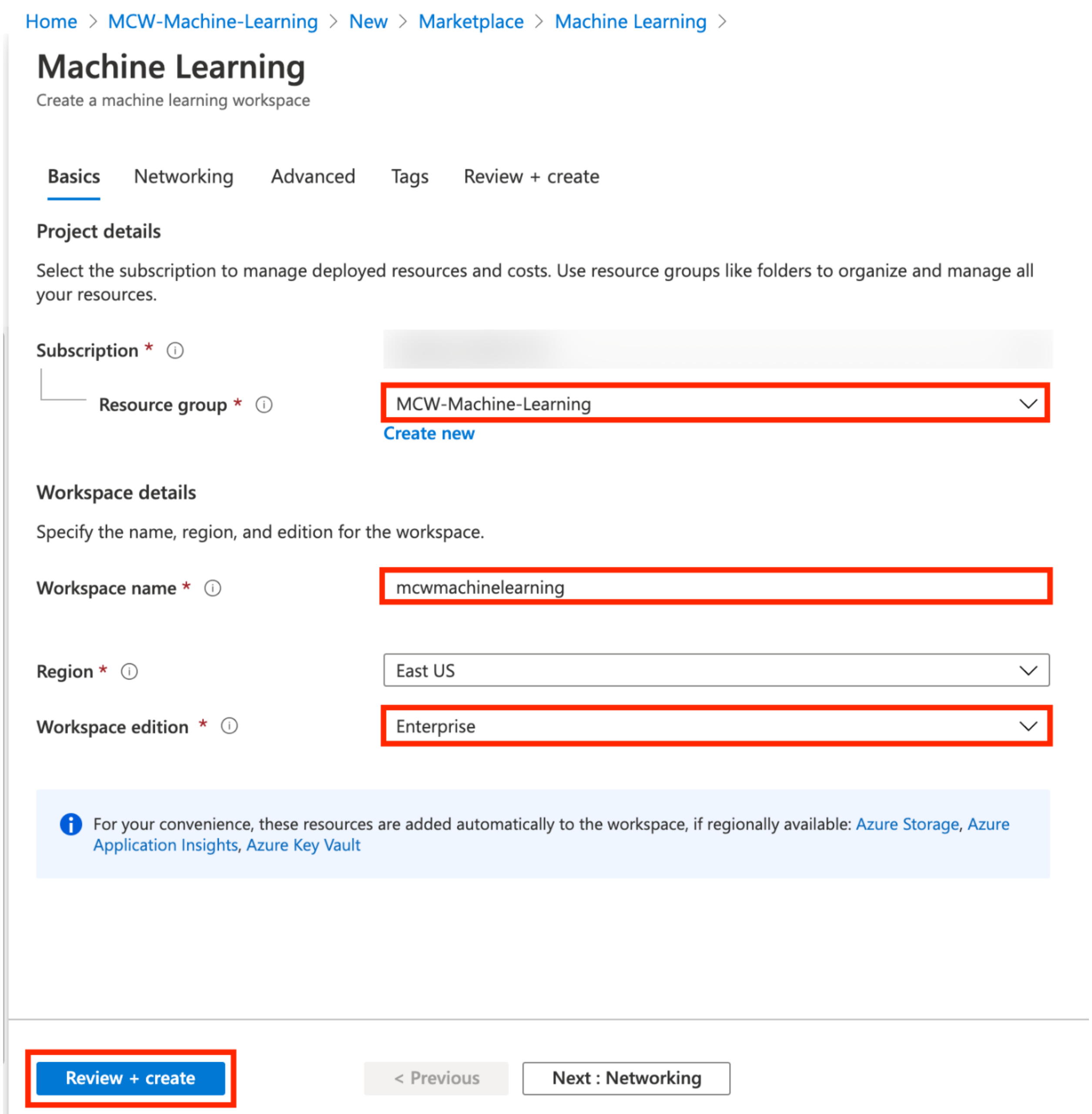
Azure Machine Learning empowers developers and data scientists with a wide range of productive experiences for building, training, and deploying machine learning models. Create an Azure Machine Learning workspace to train, manage, and deploy machine-learning experiments and web services.

Useful Links
Documentation
Pricing Details
Azure AI Gallery

contosoWorkspace
Getting Started
Explore your Azure Machine Learning service workspace
View Documentation
View more samples at GitHub

3. In the Create Machine Learning Workspace dialog that appears, provide the following values:

   - **Workspace Name**: `mcwmachinelearning`
   - **Subscription**: Choose your Azure subscription.
   - **Resource group**: Select the resource group in which you deployed your Azure Databricks workspace.
   - **Location**: Choose a region closest to you (it is OK if the Azure Databricks Workspace and the Azure Machine Learning Workspace are in different locations).
   - **Workspace edition**: Select `Enterprise`.
     If did not showing **Workspace edition**, please ignore it.

4. 

Home > MCW-Machine-Learning > New > Marketplace > Machine Learning >

Machine Learning

Create a machine learning workspace

Project details

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

Subscription * [redacted]

Resource group * MCW-Machine-Learning

Workspace details

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

Workspace name * mcwmachinelearning

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

5. **Note**: The option to set the **Workspace edition** may not be present.

6. Select **Review + Create** and then select **Create** when the form values passes validation.

You should follow all this steps provided *before* attending the Hands-on lab.

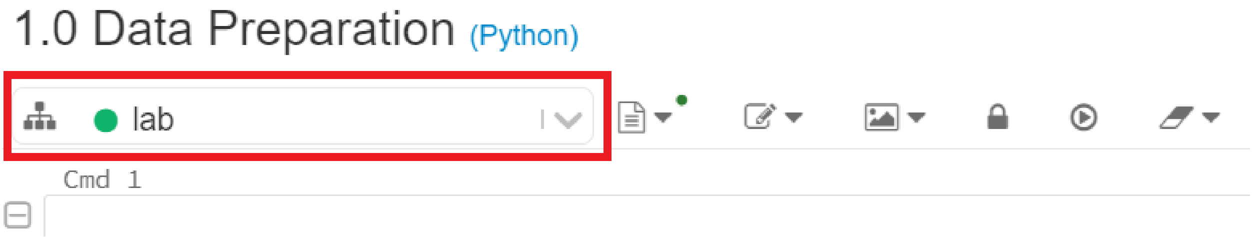
Exercise 1: Data exploration and preprocessing

Duration: 40 minutes

Understanding data through data exploration is one of the core challenges faced today by data engineers and data scientists. Using raw data for modeling can produce misleading results, since data is often noisy and unreliable, and may be missing values. In this exercise, you will explore the raw data, transform and register the dataset in the Datastore. You will use it to train a forecasting model later in this hands-on-lab. The data preparation steps will be performed on the Azure Databricks cluster.

Task 1: Load, explore and prepare the dataset using an Azure Databricks notebook

1. Browse to your Azure Databricks Workspace and navigate to `AI with Databricks and AML \ 1.0 Data Preparation`. This is the notebook you will step through executing in this exercise.
2. When you execute a notebook, you will need to attach it to a cluster. You can specify that the **lab** cluster should be used in the upper left-hand corner of the notebook.



1.0 Data Preparation (Python)

3. Follow the instructions within the notebook to complete the exercise. Press Shift+Enter to execute a cell.

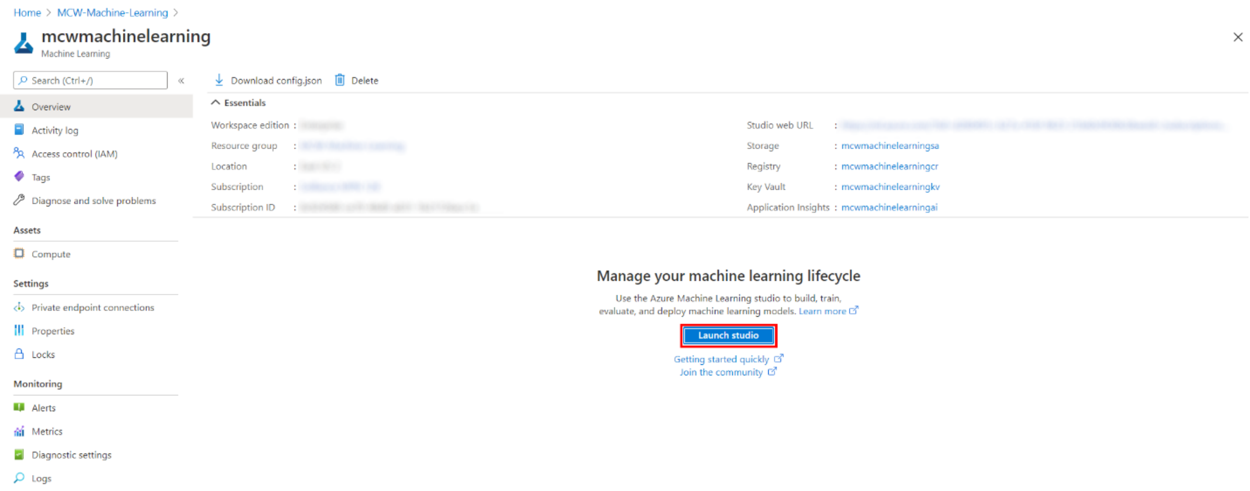
Exercise 2: Creating a forecast model using automated machine learning

Duration: 60 minutes

In this exercise, you will create a model that predicts battery failure from time-series data using the visual interface to automated machine learning in an Azure Machine Learning workspace. You will deploy the best model to Azure Container Instances (ACI), and you will also perform batch scoring from a Databricks notebooks.

Task 1: Create an automated machine learning experiement

1. Navigate to your Azure Machine Learning workspace in the Azure Portal. Select **Try the new Azure Machine Learning studio, Launch now**.



