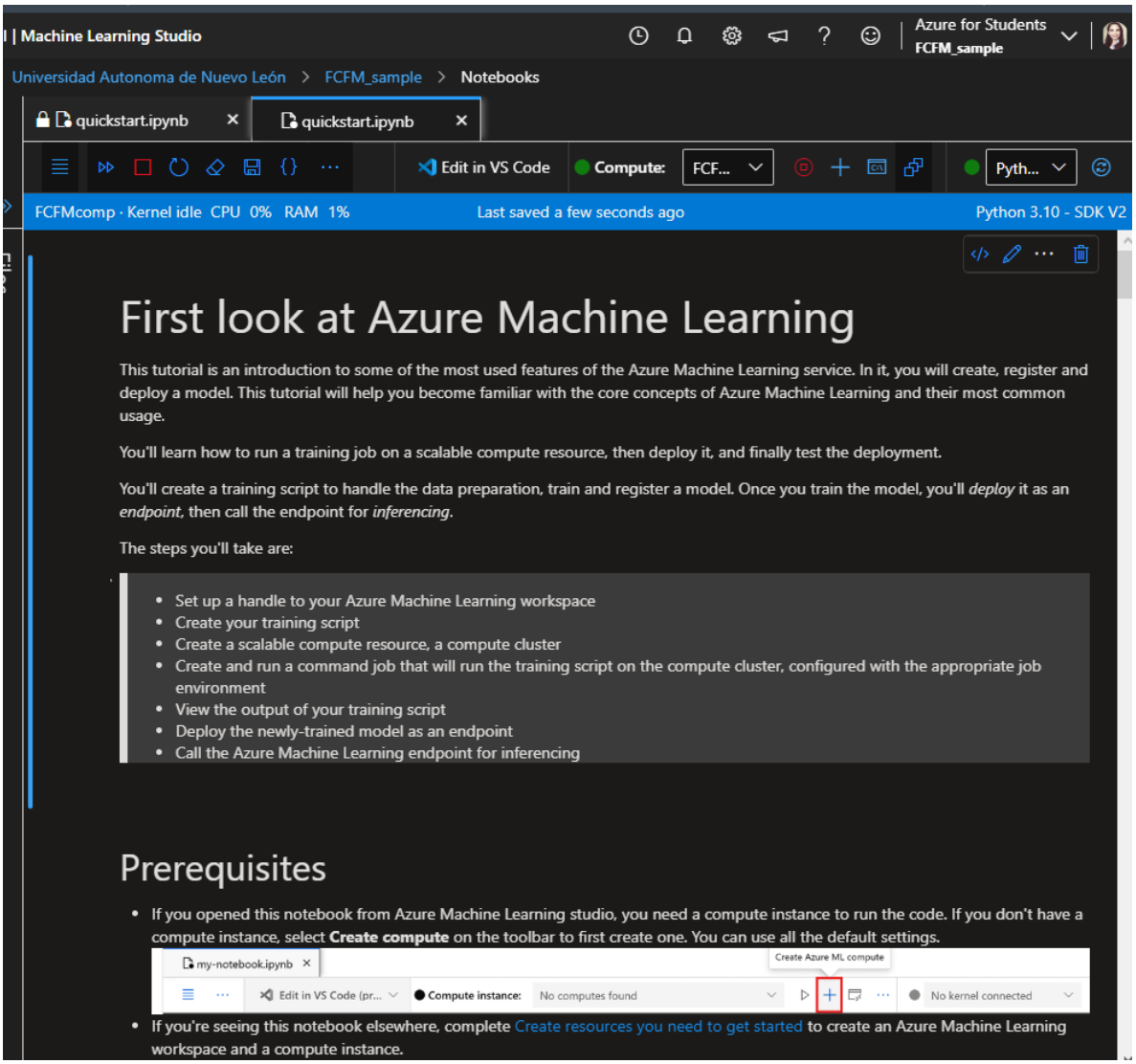
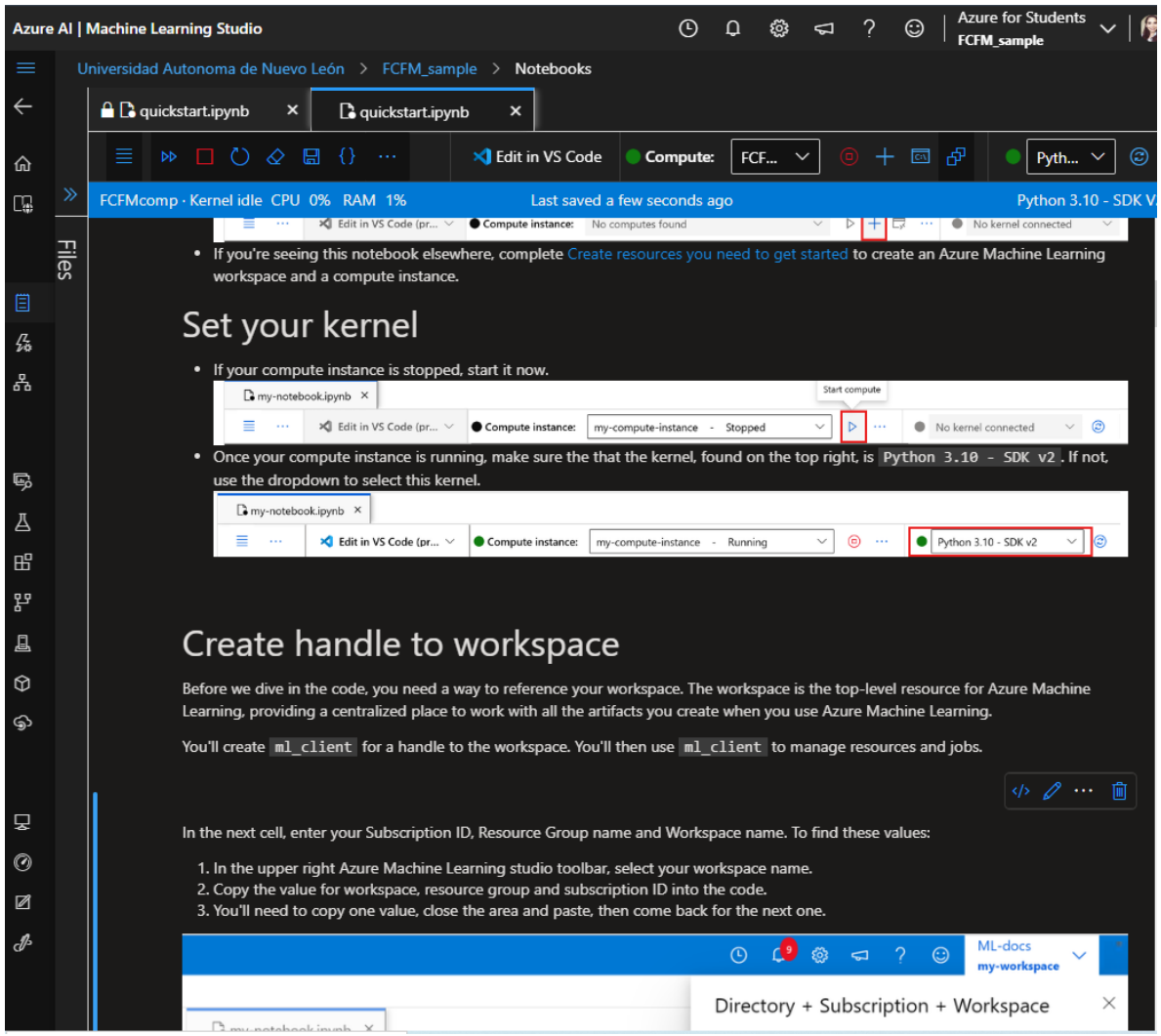
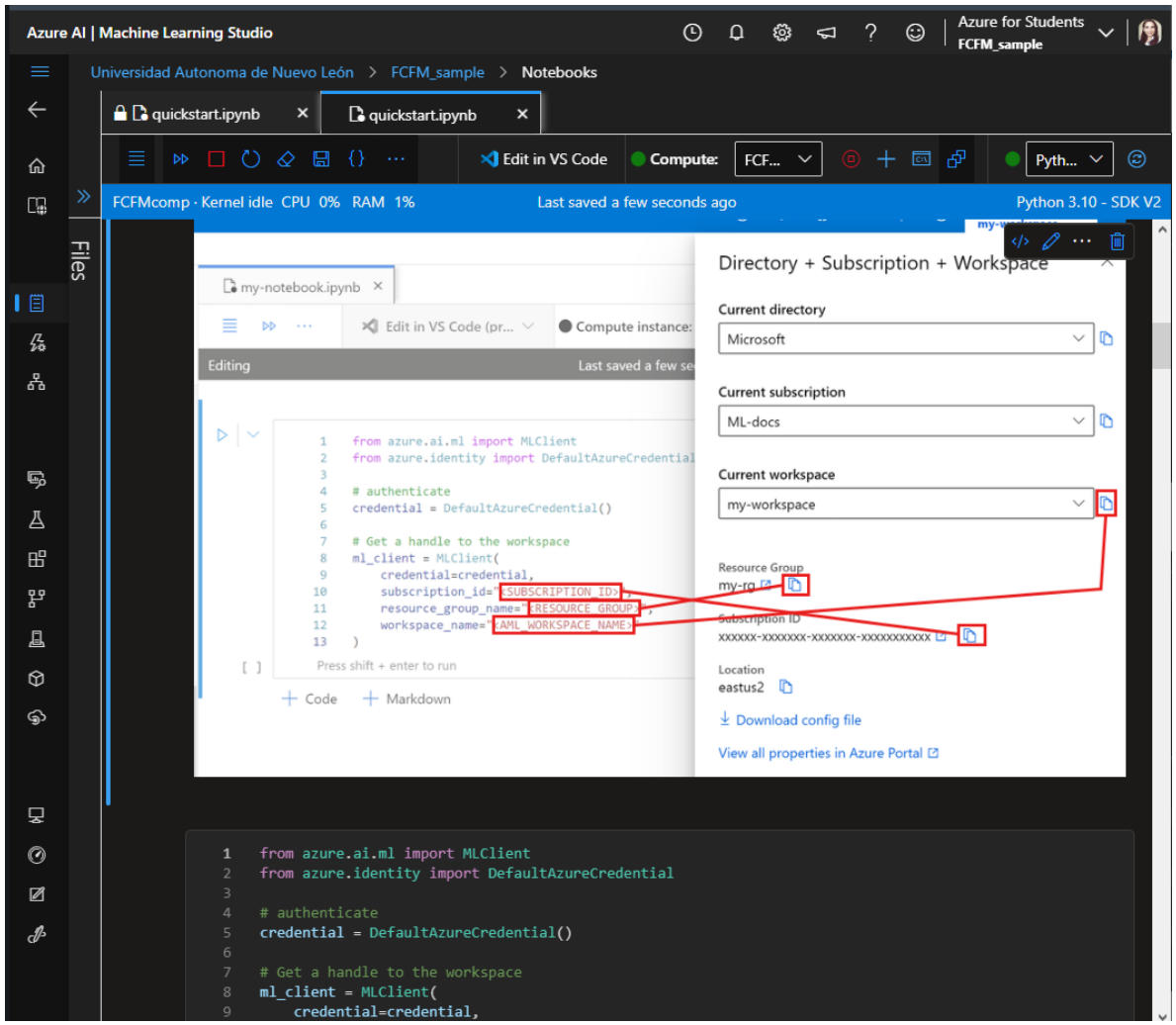
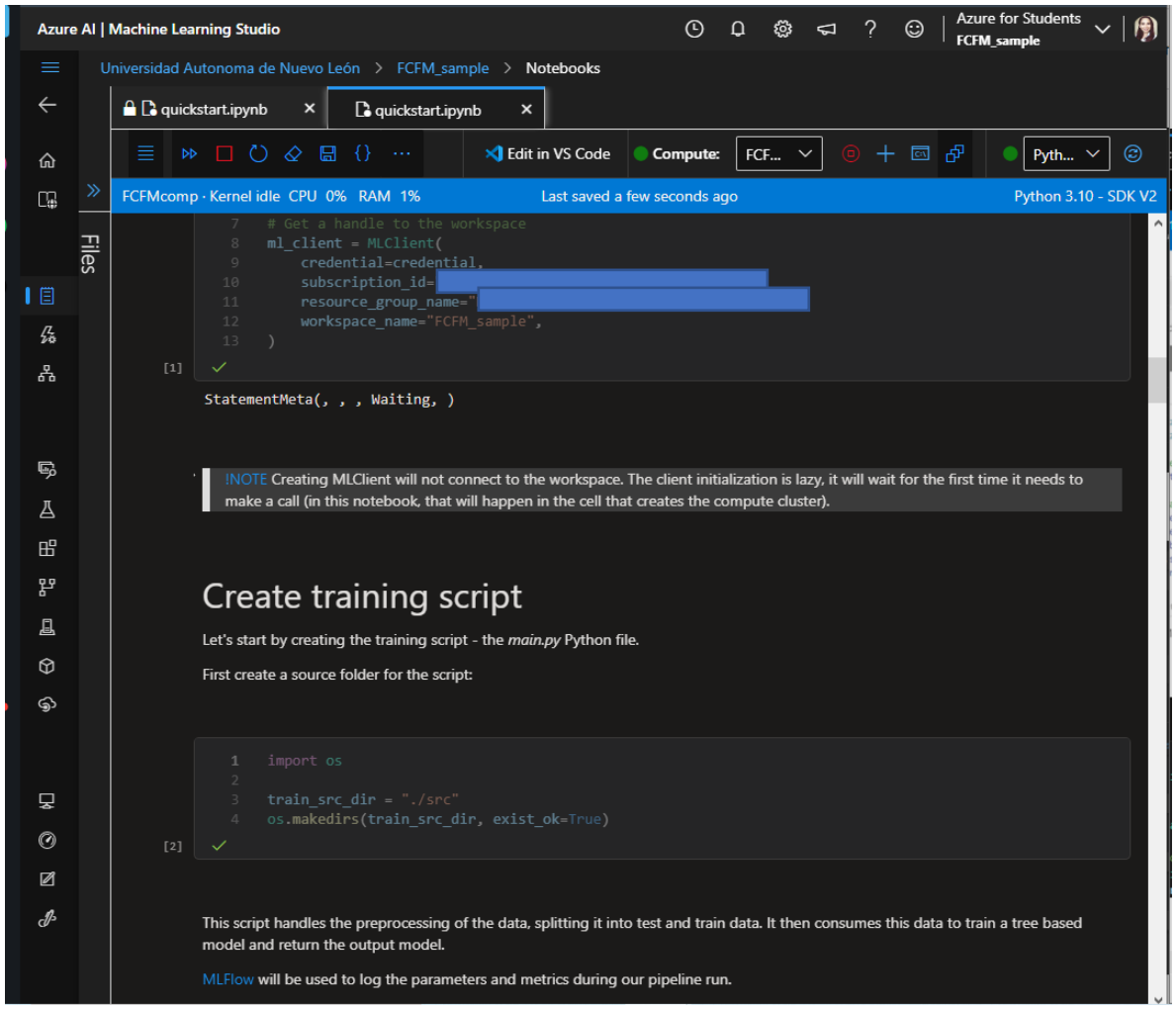
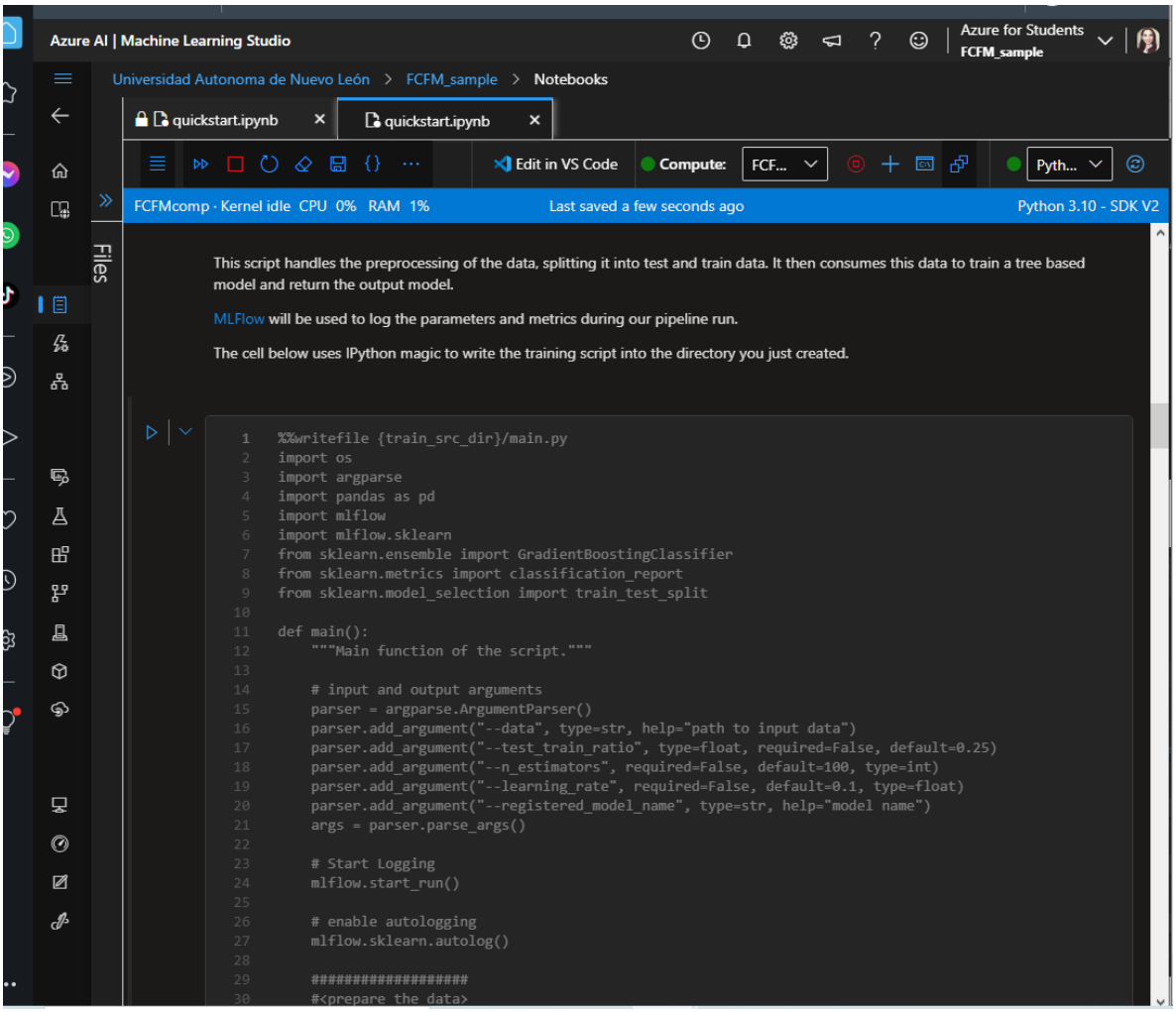
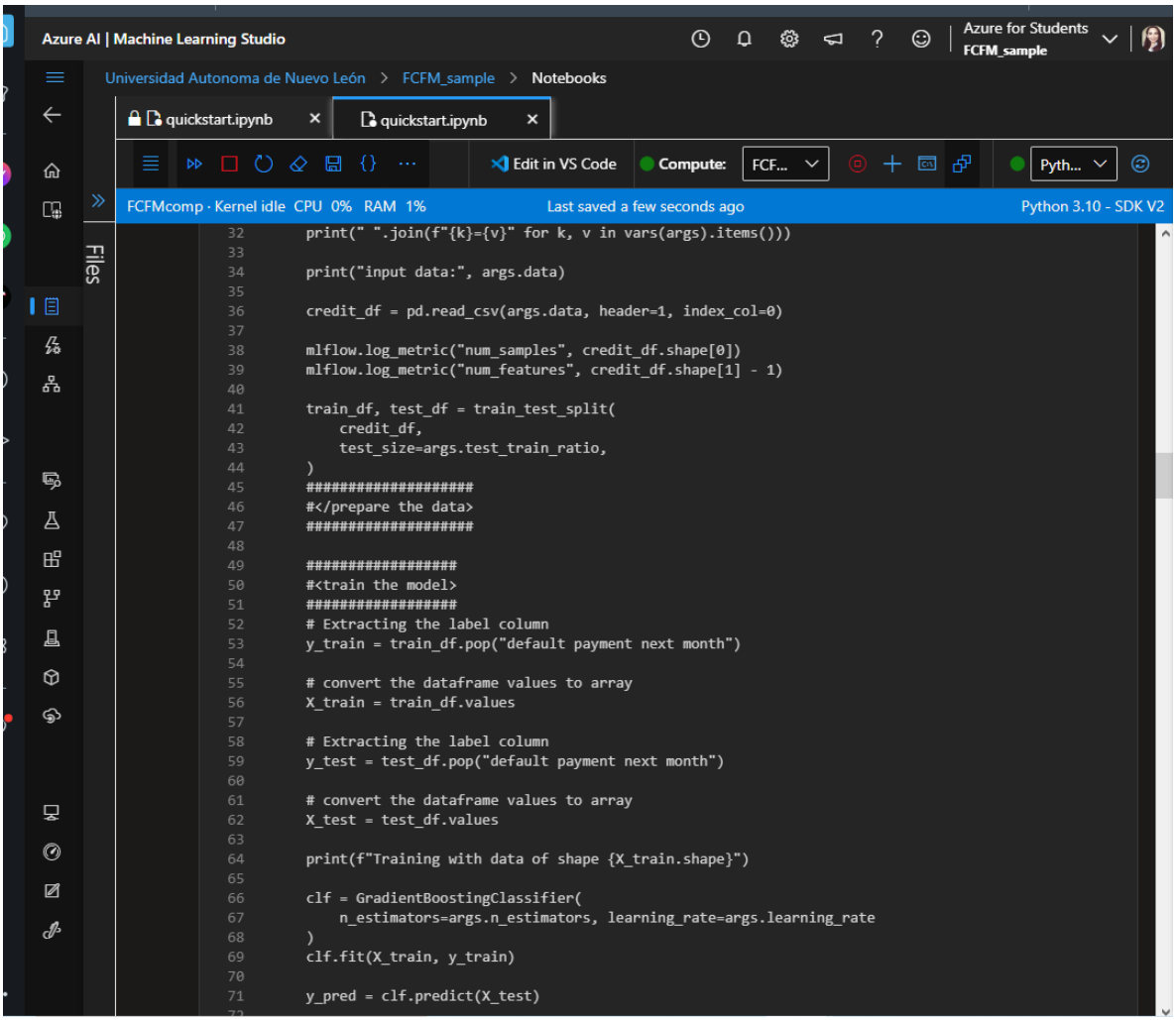

