Learn how to log and register an MLflow model in Azure Machine Learning.

**Learning objectives**

In this module, you'll learn how to:

* Log models with MLflow.
* Understand the MLmodel format.
* Register an MLflow model in Azure Machine Learning.

--- **Despliegue de Modelos con MLflow en Azure Machine Learning**

Después de entrenar un modelo de machine learning, el siguiente paso es implementarlo para integrarlo en una aplicación. En Azure Machine Learning, puedes desplegar fácilmente un modelo a un endpoint en lote o en línea al registrarlo con MLflow.

Imagina que eres un científico de datos que trabaja para una empresa que desarrolla una aplicación para profesionales de la salud, diseñada para ayudar en el diagnóstico de diabetes en pacientes. Los profesionales ingresan la información médica del paciente en la aplicación y reciben una respuesta indicando si el paciente tiene probabilidades de padecer diabetes.

**Proceso de Reentrenamiento y Actualización del Modelo**

En este caso, es fundamental tener un modelo actualizado y preciso. Es probable que necesites reentrenar el modelo periódicamente a medida que haya más datos disponibles. Esto permite mejorar la precisión del modelo con el tiempo. Cada vez que el modelo es reentrenado, querrás actualizar el modelo que está desplegado en el endpoint, para que la aplicación esté siempre utilizando la versión más reciente.

**Registro del Modelo con MLflow para Despliegue**

MLflow es una herramienta de administración de modelos integrada en Azure Machine Learning que facilita el registro de modelos para su despliegue. El registro de un modelo con MLflow en Azure es un paso esencial para preparar el modelo para su implementación en producción.

--- **Registrar Modelos con MLflow en Azure Machine Learning**

MLflow es una plataforma de código abierto que simplifica el despliegue de modelos de machine learning, independientemente del framework utilizado. La integración de MLflow con Azure Machine Learning facilita la implementación de modelos entrenados y rastreados con MLflow, permitiendo opciones como el despliegue sin código en Azure.

**Ventajas de Usar MLflow**

Cuando entrenas un modelo en Azure Machine Learning, puedes usar MLflow para registrar el modelo, lo que estandariza el empaquetado y facilita la transferencia del modelo entre distintos entornos, como desarrollo y producción. Al registrar un modelo, se crea un archivo MLmodel que contiene los metadatos del modelo, permitiendo su rastreo y uso en diferentes pipelines o despliegues.

**Formas de Registrar Modelos con MLflow**

**1. Autologging**

MLflow permite la opción de autologging mediante mlflow.autolog(), que registra automáticamente parámetros, métricas, artefactos y el modelo al llamar el método .fit(). Puedes especificar el tipo de modelo (flavor) que estás entrenando, como Scikit-learn, Keras, TensorFlow, etc., utilizando mlflow.<flavor>.autolog(). Esto generará una carpeta de salida que contiene todos los artefactos y el archivo MLmodel.

Ejemplo de autologging con Scikit-learn:

import mlflow.sklearn

mlflow.sklearn.autolog()

**2. Registro Manual de un Modelo**

Si prefieres tener un control más detallado sobre el registro del modelo, puedes desactivar el autologging del modelo y añadirlo manualmente. También puedes personalizar la firma del modelo (signature), que define el esquema de entrada y salida y se guarda en formato JSON en el archivo MLmodel. La firma asegura que los datos de entrada y salida coincidan con lo esperado al desplegar el modelo.

Texto

Descripción generada automáticamenteEjemplo de registro manual con firma:

**3. Definir la Firma Manualmente**

Texto

Descripción generada automáticamenteTambién puedes definir la firma manualmente utilizando ModelSignature, que especifica el esquema de los datos de entrada y salida:

--- **Comprendiendo el Formato de Modelo MLflow en Azure Machine Learning**

MLflow utiliza un formato estándar de modelo conocido como **MLModel**, que almacena todos los activos del modelo en una carpeta o directorio. Uno de los archivos esenciales dentro de esta estructura es el archivo MLmodel, que actúa como la fuente de verdad sobre cómo debe cargarse y usarse el modelo. Este archivo contiene metadatos que definen aspectos críticos del modelo, incluyendo su "flavor" y su "signature".