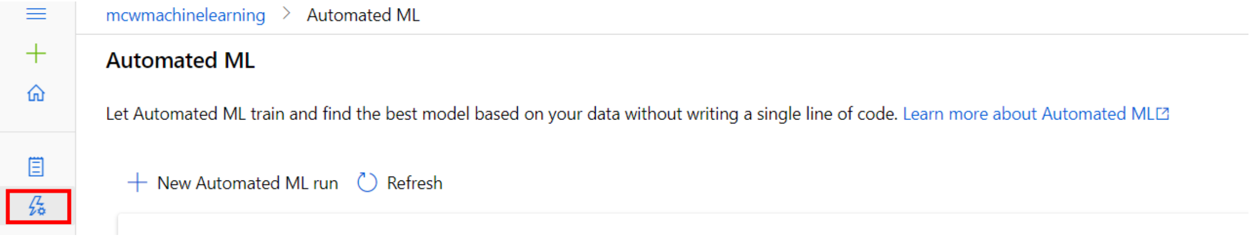
Workspace edition : [redacted]
Resource group : [redacted]
Location : [redacted]
Subscription : [redacted]
Subscription ID : [redacted]
Studio web URL : [redacted]
Storage : mcwmachinelearningsa
Registry : mcwmachinelearningcr
Key Vault : mcwmachinelearningkv
Application Insights : mcwmachinelearningai

Manage your machine learning lifecycle

Use the Azure Machine Learning studio to build, train, evaluate, and deploy machine learning models. Learn more

Note: Alternatively, you can sign-in directly to the Azure Machine Learning studio portal.

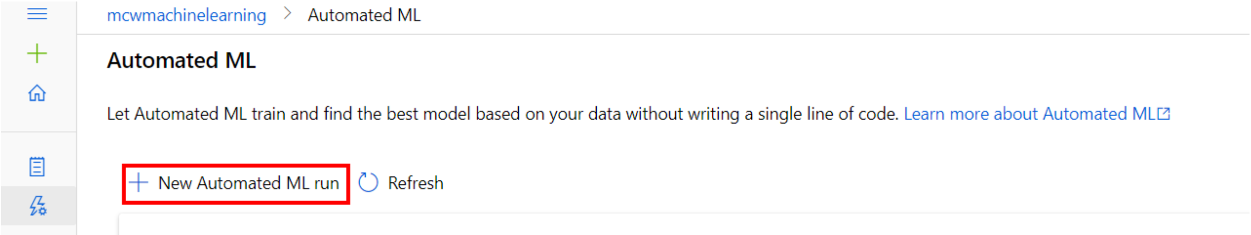
2. Select **Automated ML icon** in the left navigation bar.



Automated ML

Let Automated ML train and find the best model based on your data without writing a single line of code. Learn more about Automated ML

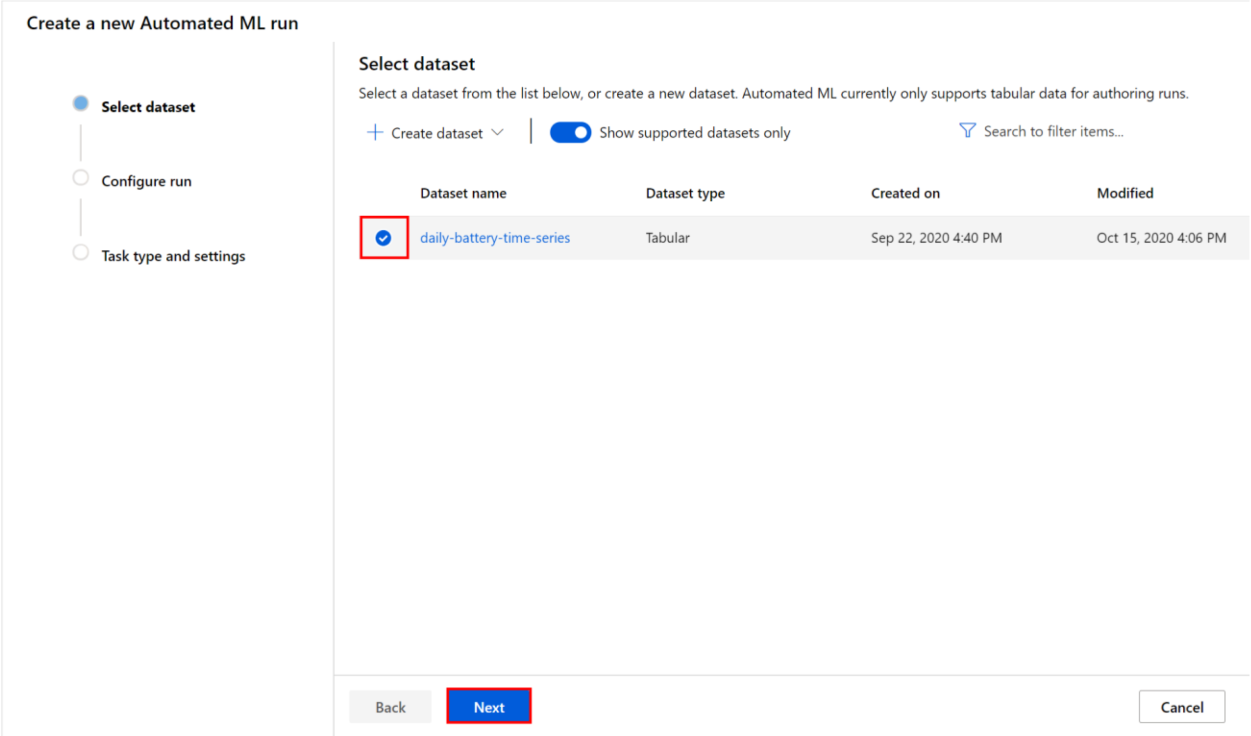
3. Select **+ New automated ML run**.



Automated ML

Let Automated ML train and find the best model based on your data without writing a single line of code. Learn more about Automated ML

4. Select the `daily-battery-time-series` dataset from the list of registered datasets and then select **Next**. (This dataset was registered as a final step of the previous exercise, from the Azure Databricks notebook.)



Create a new Automated ML run

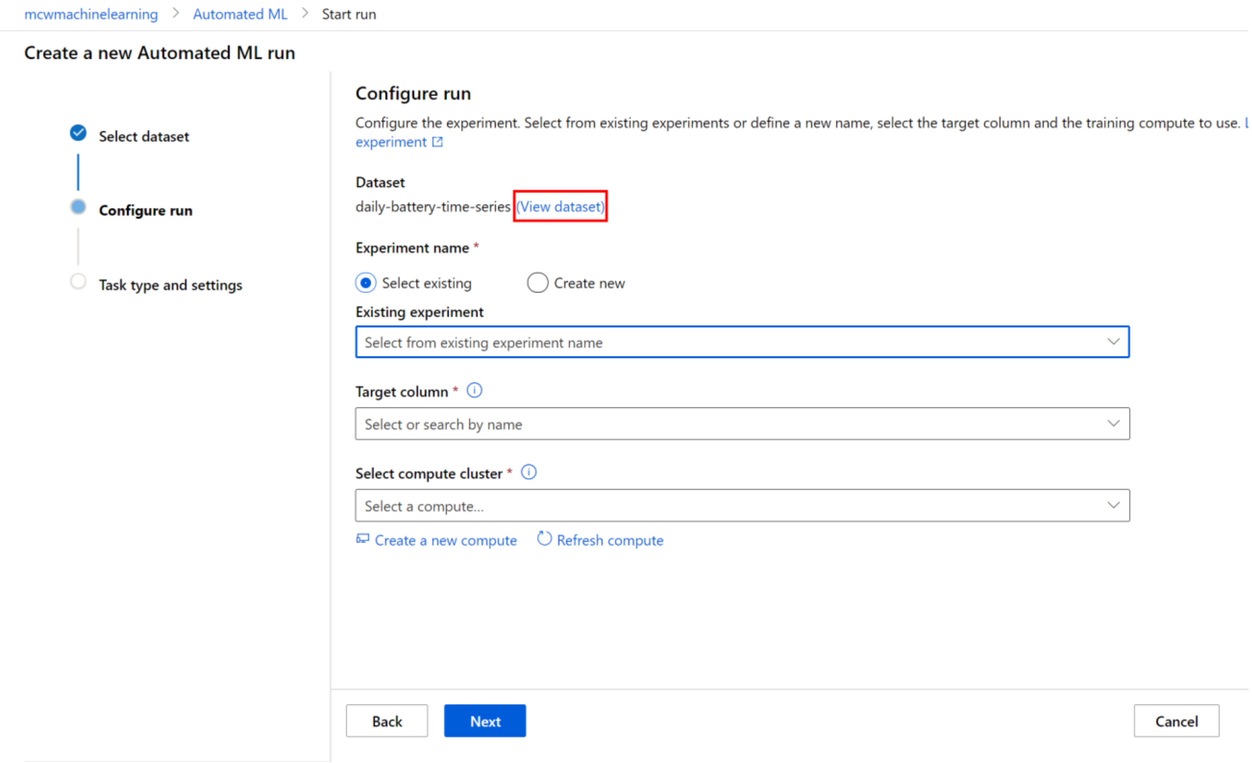
Select dataset

Select a dataset from the list below, or create a new dataset. Automated ML currently only supports tabular data for authoring runs.