| Machine Learning Studio
Azure for Students
FCFM_sample
Universidad Autonoma de Nuevo León > FCFM_sample > Notebooks
quickstart.ipynb
quickstart.ipynb
Edit in VS Code
Compute:
FCF...
Pyth...
FCFMcomp · Kernel idle CPU 0% RAM 1%
Last saved a few seconds ago
Python 3.10 - SDK V2
First look at Azure Machine Learning
This tutorial is an introduction to some of the most used features of the Azure Machine Learning service. In it, you will create, register and deploy a model. This tutorial will help you become familiar with the core concepts of Azure Machine Learning and their most common usage.
You'll learn how to run a training job on a scalable compute resource, then deploy it, and finally test the deployment.
You'll create a training script to handle the data preparation, train and register a model. Once you train the model, you'll deploy it as an endpoint, then call the endpoint for inferencing.
The steps you'll take are:
Set up a handle to your Azure Machine Learning workspace
Create your training script
Create a scalable compute resource, a compute cluster
Create and run a command job that will run the training script on the compute cluster, configured with the appropriate job environment
View the output of your training script
Deploy the newly-trained model as an endpoint
Call the Azure Machine Learning endpoint for inferencing
Prerequisites
If you opened this notebook from Azure Machine Learning studio, you need a compute instance to run the code. If you don't have a compute instance, select Create compute on the toolbar to first create one. You can use all the default settings.
my-notebook.ipynb
Create Azure ML compute
Edit in VS Code (pr...
Compute instance:
No computes found
No kernel connected
If you're seeing this notebook elsewhere, complete Create resources you need to get started to create an Azure Machine Learning workspace and a compute instance.

