**Track model training in Jupyter notebooks with MLflow**

Learn how to use MLflow for model tracking when experimenting in notebooks.

**Learning objectives**

In this module, you'll learn how to:

* Configure to use MLflow in notebooks
* Use MLflow for model tracking in notebooks

-- **Introducción**  
Imagina que eres un científico de datos en una empresa que desarrolla una aplicación para un laboratorio de investigación sobre el cáncer. La aplicación permitirá a los investigadores cargar una imagen de tejido para determinar si es saludable o no. Te han solicitado entrenar un modelo para detectar cáncer de mama usando una gran base de datos de imágenes que contiene ejemplos de tejido sano y enfermo.

Como estás familiarizado con Jupyter Notebooks, planeas usarlos para desarrollar el modelo. Además, quieres reentrenar el modelo periódicamente para mejorar su desempeño y luego desplegarlo para que los investigadores puedan usarlo en su aplicación.

Aprenderás a hacer un seguimiento del entrenamiento del modelo en notebooks utilizando MLflow en Azure Machine Learning.

-- **Configurar MLflow para el seguimiento de modelos en notebooks**

Como científico de datos, al desarrollar un modelo en un notebook, te interesa que los resultados sean reproducibles y que puedas revisar tu trabajo en cualquier momento. MLflow es una biblioteca de código abierto para gestionar y hacer seguimiento de tus experimentos de machine learning, permitiéndote registrar parámetros, métricas y artefactos del modelo.

Para usar MLflow en notebooks dentro del entorno de Azure Machine Learning, necesitas instalar las bibliotecas necesarias y configurar Azure Machine Learning como el almacenamiento de seguimiento.

**Configuración de MLflow en notebooks de Azure Machine Learning**

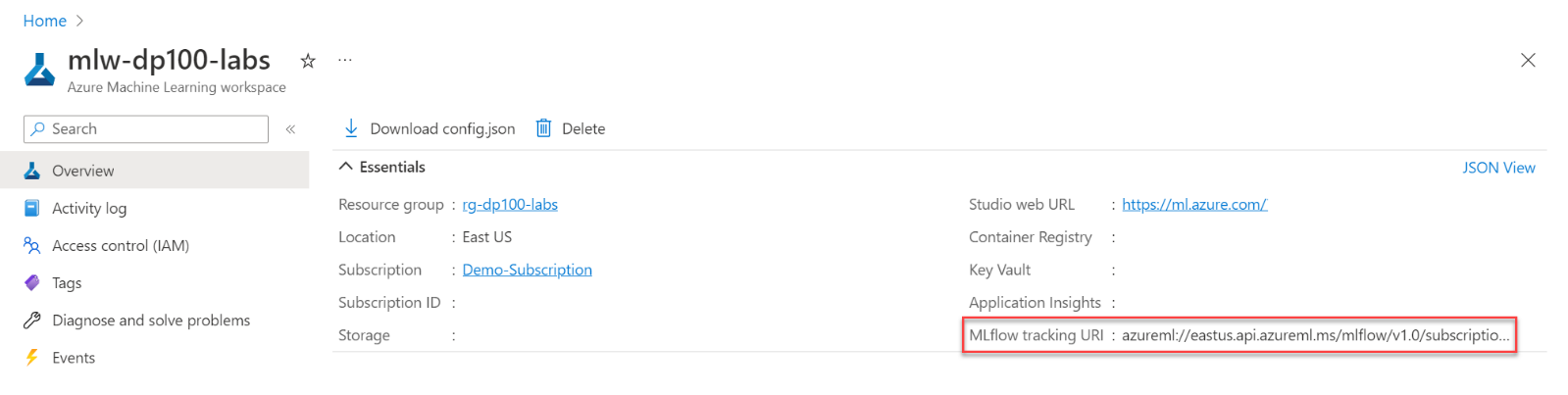
1. Dentro del workspace de Azure Machine Learning, puedes crear notebooks y conectarlos a una instancia de cómputo gestionada. Si usas notebooks en una instancia de cómputo de Azure, MLflow ya está configurado.
2. Para verificar que los paquetes necesarios están instalados, ejecuta:

**Uso de MLflow en un dispositivo local**

Si prefieres trabajar en un notebook en un dispositivo local:

1. Instala los paquetes mlflow y azureml-mlflow
2. Texto

   Descripción generada automáticamente Navega al portal de Azure Machine Learning, selecciona el workspace y copia el URI de seguimiento de MLflow.