	Dataset name	Dataset type	Created on	Modified
☑	daily-battery-time-series	Tabular	Sep 22, 2020 4:40 PM	Oct 15, 2020 4:06 PM

5. Review the dataset details in the `Configure run` section, by selecting the **View dataset** link next to the dataset

name.



mcwmachinelearning > Automated ML > Start run

Create a new Automated ML run

- Select dataset
- Configure run
- Task type and settings

Configure run

Configure the experiment. Select from existing experiments or define a new name, select the target column and the training compute to use. l experiment

Dataset
daily-battery-time-series (View dataset)

Experiment name *
● Select existing ○ Create new

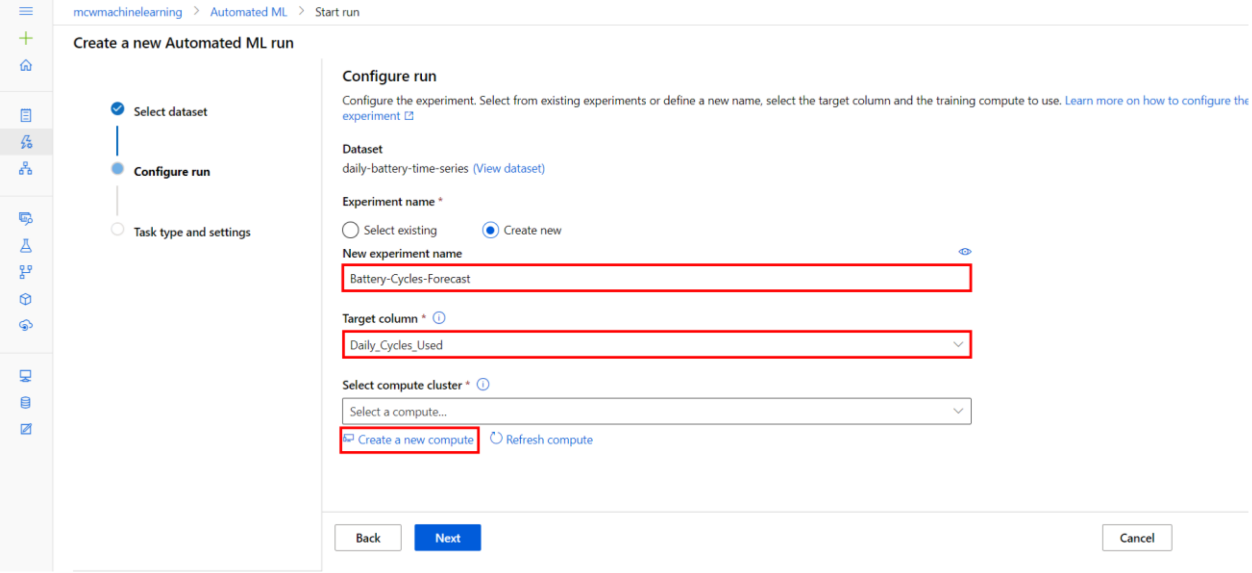
Existing experiment
Select from existing experiment name

Target column *
Select or search by name

Select compute cluster *
Select a compute...

Create a new compute Refresh compute

6. Provide the experiment name: `Battery-Cycles-Forecast` and select **Daily_Cycles_Used** as target column. Select **Create a new compute**.



mcwmachinelearning > Automated ML > Start run

Create a new Automated ML run

- Select dataset
- Configure run
- Task type and settings

Configure run

Configure the experiment. Select from existing experiments or define a new name, select the target column and the training compute to use. Learn more on how to configure the experiment

Dataset
daily-battery-time-series (View dataset)

Experiment name *
○ Select existing ● Create new

New experiment name
Battery-Cycles-Forecast

Target column *
Daily_Cycles_Used

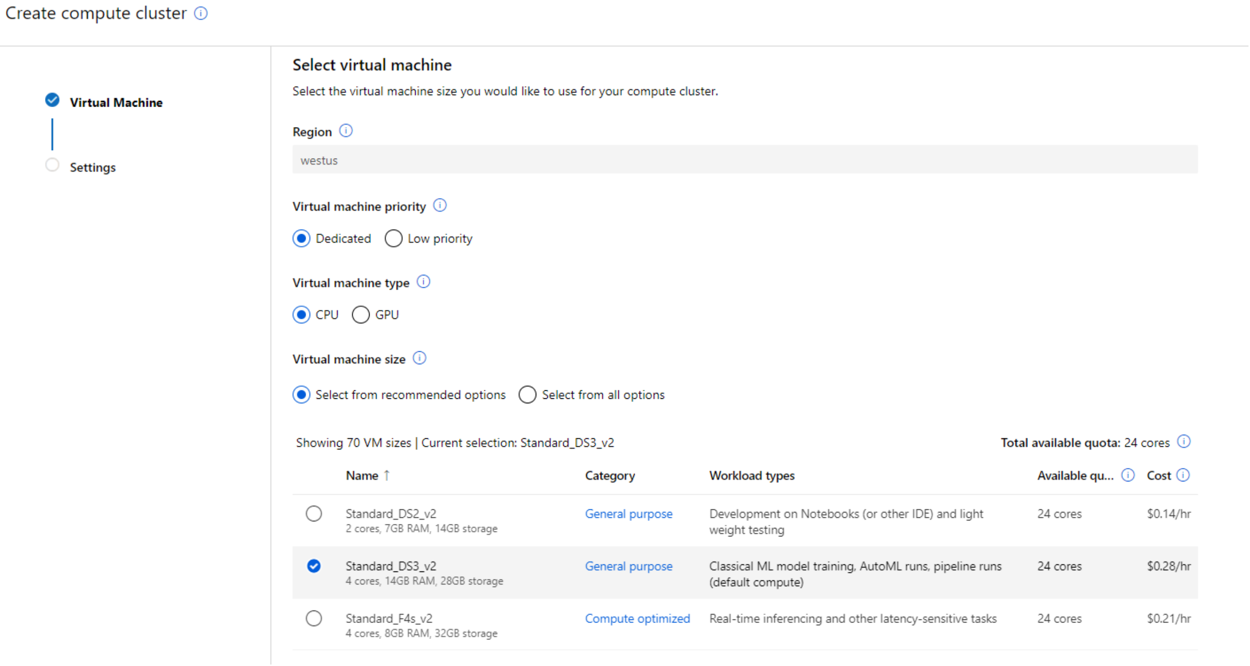
Select compute cluster *
Select a compute...

Create a new compute Refresh compute

7. For the new compute, provide the following values and then select **Next**:

   - **Virtual machine priority**: `Dedicated`
   - **Virtual machine type**: `CPU`

- **Virtual machine size**: `Select from recommended options --> Standard_DS3_v2`



Create compute cluster ⓘ

- ● Virtual Machine
- ○ Settings

Select virtual machine

Select the virtual machine size you would like to use for your compute cluster.

Region ⓘ

westus

Virtual machine priority ⓘ

● Dedicated ○ Low priority

Virtual machine type ⓘ

● CPU ○ GPU

Virtual machine size ⓘ

● Select from recommended options ○ Select from all options

Showing 70 VM sizes | Current selection: Standard_DS3_v2

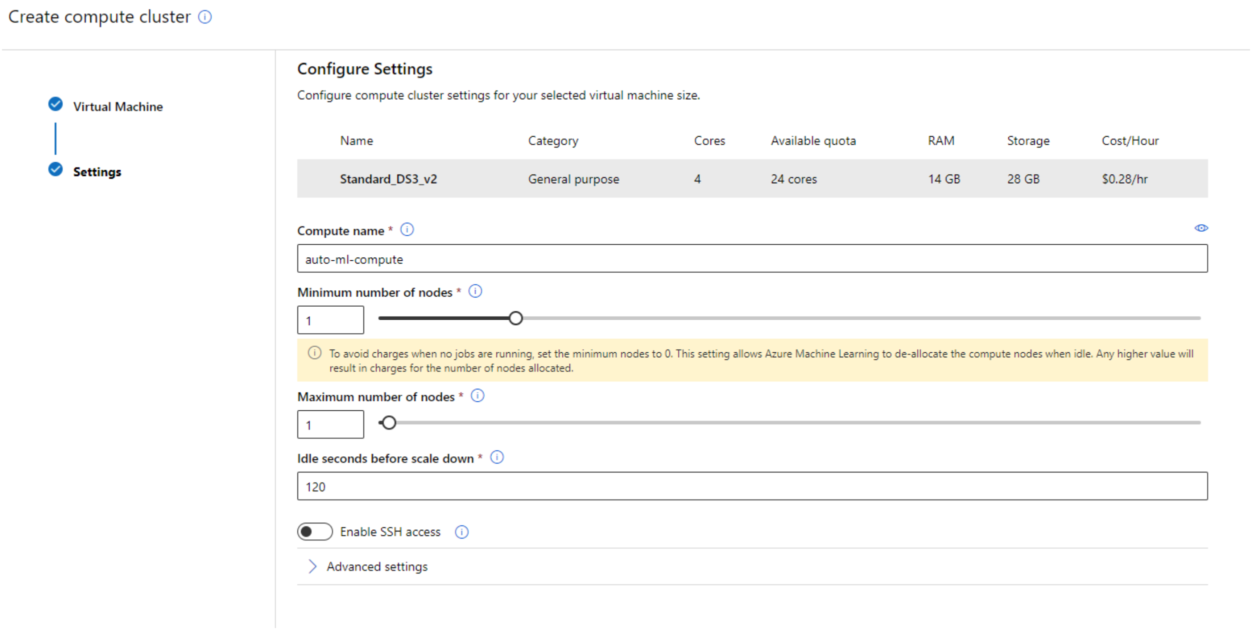
Total available quota: 24 cores ⓘ

	Name ↑	Category	Workload types	Available qu... ⓘ	Cost ⓘ
○	Standard_DS2_v2 2 cores, 7GB RAM, 14GB storage	General purpose	Development on Notebooks (or other IDE) and light weight testing	24 cores	$0.14/hr
●	Standard_DS3_v2 4 cores, 14GB RAM, 28GB storage	General purpose	Classical ML model training, AutoML runs, pipeline runs (default compute)	24 cores	$0.28/hr
○	Standard_F4s_v2 4 cores, 8GB RAM, 32GB storage	Compute optimized	Real-time inferencing and other latency-sensitive tasks	24 cores	$0.21/hr

8. On the next page, provide the following parameters for your compute cluster. Then, select **Create**.

   - **Compute name**: `auto-ml-compute`
   - **Minimum number of nodes**: `1`
   - **Maximum number of nodes**: `1`
   - **Idle seconds before scale down**: `120`

   Note: The creation of the new compute may take several minutes. Once the process is completed, select **Next** in the `Configure run` section.