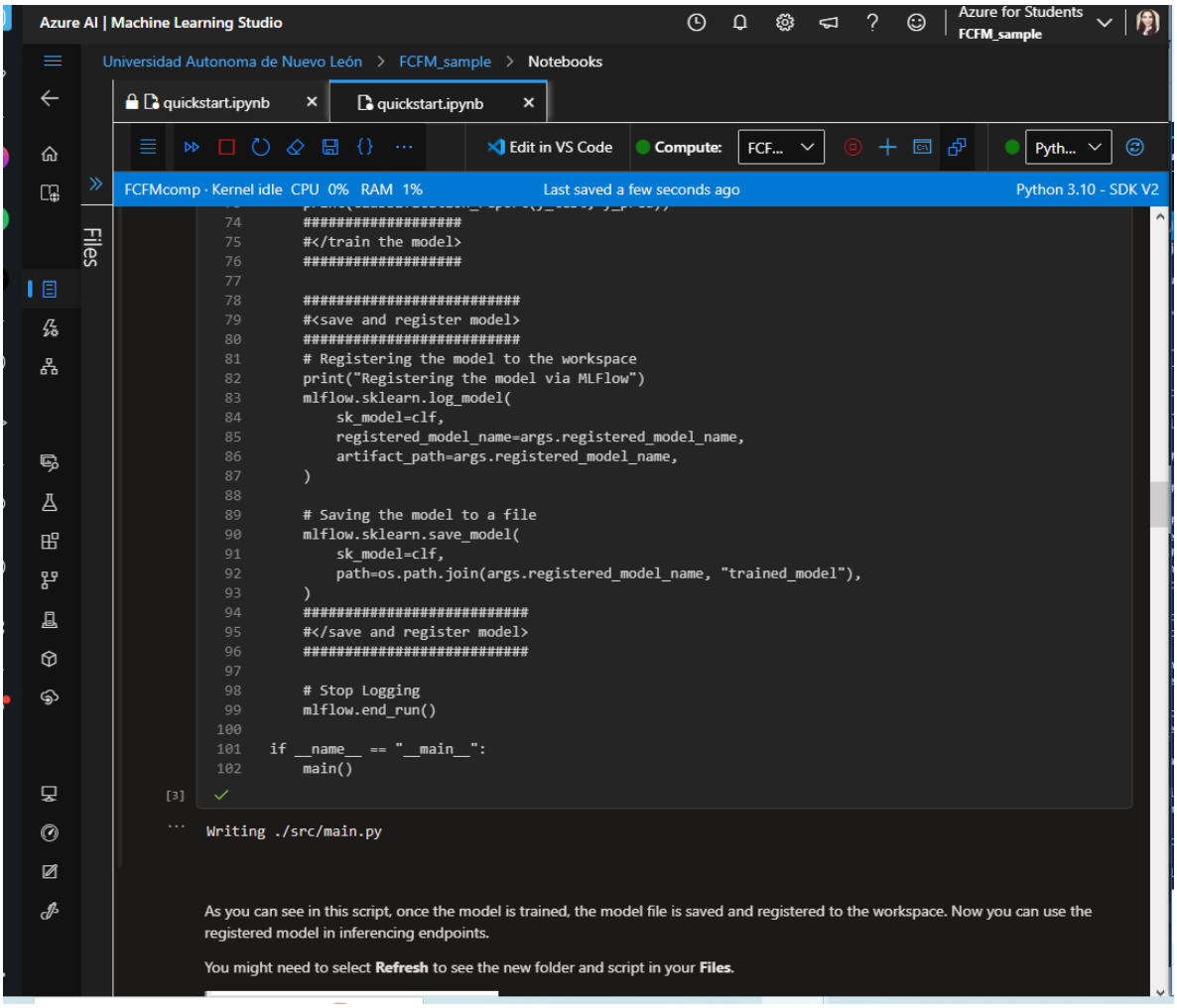

Azure AI | Machine Learning Studio
Azure for Students
FCFM_sample
Universidad Autonoma de Nuevo León > FCFM_sample > Notebooks
quickstart.ipynb
quickstart.ipynb
Edit in VS Code
Compute:
FCF...
Pyth...
FCFMcomp · Kernel idle CPU 0% RAM 1%
Last saved a few seconds ago
Python 3.10 - SDK V
Files
Edit in VS Code (pr...
Compute instance:
No computes found
No kernel connected
If you're seeing this notebook elsewhere, complete Create resources you need to get started to create an Azure Machine Learning workspace and a compute instance.
Set your kernel
If your compute instance is stopped, start it now.
my-notebook.ipynb
Start compute
Edit in VS Code (pr...
Compute instance:
my-compute-instance - Stopped
No kernel connected
Once your compute instance is running, make sure the that the kernel, found on the top right, is Python 3.10 - SDK v2. If not, use the dropdown to select this kernel.
my-notebook.ipynb
Edit in VS Code (pr...
Compute instance:
my-compute-instance - Running
Python 3.10 - SDK v2
Create handle to workspace
Before we dive in the code, you need a way to reference your workspace. The workspace is the top-level resource for Azure Machine Learning, providing a centralized place to work with all the artifacts you create when you use Azure Machine Learning.
You'll create ml_client for a handle to the workspace. You'll then use ml_client to manage resources and jobs.
In the next cell, enter your Subscription ID, Resource Group name and Workspace name. To find these values:
1. In the upper right Azure Machine Learning studio toolbar, select your workspace name.
2. Copy the value for workspace, resource group and subscription ID into the code.
3. You'll need to copy one value, close the area and paste, then come back for the next one.
ML-docs
my-workspace
Directory + Subscription + Workspace

