Highlight Note

Using MLflow

MLflow is an Open Source library for managing machine learning experiments, and includes a tracking component for logging. If your organization already uses MLflow, you can continue to use it to track metrics in Azure Machine Learning.

Using MLflow Inline

You can use MLflow to run an inline experiment, as shown in this example:

```
from azureml.core import Experiment
import pandas as pd
import mlflow

# Set the MLflow tracking URI to the workspace
mlflow.set_tracking_uri(ws.get_mlflow_tracking_uri())

# Create an Azure ML experiment in your workspace
experiment = Experiment(workspace=ws, name='my-experiment')
mlflow.set_experiment(experiment.name)

# start the MLflow experiment
with mlflow.start_run():

    print("Starting experiment:", experiment.name)

    # load the data and count the rows
    data = pd.read_csv('data.csv')
    row_count = (len(data))

    # Log the row count
    mlflow.log_metric('observations', row_count)
```

Note that the code explicitly sets the MLflow tracking URI to the tracking endpoint provided by the Azure Machine Learning workspace.

Using MLflow in a Script

To use MLflow in a script-based experiment, th script must include code to start an MLflow run and log metrics; and the **ScriptRunConfig** for the experiment must include an evironment in which the **mlflow** and **azureml-mlflow** packages are installed.

Script

```
import pandas as pd
import mlflow

# start the MLflow experiment
with mlflow.start_run():
```

```
    print("Starting experiment:", experiment.name)

    # load the data and count the rows
    data = pd.read_csv('data.csv')
    row_count = (len(data))

    # Log the row count
    mlflow.log_metric('observations', row_count)
```

Code to initiate the experiment run

```
from azureml.core import Experiment, ScriptRunConfig, Environment
from azureml.core.conda_dependencies import CondaDependencies

# Create a Python environment for the experiment
mlflow_env = Environment("mlflow-env")

# Ensure the required packages are installed
packages = CondaDependencies.create(conda_packages=['pandas','pip'],
                                    pip_packages=['mlflow','azureml-mlflow'])
mlflow_env.python.conda_dependencies = packages

# Create a script config
script_config = ScriptRunConfig(source_directory='my_dir',
                                script='script.py',
                                environment=mlflow_env)

# submit the experiment
experiment = Experiment(workspace=ws, name='mlflow-script')
run = experiment.submit(config=script_config)
```

More Information: For more information about using MLflow with Azure Machine Learning, see **Track metrics and deploy models with MLflow and Azure Machine Learning** in the documentation.