Create compute cluster ⓘ

Configure Settings

Configure compute cluster settings for your selected virtual machine size.

Name	Category	Cores	Available quota	RAM	Storage	Cost/Hour
Standard_DS3_v2	General purpose	4	24 cores	14 GB	28 GB	$0.28/hr

Compute name * ⓘ: auto-ml-compute

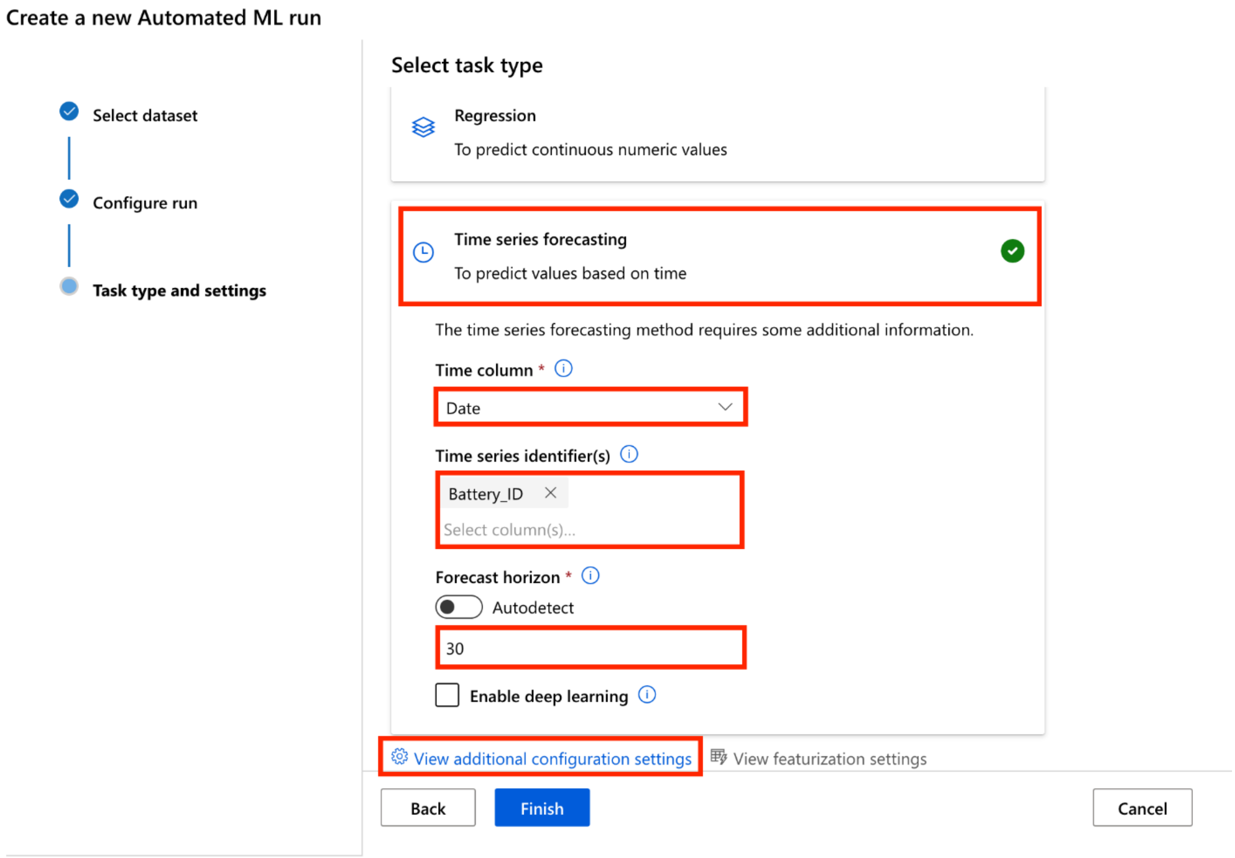
Minimum number of nodes * ⓘ: 1

ⓘ To avoid charges when no jobs are running, set the minimum nodes to 0. This setting allows Azure Machine Learning to de-allocate the compute nodes when idle. Any higher value will result in charges for the number of nodes allocated.

Maximum number of nodes * ⓘ: 1

Idle seconds before scale down * ⓘ: 120

9. Select the `Time series forecasting` task type and provide the following values and then select **View additional configuration settings**:
   - **Time column**: `Date`
   - **Time series identifier(s)**: `Battery_ID`
   - **Forecast horizon**: `30`



Create a new Automated ML run

Select task type

Regression
To predict continuous numeric values

Time series forecasting
To predict values based on time

The time series forecasting method requires some additional information.

Time column *: Date

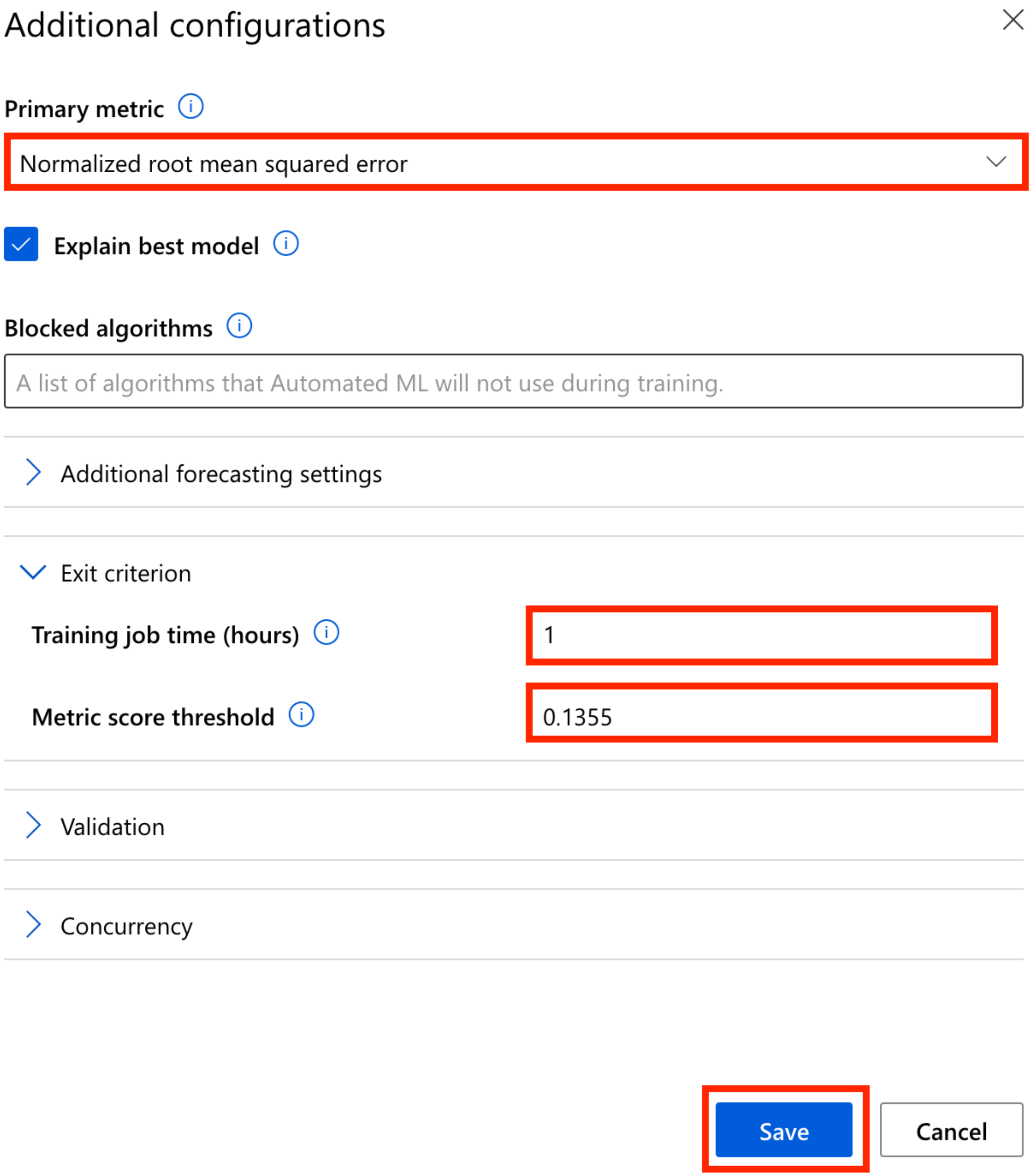
Time series identifier(s): Battery_ID

Forecast horizon *: Autodetect off, 30

□ Enable deep learning

10. For the automated machine learning run additional configurations, provide the following values and then select **Save**:

   - **Primary metric**: `Normalized root mean squared error`
   - **Explain best model**: Selected
   - **Training job time (hours)** (in the `Exit criterion` section): enter `1` as this is the lowest value currently accepted.
   - **Metric score threshold**: enter `0.1355`. When this threshold value will be reached for an iteration metric the training job will terminate.



Additional configurations

Primary metric: Normalized root mean squared error

☑ Explain best model

Blocked algorithms: A list of algorithms that Automated ML will not use during training.