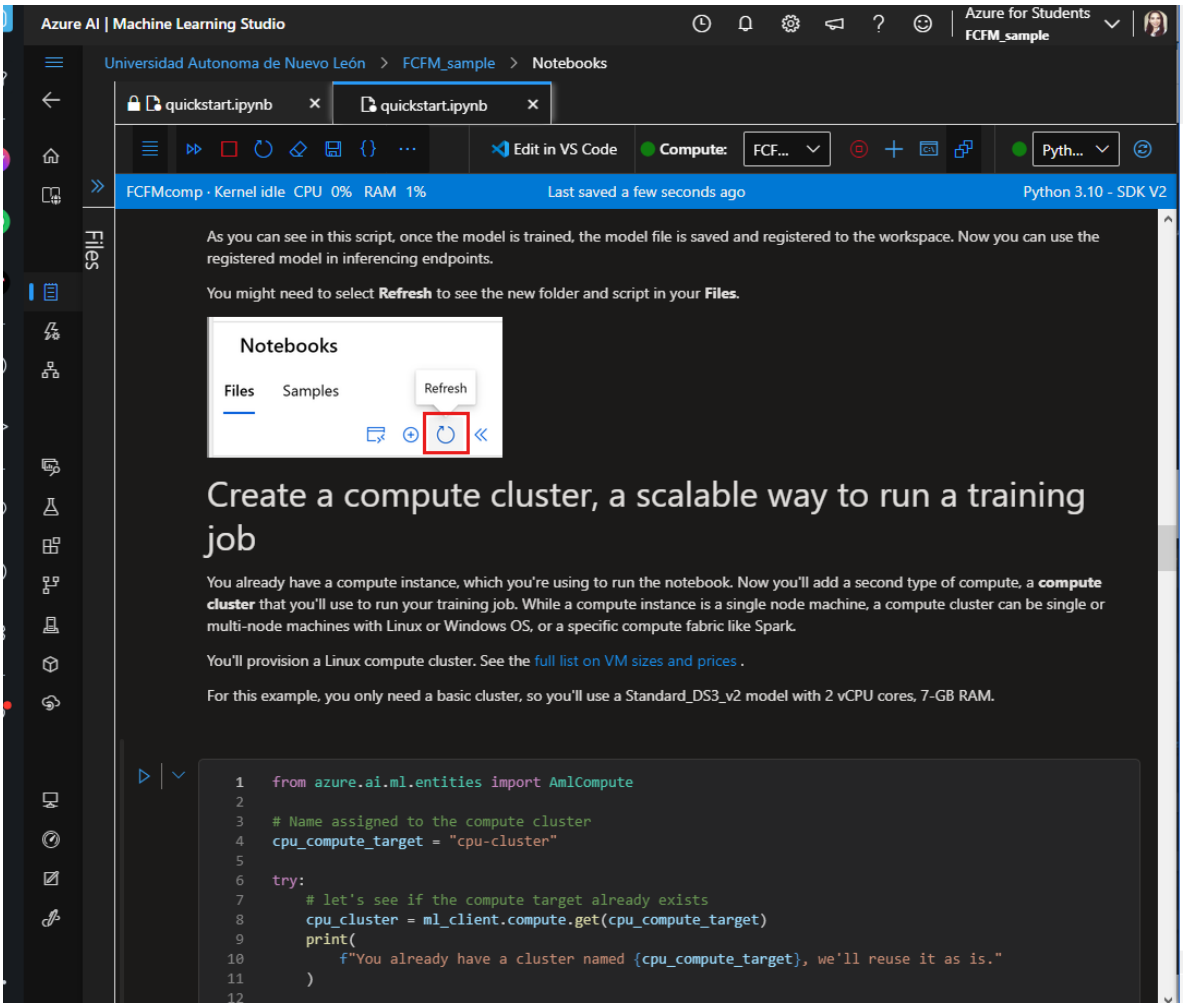

Azure AI | Machine Learning Studio
Azure for Students
FCFM_sample
Universidad Autonoma de Nuevo León > FCFM_sample > Notebooks
quickstart.ipynb
quickstart.ipynb
Edit in VS Code
Compute:
FCF...
Pyth...
FCFMcomp · Kernel idle CPU 0% RAM 1%
Last saved a few seconds ago
Python 3.10 - SDK V2
Files
my-notebook.ipynb
Edit in VS Code (pr...
Compute instance:
Editing
Last saved a few se
from azure.ai.ml import MLClient
from azure.identity import DefaultAzureCredential
# authenticate
credential = DefaultAzureCredential()
# Get a handle to the workspace
ml_client = MLClient(
credential=credential,
subscription_id="<SUBSCRIPTION_ID>",
resource_group_name="<RESOURCE_GROUP>",
workspace_name="<AML_WORKSPACE_NAME>"
)
Press shift + enter to run
Code
Markdown
Directory + Subscription + Workspace
Current directory
Microsoft
Current subscription
ML-docs
Current workspace
my-workspace
Resource Group
my-rg
Subscription ID
xxxxxx-xxxxxxx-xxxxxxx-xxxxxxxxxx
Location
eastus2
Download config file
View all properties in Azure Portal
from azure.ai.ml import MLClient
from azure.identity import DefaultAzureCredential
# authenticate
credential = DefaultAzureCredential()
# Get a handle to the workspace
ml_client = MLClient(
credential=credential,