**Estructura del Archivo MLmodel**

El archivo MLmodel puede incluir los siguientes elementos clave:

* **artifact\_path**: La ruta donde se registra el modelo durante el entrenamiento.
* **flavor**: La biblioteca de machine learning con la que se creó el modelo.
* **model\_uuid**: Identificador único del modelo registrado.
* **run\_id**: Identificador único de la ejecución en la que se creó el modelo.
* **signature**: Especifica el esquema de entradas y salidas del modelo.
  + **inputs**: Datos válidos de entrada para el modelo.
  + **outputs**: Resultados válidos que el modelo puede producir.

Texto

Descripción generada automáticamente**Ejemplo de archivo MLmodel:**

**Configuración del Flavor**

**El flavor indica la biblioteca o framework de machine learning con el que fue creado el modelo, como sklearn, fastai, tensorflow, entre otros. Esto permite que cada flavor emplee sus propios métodos de serialización, optimizando rendimiento y compatibilidad sin comprometer la interoperabilidad en el estándar MLModel. Además, python\_function es el flavor predeterminado para modelos en Python, lo que permite cargar cualquier modelo Python independientemente del framework utilizado.**

**Ejemplo de flavor en Python:**

Texto

Descripción generada automáticamente

**Configuración de la Signature**

La **signature** define el esquema de entradas y salidas del modelo y actúa como un contrato de datos entre el modelo y el servidor. Esta configuración garantiza que los datos de entrada y salida coincidan con las especificaciones esperadas.

Hay dos tipos principales de firmas:

1. **Firma basada en columnas**: Para datos tabulares, se usa pandas.DataFrame como entrada.
2. **Firma basada en tensores**: Para datos no estructurados como imágenes o texto, se usa numpy.ndarray.

La signature se crea al registrar el modelo. Si utilizas autologging en MLflow, la signature se infiere automáticamente. Sin embargo, si necesitas un esquema diferente, puedes personalizar la signature al registrar manualmente el modelo.

**Importancia de la Signature en el Despliegue**

Cuando despliegas un modelo MLflow en Azure Machine Learning, los esquemas de entrada y salida definidos en la signature se aplicarán estrictamente. Esto significa que, para que los datos enviados al modelo coincidan con los tipos y dimensiones esperados, deben cumplir con el esquema especificado.

--- **Registrar un Modelo de MLflow en Azure Machine Learning**

En Azure Machine Learning, los modelos se entrenan dentro de trabajos, y sus artefactos se almacenan en las salidas del trabajo. Para facilitar la gestión de estos modelos, puedes registrarlos en el **Azure Machine Learning Model Registry**, donde cada modelo se guarda y versiona en el workspace. Registrar modelos facilita la organización y el seguimiento, permitiendo agregar etiquetas de metadatos para realizar búsquedas rápidas.

Existen tres tipos de modelos que puedes registrar:

1. **MLflow**: Para modelos entrenados y registrados con MLflow, recomendado para casos de uso estándar.
2. **Custom**: Modelos personalizados no compatibles directamente con Azure Machine Learning.
3. **Triton**: Modelos para cargas de trabajo de deep learning, comúnmente utilizados para despliegues de TensorFlow y PyTorch.

Azure Machine Learning ofrece varias maneras de registrar un modelo de MLflow: mediante el Studio, la CLI de Azure o el SDK de Python. La integración de Azure Machine Learning con MLflow permite que durante el despliegue se generen automáticamente el entorno y el script de scoring, lo que facilita la gestión de modelos y su implementación.

**Paso 1: Entrenar el Modelo**

Para entrenar el modelo, se puede enviar un script de entrenamiento como un trabajo de comando en Azure Machine Learning. El siguiente código muestra cómo hacerlo:

Texto

Descripción generada automáticamente**Paso 2: Registrar el Modelo en Azure Machine Learning**

Texto

Descripción generada automáticamenteUna vez completado el trabajo de entrenamiento, puedes usar el nombre del trabajo (job\_name) para encontrar la ejecución y registrar el modelo desde las salidas del trabajo:

Al registrar el modelo, se guarda en el Model Registry de Azure Machine Learning, disponible en la página de **Models** en Azure Machine Learning Studio. Aquí podrás acceder al archivo MLmodel y otros artefactos necesarios para el despliegue y uso del modelo.