Exit criterion

Training job time (hours): 1

Metric score threshold: 0.1355

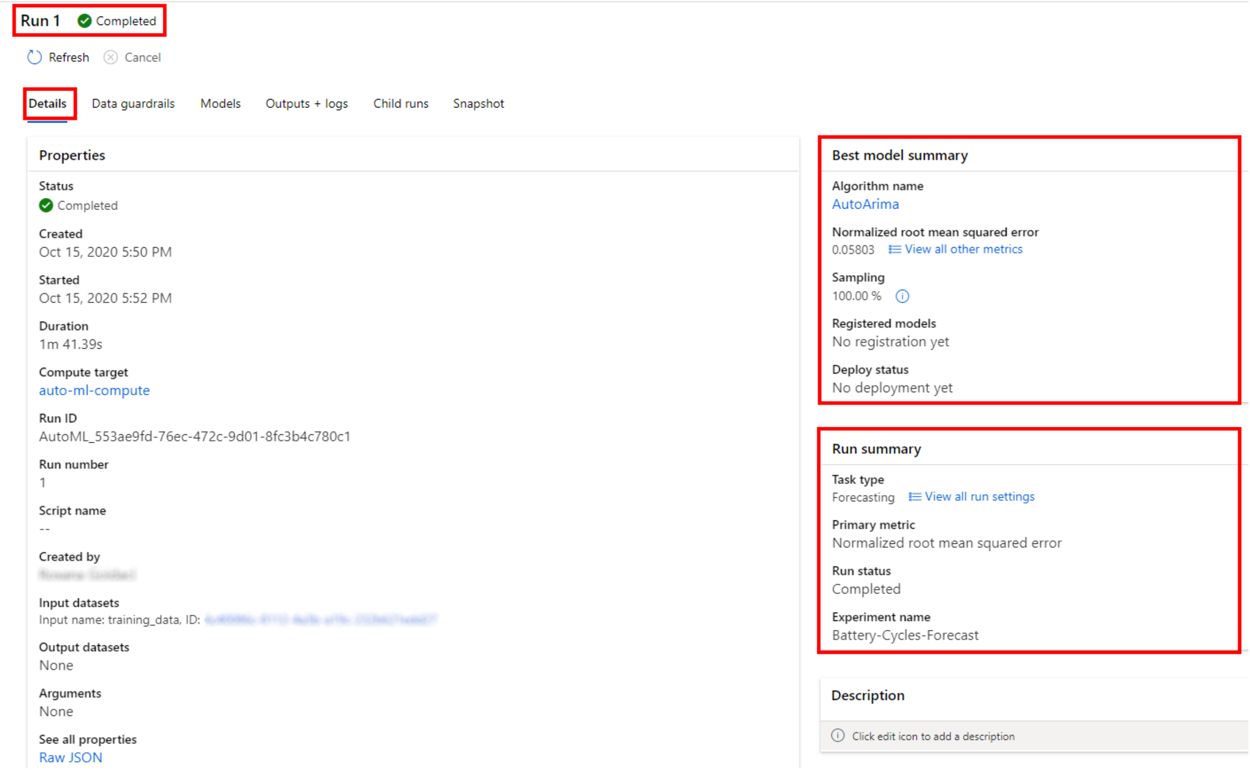
Note: We are setting a metric score threshold to limit the training time. In practice, for initial experiments, you will typically only set the training job time to allow AutoML to discover the best algorithm to use for your specific data.

11. Select **Finish** to start the new automated machine learning run.

Note: The experiment should run for up to 10 minutes. If the run time exceeds 15 minutes, cancel the run and start a new one (steps 3, 9, 10). Make sure you provide a higher value for `Metric score threshold` in step 10.

Task 2: Review the experiment run results

1. Once the experiment completes, select `Details` to examine the details of the run containing information about the best model and the run summary.



Run 1 Completed

Refresh Cancel

Details | Data guardrails | Models | Outputs + logs | Child runs | Snapshot

Properties

Status
Completed

Created
Oct 15, 2020 5:50 PM

Started
Oct 15, 2020 5:52 PM

Duration
1m 41.39s

Compute target
auto-ml-compute

Run ID
AutoML_553ae9fd-76ec-472c-9d01-8fc3b4c780c1

Run number
1

Script name
--

Created by
[redacted]

Input datasets
Input name: training_data, ID: [redacted]

Output datasets
None

Arguments
None

See all properties
Raw JSON

Best model summary

Algorithm name
AutoArima

Normalized root mean squared error
0.05803 View all other metrics

Sampling
100.00 %

Registered models
No registration yet

Deploy status
No deployment yet

Run summary

Task type
Forecasting View all run settings

Primary metric
Normalized root mean squared error

Run status
Completed

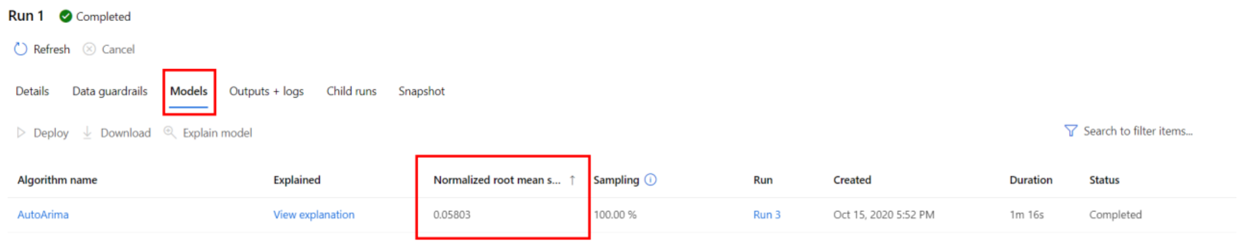
Experiment name
Battery-Cycles-Forecast

Description

Click edit icon to add a description

2. Select `Models` to see a table view of different iterations and the `Normalized root mean squared error` score for each iteration. Note that the normalized root mean square error measures the error between the predicted value and actual value. In this case, the model with the lowest normalized root mean square error is the best model. Note that Azure Machine Learning Python SDK updates over time and gives you the best performing model at the time you run the experiment. Thus, it is possible that the best model you observe can be different than the one shown

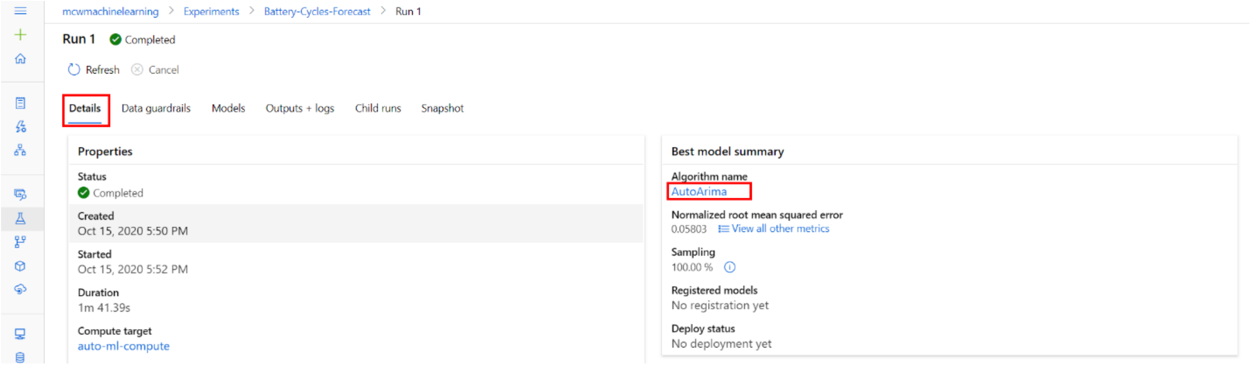
below.



Run 1 ✓ Completed

Algorithm name	Explained	Normalized root mean s... ↑	Sampling ⓘ	Run	Created	Duration	Status
AutoArima	View explanation	0.05803	100.00 %	Run 3	Oct 15, 2020 5:52 PM	1m 16s	Completed

3. Return to the details of your experiment run and select the best model **Algorithm name**.



mcwmachinelearning > Experiments > Battery-Cycles-Forecast > Run 1

Run 1 ✓ Completed

Properties

Status
✓ Completed

Created
Oct 15, 2020 5:50 PM

Started
Oct 15, 2020 5:52 PM

Duration
1m 41.39s

Compute target
auto-ml-compute

Best model summary

Algorithm name
AutoArima

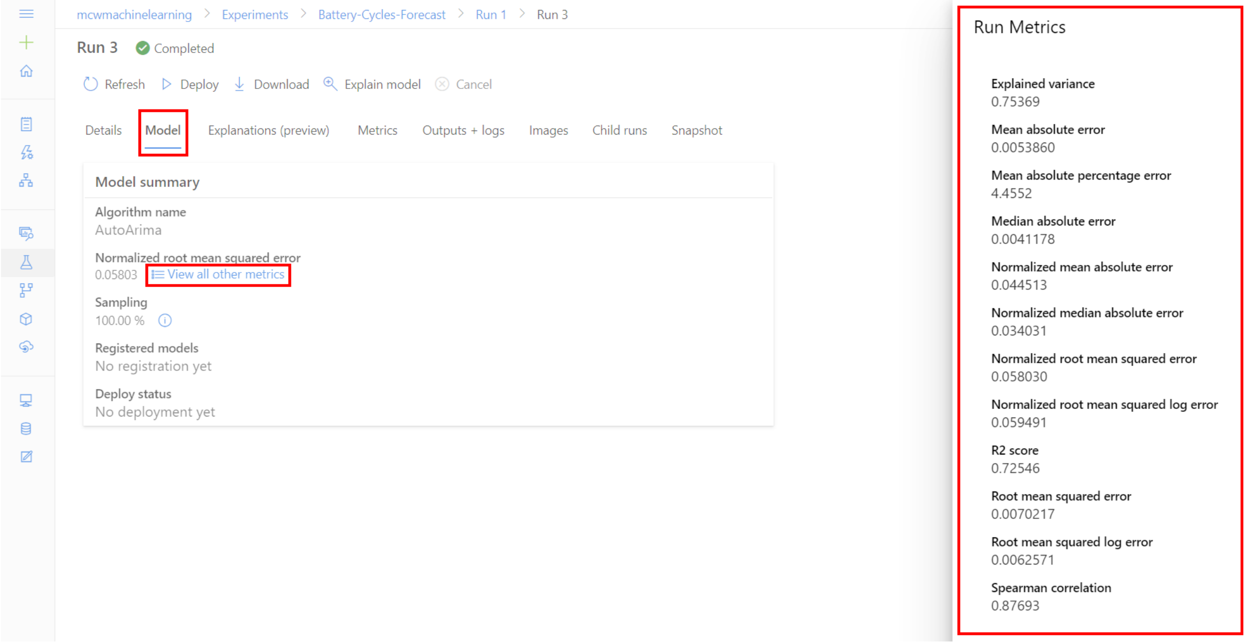
Normalized root mean squared error
0.05803 View all other metrics