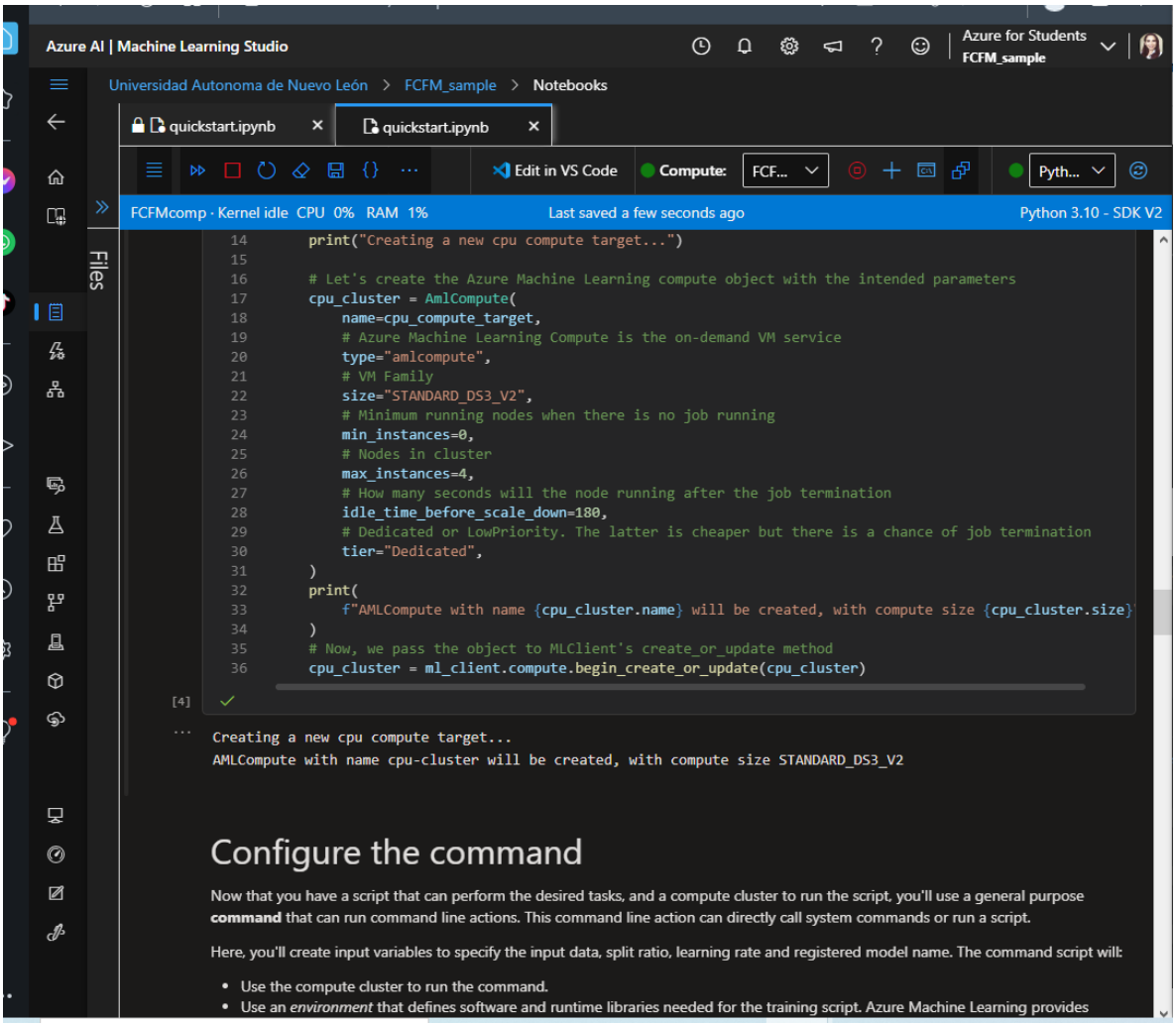

Azure AI | Machine Learning Studio
Azure for Students
FCFM_sample
Universidad Autonoma de Nuevo León > FCFM_sample > Notebooks
quickstart.ipynb
quickstart.ipynb
Edit in VS Code
Compute:
FCF...
Pyth...
FCFMcomp · Kernel idle CPU 0% RAM 1%
Last saved a few seconds ago
Python 3.10 - SDK V2
Files
# Get a handle to the workspace
ml_client = MLClient(
credential=credential,
subscription_id=
resource_group_name="
workspace_name="FCFM_sample",
)
[1]
StatementMeta(, , , Waiting,)
!NOTE Creating MLClient will not connect to the workspace. The client initialization is lazy, it will wait for the first time it needs to make a call (in this notebook, that will happen in the cell that creates the compute cluster).
Create training script
Let's start by creating the training script - the main.py Python file.
First create a source folder for the script:
import os
train_src_dir = "./src"
os.makedirs(train_src_dir, exist_ok=True)
[2]
This script handles the preprocessing of the data, splitting it into test and train data. It then consumes this data to train a tree based model and return the output model.
MLFlow will be used to log the parameters and metrics during our pipeline run.

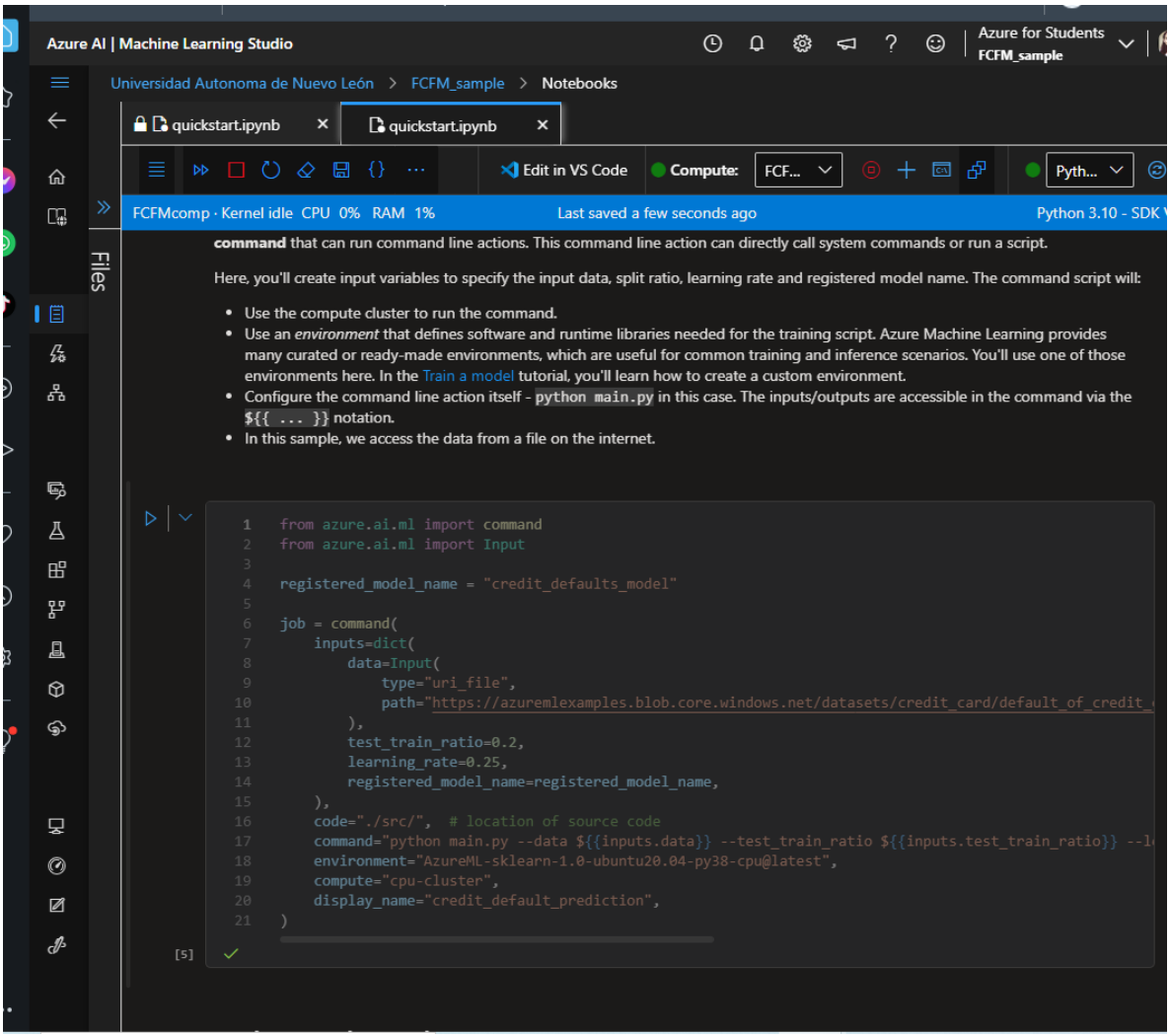

Azure AI | Machine Learning Studio
Azure for Students
FCFM_sample
Universidad Autonoma de Nuevo León > FCFM_sample > Notebooks
quickstart.ipynb
quickstart.ipynb
Edit in VS Code
Compute:
FCF...
Pyth...
FCFMcomp · Kernel idle CPU 0% RAM 1%
Last saved a few seconds ago
Python 3.10 - SDK V2
Files
This script handles the preprocessing of the data, splitting it into test and train data. It then consumes this data to train a tree based model and return the output model.
MLFlow will be used to log the parameters and metrics during our pipeline run.
The cell below uses IPython magic to write the training script into the directory you just created.
%%writefile {train_src_dir}/main.py
import os
import argparse
import pandas as pd
import mlflow
import mlflow.sklearn
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.metrics import classification_report
from sklearn.model_selection import train_test_split
def main():
"""Main function of the script."""
# input and output arguments
parser = argparse.ArgumentParser()
parser.add_argument("--data", type=str, help="path to input data")
parser.add_argument("--test_train_ratio", type=float, required=False, default=0.25)
parser.add_argument("--n_estimators", required=False, default=100, type=int)
parser.add_argument("--learning_rate", required=False, default=0.1, type=float)
parser.add_argument("--registered_model_name", type=str, help="model name")
args = parser.parse_args()
# Start Logging
mlflow.start_run()
# enable autologging
mlflow.sklearn.autolog()
###################
#<prepare the data>