1. Configura MLflow en tu notebook local para apuntar al workspace de Azure usando el URI de seguimiento:

mlflow.set\_tracking\_uri("MLFLOW-TRACKING-URI")

Con esta configuración, MLflow comenzará a registrar los resultados de tus experimentos en el workspace de Azure Machine Learning, permitiéndote explorar y evaluar los modelos de forma reproducible.

-- **Entrenar y hacer seguimiento de modelos en notebooks con MLflow**

Para agrupar los resultados de tus entrenamientos y registrar métricas de modelos al entrenarlos en un notebook, puedes usar la funcionalidad de registro de MLflow.

**Crear un experimento en MLflow**

Para organizar los entrenamientos, puedes crear un experimento. Si no creas uno, MLflow utilizará un experimento predeterminado llamado "Default".

Para crear un experimento específico, usa el siguiente comando en un notebook:

Texto

Descripción generada automáticamente con confianza media

**Registrar resultados con MLflow**

Al entrenar el modelo, puedes iniciar un "run" con start\_run() para que MLflow registre los datos del modelo. Hay dos métodos para hacer seguimiento de los resultados:

1. **Habilitar autologging**: MLflow permite el registro automático de métricas, parámetros, artefactos y modelos para algunas bibliotecas populares. Esto simplifica el seguimiento cuando el framework soporta esta funcionalidad.

Por ejemplo, para activar el autologging en modelos de XGBoost: mlflow.xgboost.autolog()

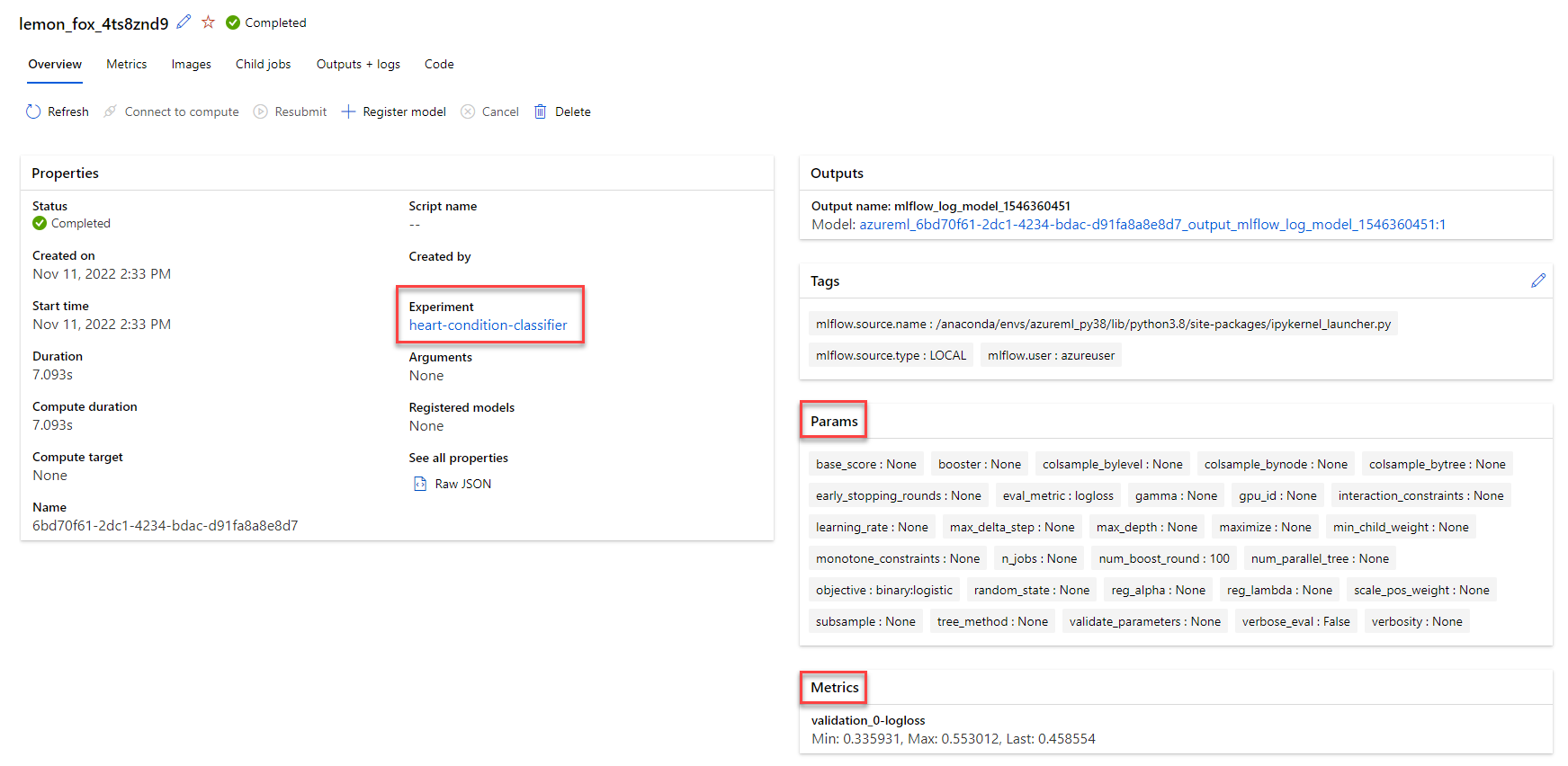
**Ejemplo de entrenamiento con autologging activado**:

Texto

Descripción generada automáticamente

Una vez que llamas a mlflow.xgboost.autolog(), MLflow registra automáticamente los detalles del entrenamiento en el workspace de Azure Machine Learning. Al finalizar el trabajo, puedes revisar todas las métricas y registros en el Azure Machine Learning Studio.

Este enfoque te permite monitorear y comparar los entrenamientos realizados, simplificando el proceso de evaluación y selección de los mejores modelos en tus experimentos.



**Uso de registro personalizado con MLflow**

Además de habilitar el **autologging**, también puedes registrar manualmente tu modelo con **MLflow**. Esto es útil para registrar información adicional o personalizada que no se captura automáticamente.

**Funciones comunes para el registro personalizado:**

1. **mlflow.log\_param()**: Registra un parámetro en formato clave-valor. Úsalo para cualquier parámetro de entrada que quieras registrar.

mlflow.log\_param("max\_depth", model.max\_depth)

1. **mlflow.log\_metric()**: Registra una métrica en formato clave-valor. El valor debe ser un número. Úsalo para almacenar cualquier salida con el run.

mlflow.log\_metric("accuracy", accuracy)

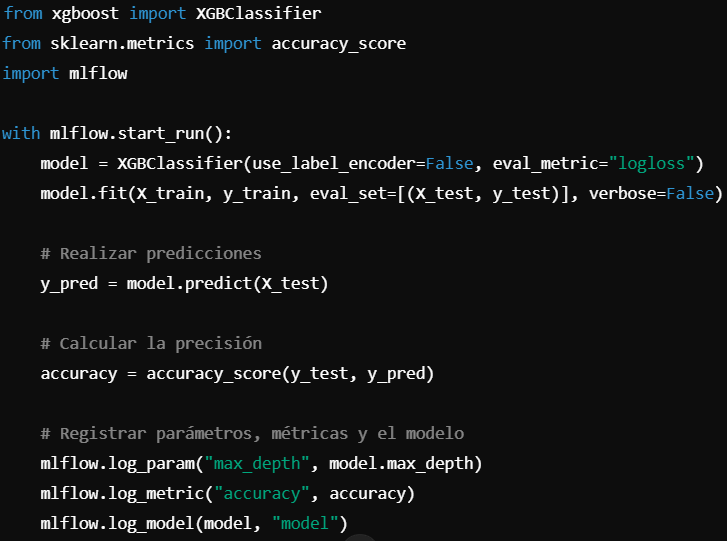
1. **mlflow.log\_artifact()**: Registra un archivo. Úsalo para cualquier gráfico que quieras registrar, asegurándote de guardar la imagen primero.

mlflow.log\_artifact("path/to/plot.png")

1. **mlflow.log\_model()**: Registra un modelo. Esta función permite crear un modelo de MLflow que puede incluir una firma personalizada, un entorno y ejemplos de entrada.

mlflow.log\_model(model, "model")

**Ejemplo de registro personalizado en un notebook**

Aquí tienes un ejemplo que muestra cómo usar el registro personalizado al entrenar un modelo con **XGBoost**:

**Beneficios del registro personalizado**

El registro personalizado te ofrece más flexibilidad, pero también implica más trabajo, ya que debes definir explícitamente cualquier parámetro, métrica o artefacto que desees registrar. Una vez completado el trabajo, puedes revisar todas las métricas registradas en el **Azure Machine Learning Studio**.

Al utilizar tanto el **autologging** como el registro personalizado, puedes aprovechar lo mejor de ambos mundos, asegurando que capturas toda la información relevante sobre tu modelo.