Sampling
100.00 % ⓘ

Registered models
No registration yet

Deploy status
No deployment yet

4. From the `Model` tab, select **View all other metrics** to review the various `Run Metrics` to evaluate the model performance.



mcwmachinelearning > Experiments > Battery-Cycles-Forecast > Run 1 > Run 3

Run 3 ✓ Completed

Model summary

Algorithm name
AutoArima

Normalized root mean squared error
0.05803 View all other metrics

Sampling
100.00 % ⓘ

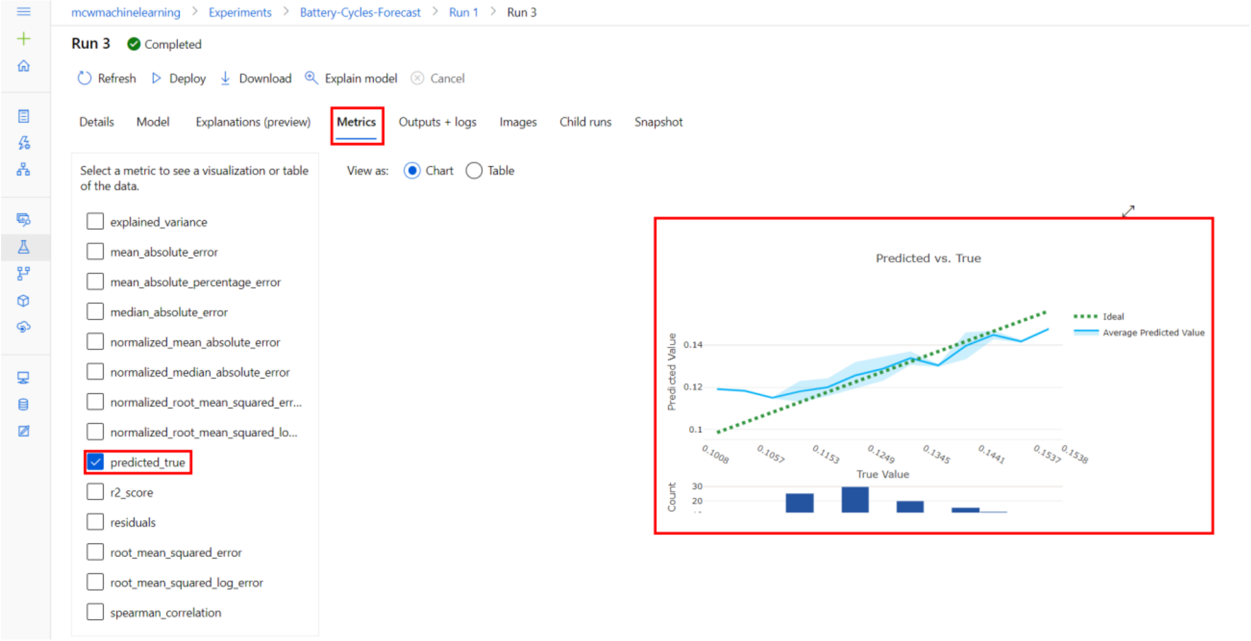
Registered models
No registration yet

Deploy status
No deployment yet

Run Metrics

Explained variance	0.75369
Mean absolute error	0.0053860
Mean absolute percentage error	4.4552
Median absolute error	0.0041178
Normalized mean absolute error	0.044513
Normalized median absolute error	0.034031
Normalized root mean squared error	0.058030
Normalized root mean squared log error	0.059491
R2 score	0.72546
Root mean squared error	0.0070217
Root mean squared log error	0.0062571
Spearman correlation	0.87693

5. Next, select **Metrics, predicted_true** to review the model performance curve: `Predicted vs True`.



mcwmachinelearning > Experiments > Battery-Cycles-Forecast > Run 1 > Run 3

Run 3 ✓ Completed

Refresh | Deploy | Download | Explain model | Cancel

Details | Model | Explanations (preview) | Metrics | Outputs + logs | Images | Child runs | Snapshot

Select a metric to see a visualization or table of the data.

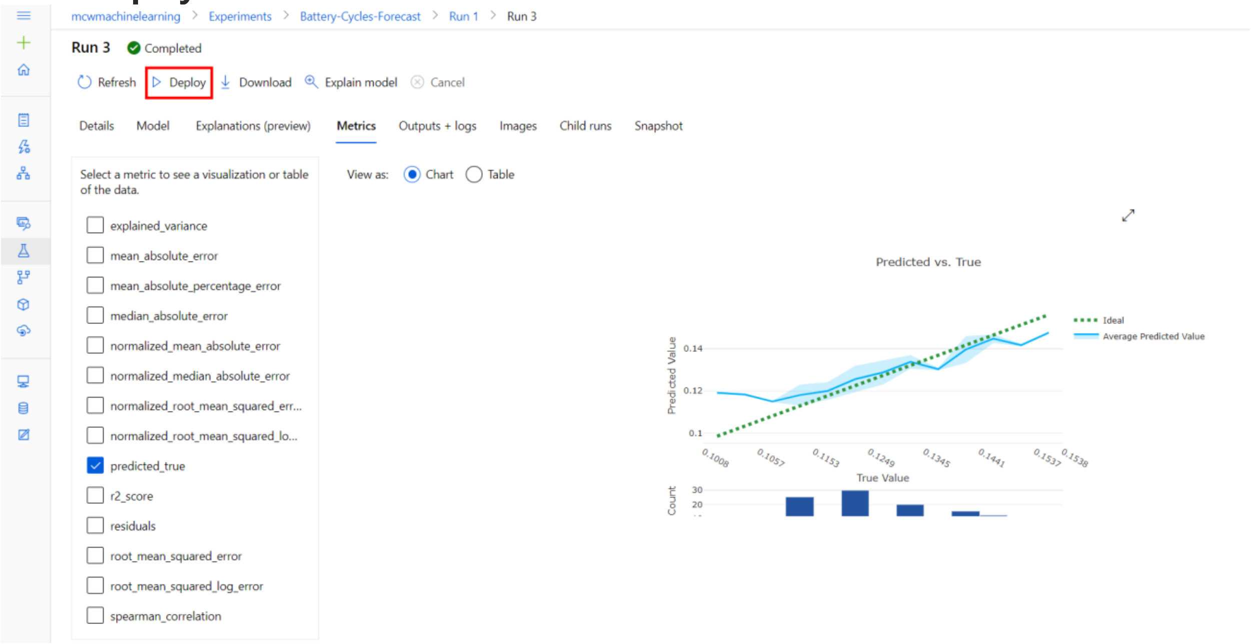
View as: ● Chart ○ Table

- □ explained_variance
- □ mean_absolute_error
- □ mean_absolute_percentage_error
- □ median_absolute_error
- □ normalized_mean_absolute_error
- □ normalized_median_absolute_error
- □ normalized_root_mean_squared_err...
- □ normalized_root_mean_squared_lo...
- ☑ predicted_true
- □ r2_score
- □ residuals
- □ root_mean_squared_error
- □ root_mean_squared_log_error
- □ spearman_correlation

Note: You may need to deselect the other metrics.

Task 3: Deploy the Best Model

1. From the top toolbar select **Deploy**.



mcwmachinelearning > Experiments > Battery-Cycles-Forecast > Run 1 > Run 3

Run 3 ✓ Completed

Refresh | Deploy | Download | Explain model | Cancel

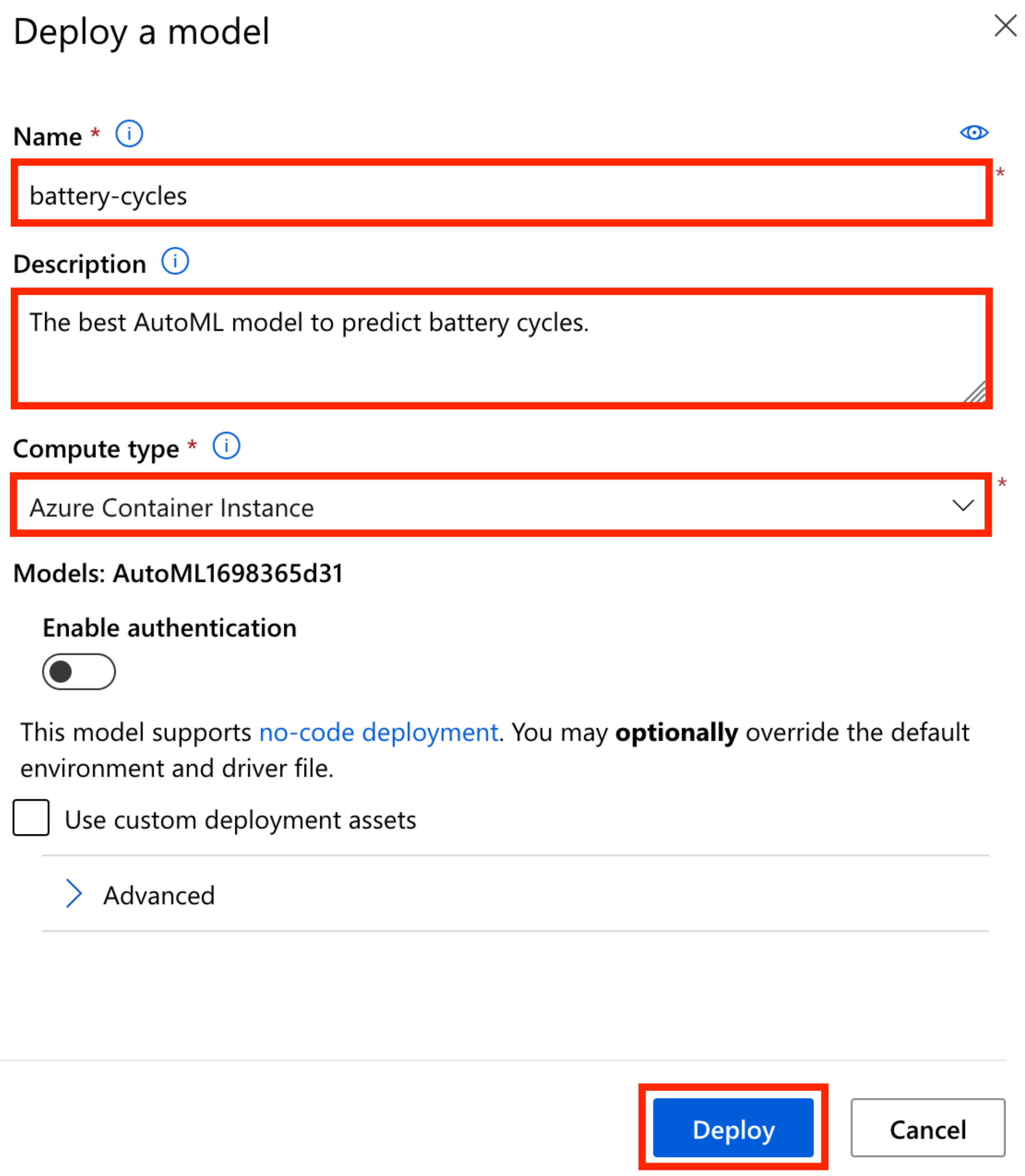
Details | Model | Explanations (preview) | Metrics | Outputs + logs | Images | Child runs | Snapshot