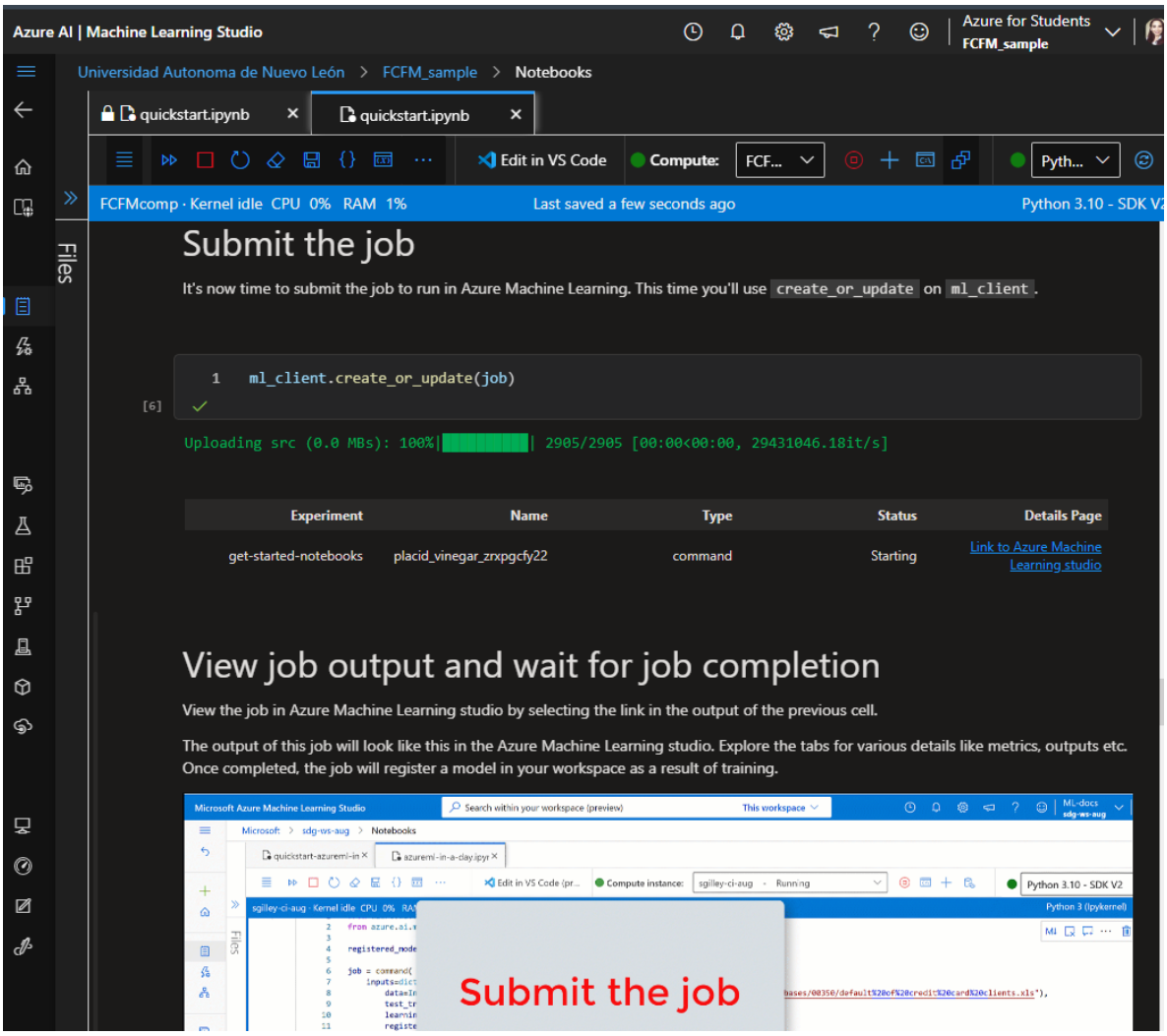

Azure AI | Machine Learning Studio
Azure for Students
FCFM_sample
Universidad Autonoma de Nuevo León > FCFM_sample > Notebooks
quickstart.ipynb
quickstart.ipynb
Edit in VS Code
Compute:
FCF...
Pyth...
FCFMcomp - Kernel idle CPU 0% RAM 1%
Last saved a few seconds ago
Python 3.10 - SDK V2
Files
print(" ".join(f"{k}={v}" for k, v in vars(args).items()))
print("input data:", args.data)
credit_df = pd.read_csv(args.data, header=1, index_col=0)
mlflow.log_metric("num_samples", credit_df.shape[0])
mlflow.log_metric("num_features", credit_df.shape[1] - 1)
train_df, test_df = train_test_split(
credit_df,
test_size=args.test_train_ratio,
)
####################
#</prepare the data>
####################
##################
#<train the model>
##################
# Extracting the label column
y_train = train_df.pop("default payment next month")
# convert the dataframe values to array
X_train = train_df.values
# Extracting the label column
y_test = test_df.pop("default payment next month")
# convert the dataframe values to array
X_test = test_df.values
print(f"Training with data of shape {X_train.shape}")
clf = GradientBoostingClassifier(
n_estimators=args.n_estimators, learning_rate=args.learning_rate
)
clf.fit(X_train, y_train)
y_pred = clf.predict(X_test)

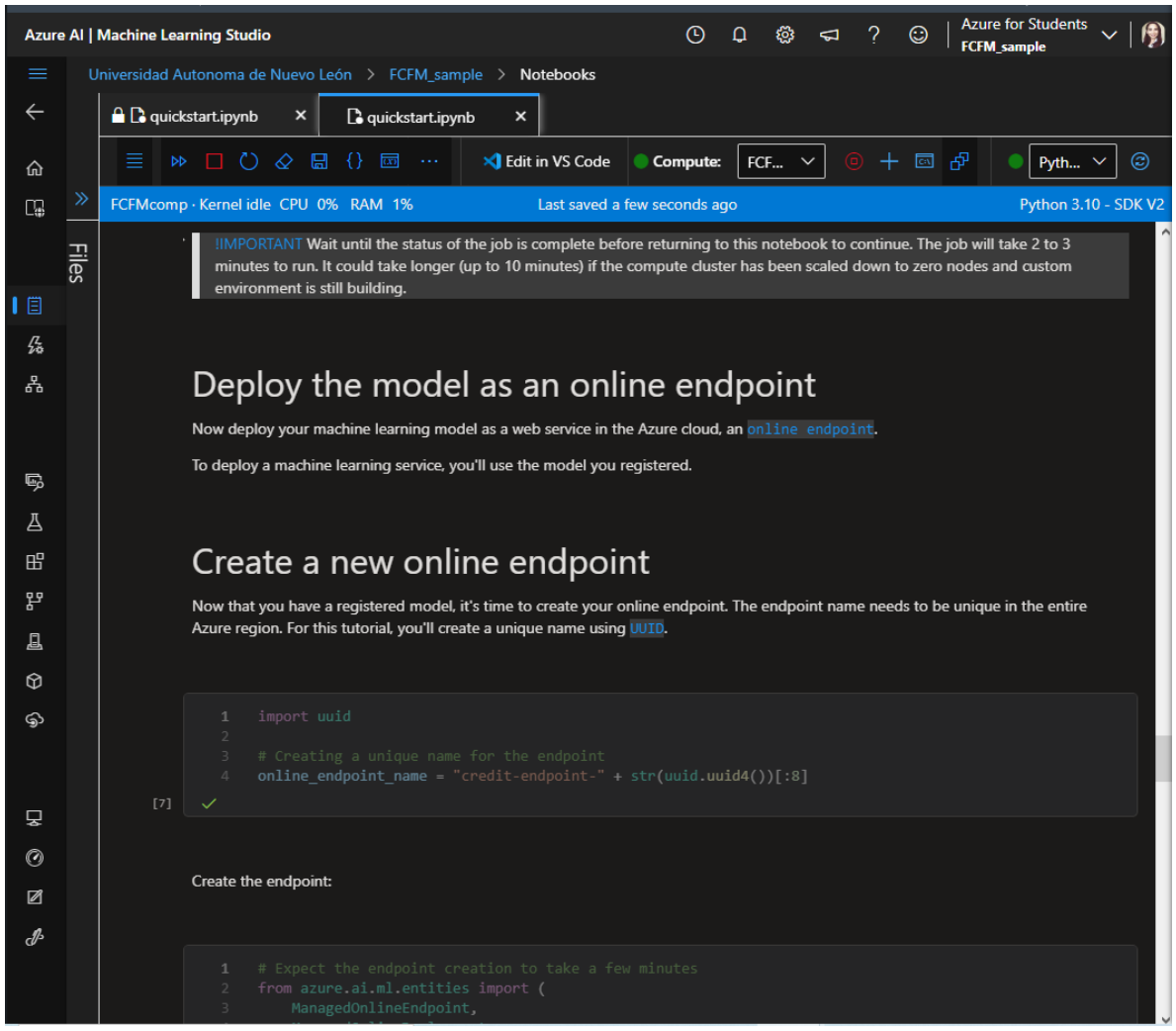

Azure AI | Machine Learning Studio
Azure for Students
FCFM_sample
Universidad Autonoma de Nuevo León > FCFM_sample > Notebooks
quickstart.ipynb
quickstart.ipynb
Edit in VS Code
Compute:
FCF...
Pyth...
FCFMcomp · Kernel idle CPU 0% RAM 1%
Last saved a few seconds ago
Python 3.10 - SDK V2
Files
74 ##################
75 #</train the model>
76 ##################
77
78 ##########################
79 #<save and register model>
80 ##########################
81 # Registering the model to the workspace
82 print("Registering the model via MLFlow")
83 mlflow.sklearn.log_model(
84 sk_model=clf,
85 registered_model_name=args.registered_model_name,
86 artifact_path=args.registered_model_name,
87)
88
89 # Saving the model to a file
90 mlflow.sklearn.save_model(
91 sk_model=clf,
92 path=os.path.join(args.registered_model_name, "trained_model"),
93)
94 ##########################
95 #</save and register model>
96 ##########################
97
98 # Stop Logging
99 mlflow.end_run()
100
101 if __name__ == "__main__":
102 main()
[3]
Writing ./src/main.py
As you can see in this script, once the model is trained, the model file is saved and registered to the workspace. Now you can use the registered model in inferencing endpoints.
You might need to select Refresh to see the new folder and script in your Files.

