







**Validate the model before deployment**

Run the following code to validate model inference works on the example input data and logged model dependencies, prior to deploying it to a serving endpoint

import mlflow

model\_uri = 'runs:/6bfc465a59c34b41a317bd9c2d0f6137/model'

# Replace INPUT\_EXAMPLE with your own input example to the model

# A valid input example is a data instance suitable for pyfunc prediction

input\_data = INPUT\_EXAMPLE

# Verify the model with the provided input data using the logged dependencies.

# For more details, refer to:

# https://mlflow.org/docs/latest/models.html#validate-models-before-deployment

mlflow.models.predict(

model\_uri=model\_uri,

input\_data=input\_data,

env\_manager="uv",

)

**Make Predictions**

Predict on a Pandas DataFrame:

import mlflow

logged\_model = 'runs:/6bfc465a59c34b41a317bd9c2d0f6137/model'

# Load model as a PyFuncModel.

loaded\_model = mlflow.pyfunc.load\_model(logged\_model)

# Predict on a Pandas DataFrame.

import pandas as pd

import mlflow

from pyspark.sql.functions import struct, col

logged\_model = 'runs:/6bfc465a59c34b41a317bd9c2d0f6137/model'

# Load model as a Spark UDF. Override result\_type if the model does not return double values.

loaded\_model = mlflow.pyfunc.spark\_udf(spark, model\_uri=logged\_model)

# Predict on a Spark DataFrame.

df.withColumn('predictions', loaded\_model(struct(\*map(col, df.columns))))

loaded\_model.predict(pd.DataFrame(data))

**Predict on a Spark DataFrame:**

import mlflow

from pyspark.sql.functions import struct, col

logged\_model = 'runs:/6bfc465a59c34b41a317bd9c2d0f6137/model'

# Load model as a Spark UDF. Override result\_type if the model does not return double values.

loaded\_model = mlflow.pyfunc.spark\_udf(spark, model\_uri=logged\_model)

# Predict on a Spark DataFrame.

df.withColumn('predictions', loaded\_model(struct(\*map(col, df.columns))))