Select a metric to see a visualization or table of the data.

View as: ● Chart ○ Table

- □ explained_variance
- □ mean_absolute_error
- □ mean_absolute_percentage_error
- □ median_absolute_error
- □ normalized_mean_absolute_error
- □ normalized_median_absolute_error
- □ normalized_root_mean_squared_err...
- □ normalized_root_mean_squared_lo...
- ☑ predicted_true
- □ r2_score
- □ residuals
- □ root_mean_squared_error
- □ root_mean_squared_log_error
- □ spearman_correlation

2. Provide the `Name`, `Description` and `Compute type`, and then select **Deploy**:
    - **Name**: **battery-cycles**
    - **Description**: **The best AutoML model to predict battery cycles.**
    - **Compute type**: Select `ACI`.



Deploy a model

Name * ⓘ

battery-cycles

Description ⓘ

The best AutoML model to predict battery cycles.

Compute type * ⓘ

Azure Container Instance

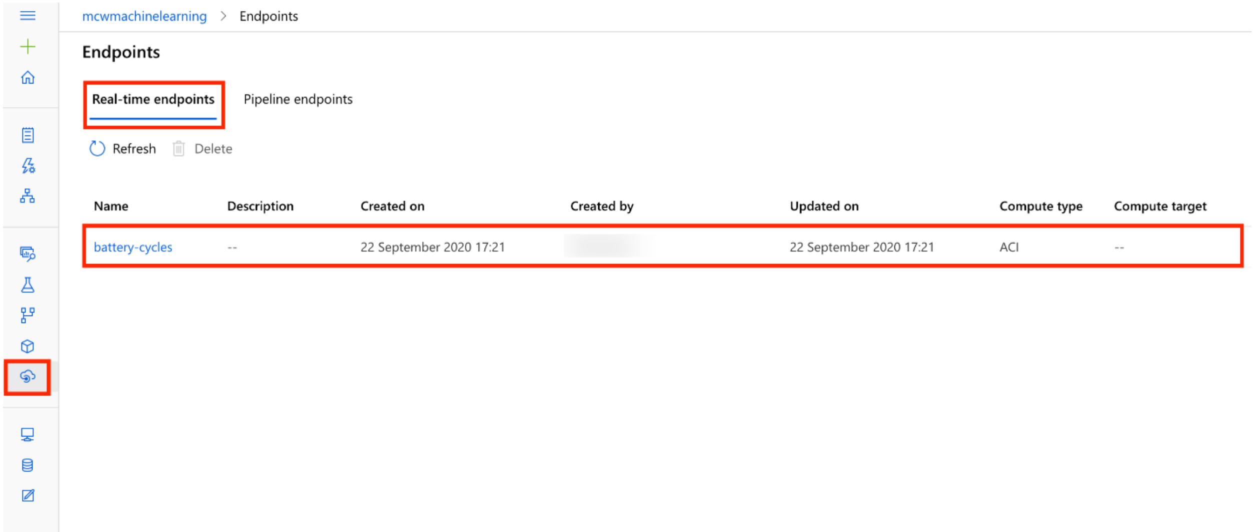
Models: AutoML1698365d31

Enable authentication

This model supports no-code deployment. You may **optionally** override the default environment and driver file.

□ Use custom deployment assets

3. The model deployment process will register the model, create the deployment image, and deploy it as a scoring webservice in an Azure Container Instance (ACI). To view the deployed model, from Azure Machine Learning studio select **Endpoints icon, Real-time endpoints**.



Name	Description	Created on	Created by	Updated on	Compute type	Compute target
battery-cycles	--	22 September 2020 17:21	[redacted]	22 September 2020 17:21	ACI	--

Note: The `battery-cycles` endpoint will show up in a matter of seconds, but the actual deployment takes several minutes. You can check the deployment state of the endpoint by selecting it and then selecting the `Details` tab. A successful deployment will have a state of `Healthy`.

4. If you see your model deployed in the above list, you are now ready to continue on to the next exercise.

Task 4: Perform batch scoring in Azure Databricks

1. Browse to your Azure Databricks Workspace and navigate to `AI with Databricks and AML \ 2.0 Batch Scoring for Timeseries`. This is the notebook you will step through executing in this exercise. Again, remember that you may need to reconnect to the **lab** cluster.
2. Follow the instructions within the notebook to complete the exercise.

Exercise 3: Creating a deep learning model (RNN) for time series data

Duration: 45 minutes

In this exercise, you will create a deep learning model (using a RNN - Recurrent Neural Network), and you will apply the forecast model to a Spark streaming job in order to make predictions against streaming data.

Task 1: Create the deep learning model and start a streaming job using a notebook

1. Browse to your Azure Databricks Workspace and navigate to `AI with Databricks and AML \ 3.0 Deep Learning with Time Series`. This is the notebook you will step through executing in this exercise.
2. Follow the instructions within the notebook to complete the exercise.

Exercise 4: Creating, training and tracking a deep learning text classification model with MLflow and Azure Machine Learning

Duration: 45 minutes

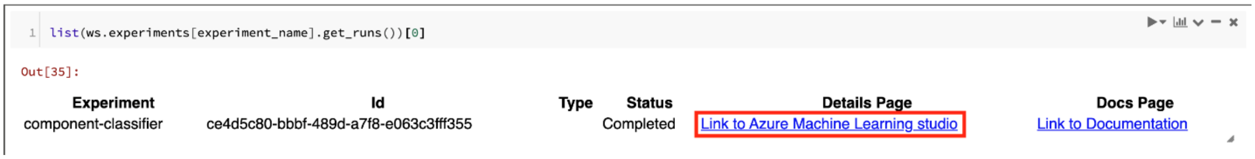
In this exercise, you create a model for classifying component text as compliant or non-compliant. You will train the model Azure Machine Learning and use MLflow integration with Azure Machine Learning to track and log experiment metrics and artifacts in the Azure Machine Learning workspace.

Task 1: Create, train and track the classification model using a notebook

1. Browse to your Azure Databricks Workspace and navigate to AIT with Databricks and AML \ 4.0 Deep Learning with Text. This is the notebook you will step through executing in this exercise.
2. Follow the instructions within the notebook to complete exercise.

Task 2: Review model performance metrics and training artifacts in Azure Machine Learning workspace

1. Select the **Link to Azure Machine Learning studio** from the output of the last cell in the notebook to open the `Run Details` page in the Azure Machine Learning studio.

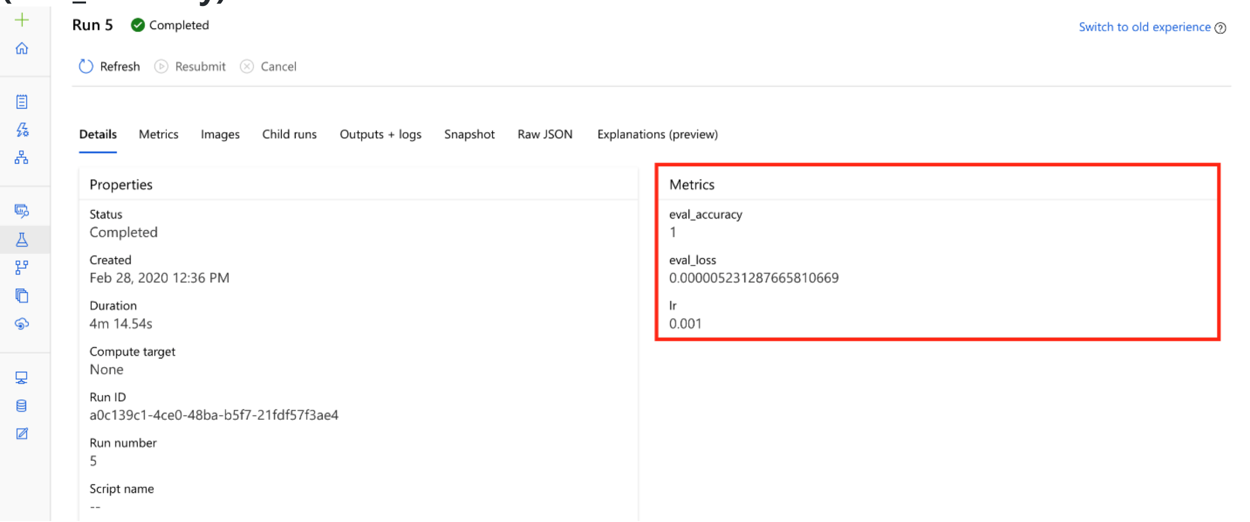


```
list(ws.experiments[experiment_name].get_runs())[0]
```

```
Out[35]:
     Experiment                         Id                  Type   Status           Details Page                    Docs Page
component-classifier  ce4d5c80-bbbf-489d-a7f8-e063c3fff355        Completed  Link to Azure Machine Learning studio  Link to Documentation
```

2. The **Run Details** page shows the three metrics that were logged via MLflow during the model training process: **Learning rate(lr)**, **evaluation loss(eval_loss)**, and **evaluation accuracy**

(eval_accuracy).