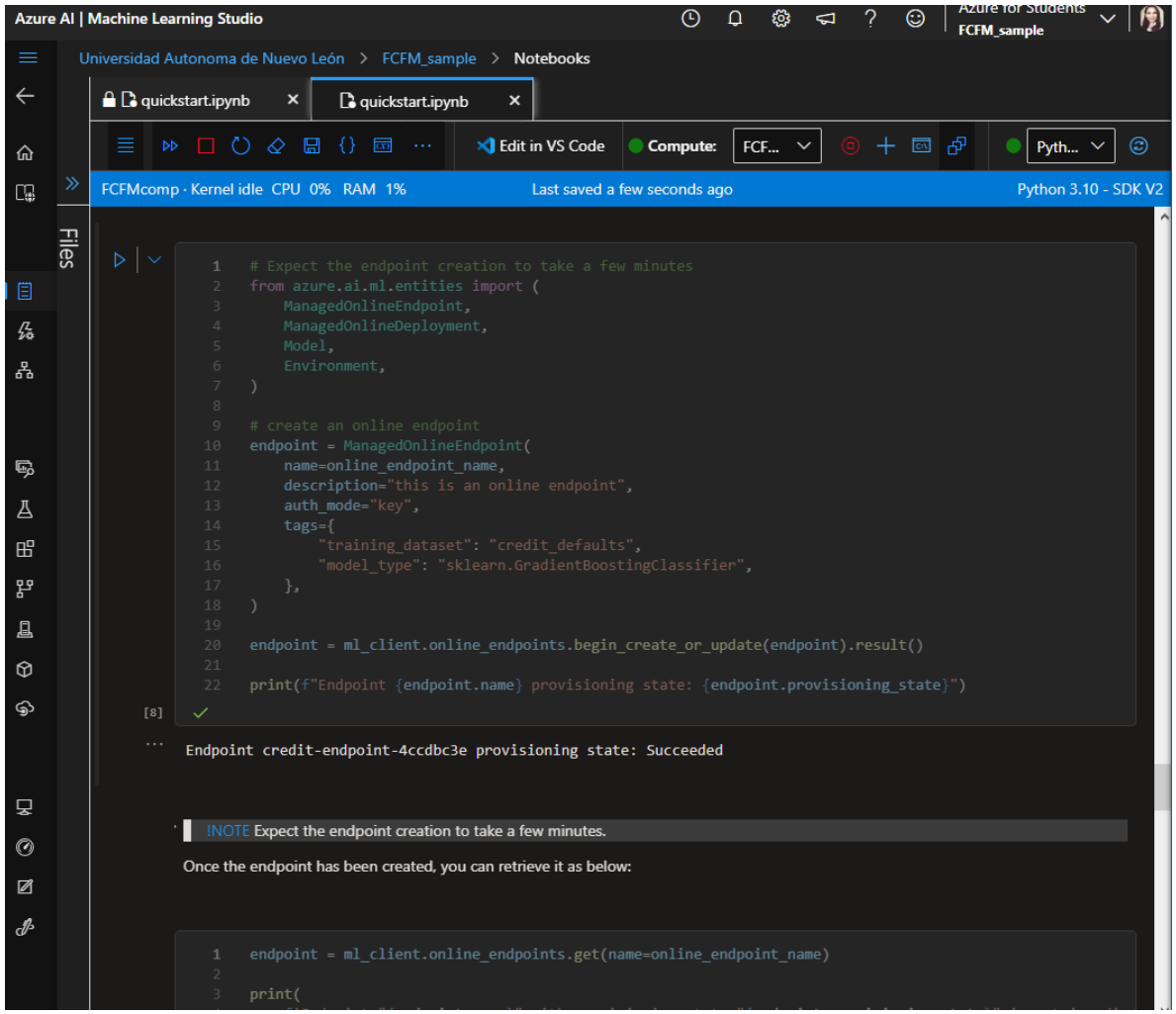

Azure AI | Machine Learning Studio
Azure for Students
FCFM_sample
Universidad Autonoma de Nuevo León > FCFM_sample > Notebooks
quickstart.ipynb
quickstart.ipynb
Edit in VS Code
Compute:
FCF...
Pyth...
FCFMcomp · Kernel idle CPU 0% RAM 1%
Last saved a few seconds ago
Python 3.10 - SDK V2
Files
As you can see in this script, once the model is trained, the model file is saved and registered to the workspace. Now you can use the registered model in inferencing endpoints.
You might need to select Refresh to see the new folder and script in your Files.
Notebooks
Files
Samples
Refresh
Create a compute cluster, a scalable way to run a training job
You already have a compute instance, which you're using to run the notebook. Now you'll add a second type of compute, a compute cluster that you'll use to run your training job. While a compute instance is a single node machine, a compute cluster can be single or multi-node machines with Linux or Windows OS, or a specific compute fabric like Spark.
You'll provision a Linux compute cluster. See the full list on VM sizes and prices .
For this example, you only need a basic cluster, so you'll use a Standard_DS3_v2 model with 2 vCPU cores, 7-GB RAM.
from azure.ai.ml.entities import AmlCompute
# Name assigned to the compute cluster
cpu_compute_target = "cpu-cluster"
try:
# let's see if the compute target already exists
cpu_cluster = ml_client.compute.get(cpu_compute_target)
print(
f"You already have a cluster named {cpu_compute_target}, we'll reuse it as is."
)

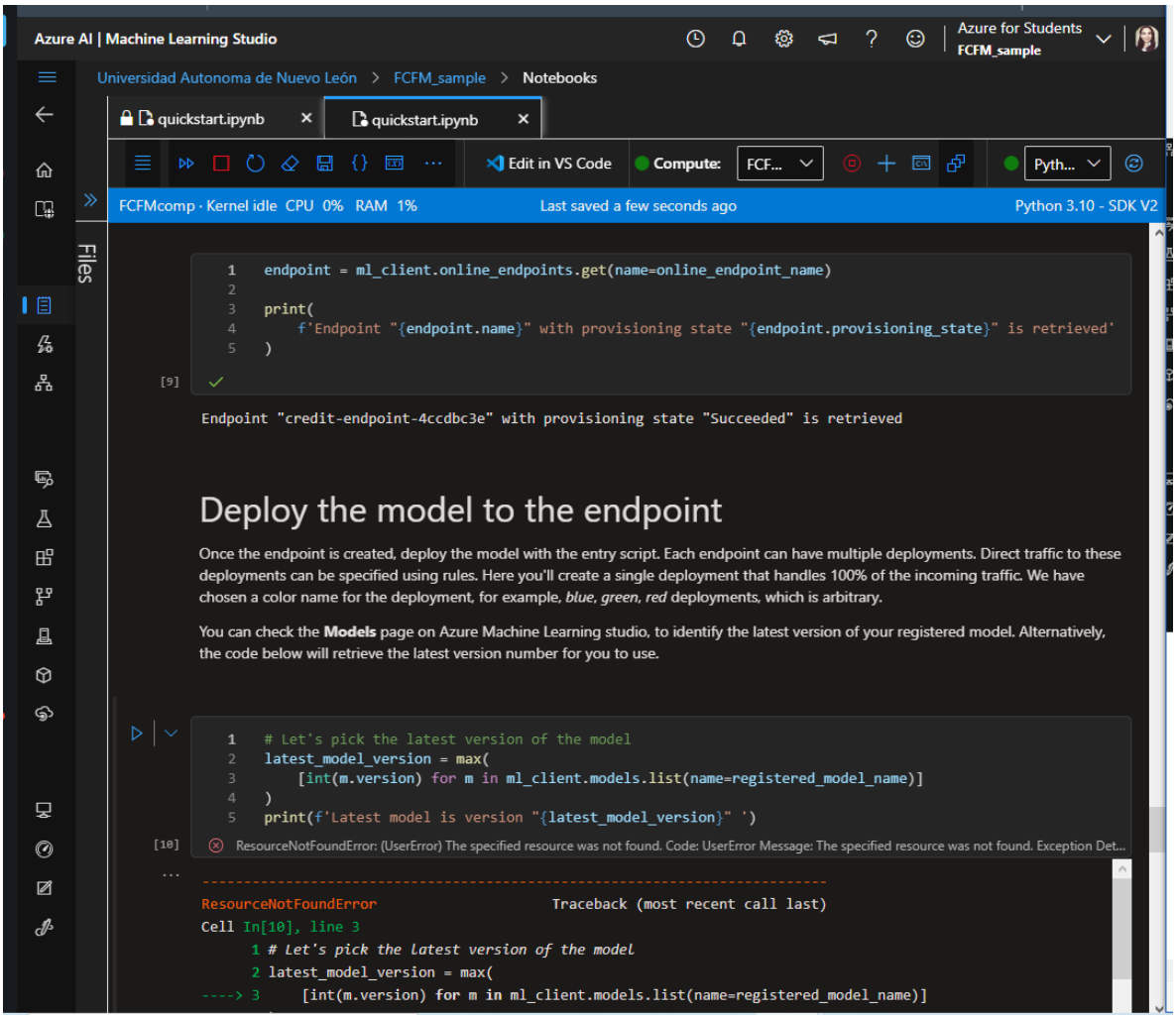

Azure AI | Machine Learning Studio
Azure for Students
FCFM_sample
Universidad Autonoma de Nuevo León > FCFM_sample > Notebooks
quickstart.ipynb
quickstart.ipynb
Edit in VS Code
Compute:
FCF...
Pyth...
FCFMcomp · Kernel idle CPU 0% RAM 1%
Last saved a few seconds ago
Python 3.10 - SDK V2
Files
print("Creating a new cpu compute target...")
# Let's create the Azure Machine Learning compute object with the intended parameters
cpu_cluster = AmlCompute(
name=cpu_compute_target,
# Azure Machine Learning Compute is the on-demand VM service
type="amlcompute",
# VM Family
size="STANDARD_DS3_V2",
# Minimum running nodes when there is no job running
min_instances=0,
# Nodes in cluster
max_instances=4,
# How many seconds will the node running after the job termination
idle_time_before_scale_down=180,
# Dedicated or LowPriority. The latter is cheaper but there is a chance of job termination
tier="Dedicated",
)
print(
f"AMLCompute with name {cpu_cluster.name} will be created, with compute size {cpu_cluster.size}"
)
# Now, we pass the object to MLClient's create_or_update method
cpu_cluster = ml_client.compute.begin_create_or_update(cpu_cluster)
[4]
Creating a new cpu compute target...
AMLCompute with name cpu-cluster will be created, with compute size STANDARD_DS3_V2
Configure the command
Now that you have a script that can perform the desired tasks, and a compute cluster to run the script, you'll use a general purpose command that can run command line actions. This command line action can directly call system commands or run a script.
Here, you'll create input variables to specify the input data, split ratio, learning rate and registered model name. The command script will:
Use the compute cluster to run the command.
Use an environment that defines software and runtime libraries needed for the training script. Azure Machine Learning provides