Properties

Status
Completed

Created
Feb 28, 2020 12:36 PM

Duration
4m 14.54s

Compute target
None

Run ID
a0c139c1-4ce0-48ba-b5f7-21fdf57f3ae4

Run number
5

Script name
--

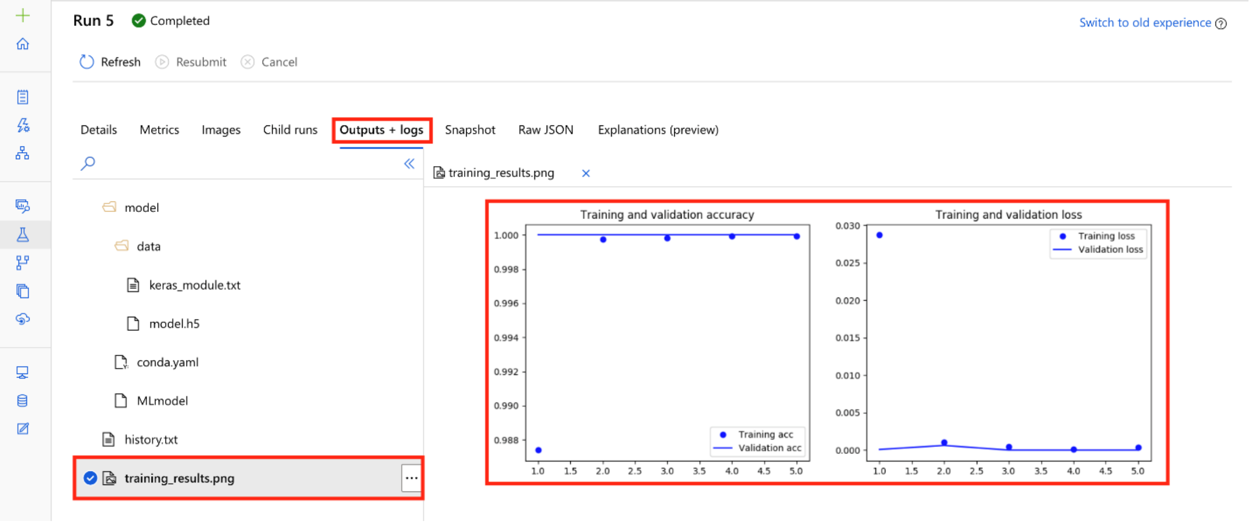
Metrics

eval_accuracy
1

eval_loss
0.000005231287665810669

lr
0.001

3. Next, select **Outputs + logs, tranining_reseults.png** to review the model training artifacts logged using MLflow. In this section, you can review the curves showing both accuracy and loss as the model training progress. You can also observe that MLflow logs the trained model and the training history with Azure Machine Learning
workspace.



Exercise 5: Evaluate model interpretability

Duration: 20 minutes

In this exercise, you will interpret the behavior of one of the models trained in previous exercises.

Task 1: Create the deep learning model and start a streaming job using a notebook

1. Browse to your Azure Databricks Workspace and navigate to `AI with Databricks and AML \ 5.0 Model Interpretability`. This is the notebook you will step through executing in this exercise.
2. Follow the instructions within the notebook to complete the exercise.

After the hands-on lab

Duration: 5 minutes

To avoid unexpected charges, it is recommended that you clean up all of your lab resources when you complete the lab.

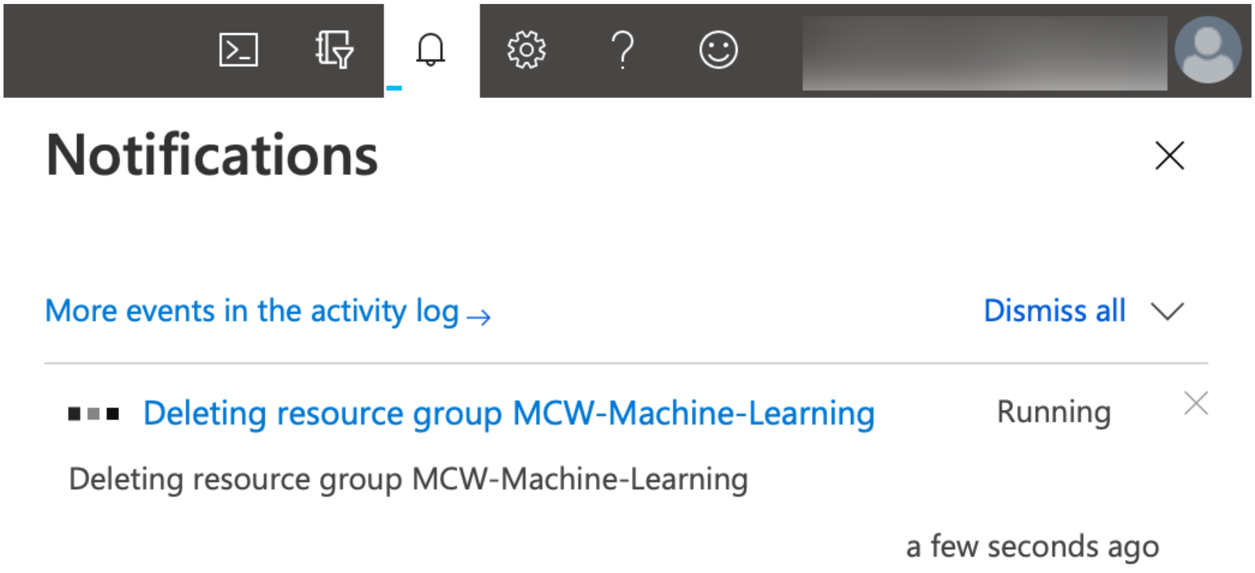
Task 1: Clean up lab resources

1. Navigate to the Azure Portal and locate the `MCW-Machine-Learning` Resources Group created for this lab.

2. Select **Delete resource group** from the command bar.

3. In the confirmation dialog that appears, enter the name of the resource group and select **Delete**.

4. Wait for the confirmation that the Resource Group has been successfully deleted. If you don't wait, and the delete fails for some reason, you may be left with resources running that were not expected. You can monitor using the Notifications dialog, which is accessible from the Alarm icon.

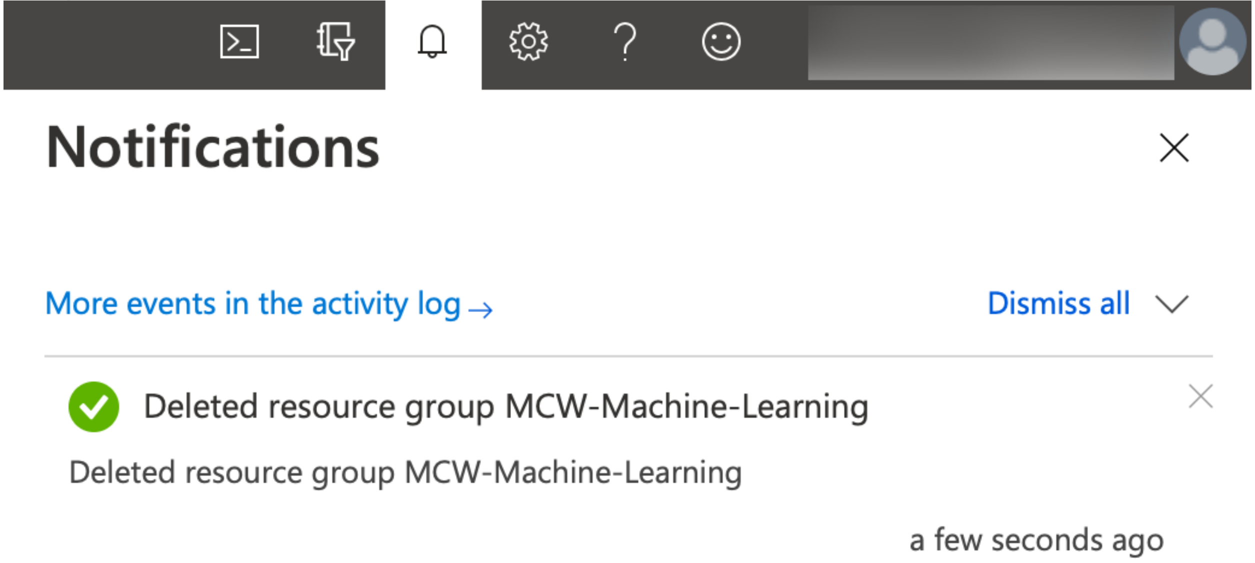


Deleting resource group MCW-Machine-Learning Running

Deleting resource group MCW-Machine-Learning

a few seconds ago

5. When the Notification indicates success, the cleanup is complete.



Notifications

More events in the activity log →

Dismiss all

Deleted resource group MCW-Machine-Learning

Deleted resource group MCW-Machine-Learning

a few seconds ago

You should follow all steps provided *after* attending the Hands-on lab.