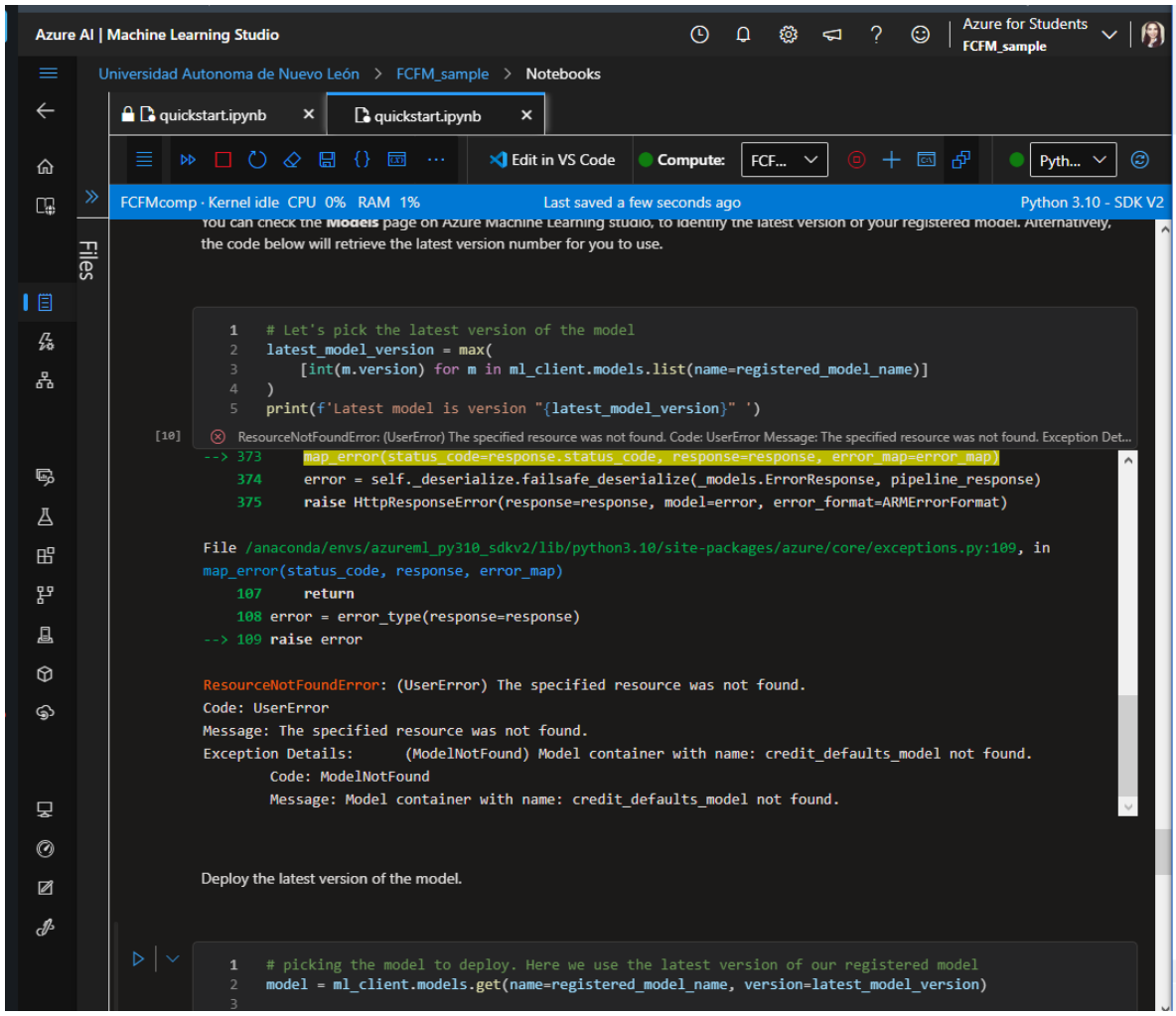

Azure AI | Machine Learning Studio
Azure for Students
FCFM_sample
Universidad Autonoma de Nuevo León > FCFM_sample > Notebooks
quickstart.ipynb
quickstart.ipynb
Edit in VS Code
Compute:
FCF...
Pyth...
FCFMcomp · Kernel idle CPU 0% RAM 1%
Last saved a few seconds ago
Python 3.10 - SDK
Files
command that can run command line actions. This command line action can directly call system commands or run a script.
Here, you'll create input variables to specify the input data, split ratio, learning rate and registered model name. The command script will:
Use the compute cluster to run the command.
Use an environment that defines software and runtime libraries needed for the training script. Azure Machine Learning provides many curated or ready-made environments, which are useful for common training and inference scenarios. You'll use one of those environments here. In the Train a model tutorial, you'll learn how to create a custom environment.
Configure the command line action itself - python main.py in this case. The inputs/outputs are accessible in the command via the ${{ ... }} notation.
In this sample, we access the data from a file on the internet.
1 from azure.ai.ml import command
2 from azure.ai.ml import Input
3
4 registered_model_name = "credit_defaults_model"
5
6 job = command(
7 inputs=dict(
8 data=Input(
9 type="uri_file",
10 path="https://azuremlexamples.blob.core.windows.net/datasets/credit_card/default_of_credit_
11),
12 test_train_ratio=0.2,
13 learning_rate=0.25,
14 registered_model_name=registered_model_name,
15),
16 code="./src/", # location of source code
17 command="python main.py --data ${{inputs.data}} --test_train_ratio ${{inputs.test_train_ratio}} --l
18 environment="AzureML-sklearn-1.0-ubuntu20.04-py38-cpu@latest",
19 compute="cpu-cluster",
20 display_name="credit_default_prediction",
21)
[5]

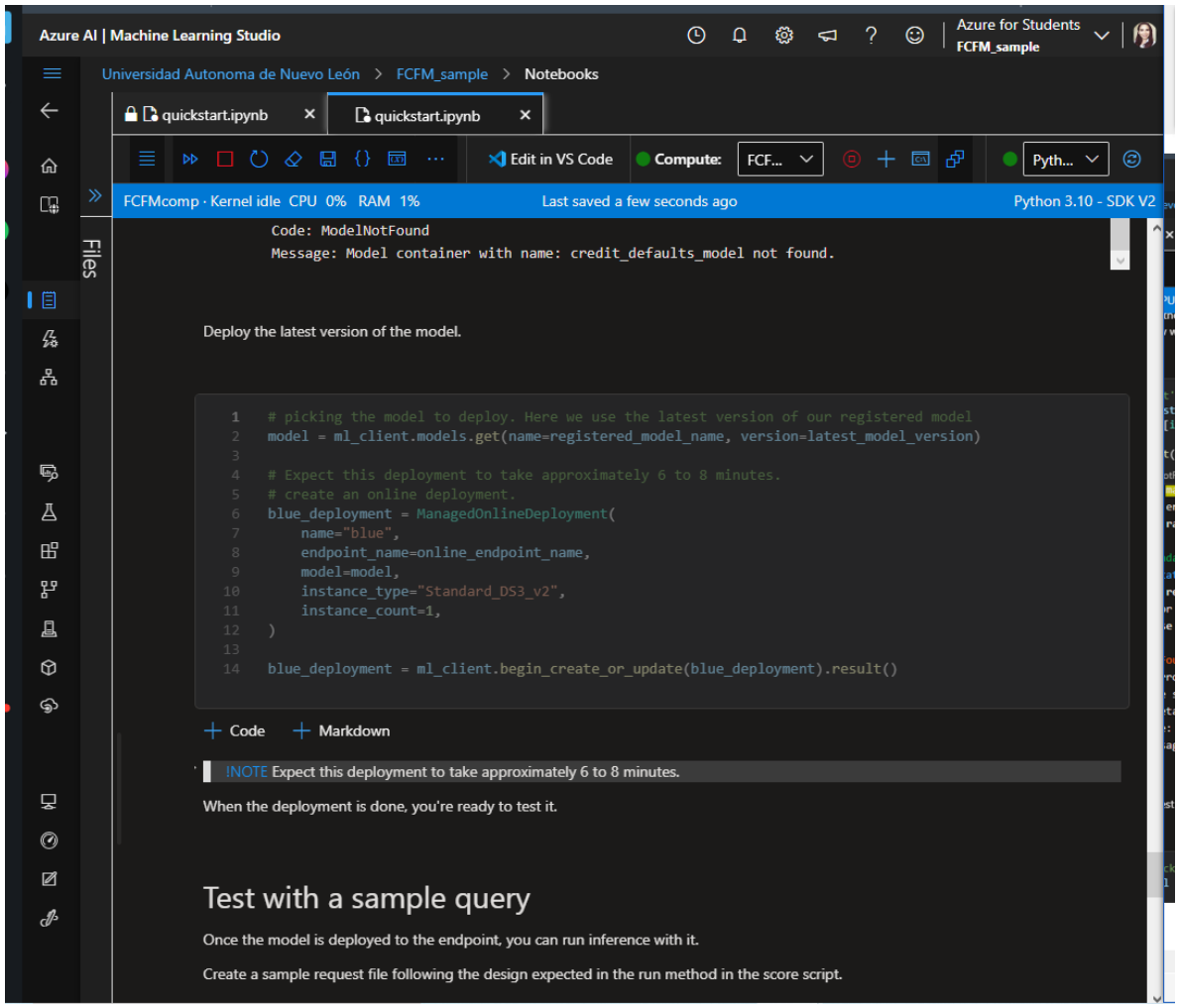

Azure AI | Machine Learning Studio
Azure for Students
FCFM_sample
Universidad Autonoma de Nuevo León > FCFM_sample > Notebooks
quickstart.ipynb
quickstart.ipynb
Edit in VS Code
Compute:
FCF...
Pyth...
FCFMcomp · Kernel idle CPU 0% RAM 1%
Last saved a few seconds ago
Python 3.10 - SDK V2
Files
Submit the job
It's now time to submit the job to run in Azure Machine Learning. This time you'll use create_or_update on ml_client.
[6]
1 ml_client.create_or_update(job)
Uploading src (0.0 MBs): 100%| | 2905/2905 [00:00<00:00, 29431046.18it/s]
Experiment
Name
Type
Status
Details Page
get-started-notebooks
placid_vinegar_zrxpgcfy22
command
Starting
Link to Azure Machine Learning studio
View job output and wait for job completion
View the job in Azure Machine Learning studio by selecting the link in the output of the previous cell.
The output of this job will look like this in the Azure Machine Learning studio. Explore the tabs for various details like metrics, outputs etc. Once completed, the job will register a model in your workspace as a result of training.
Submit the job

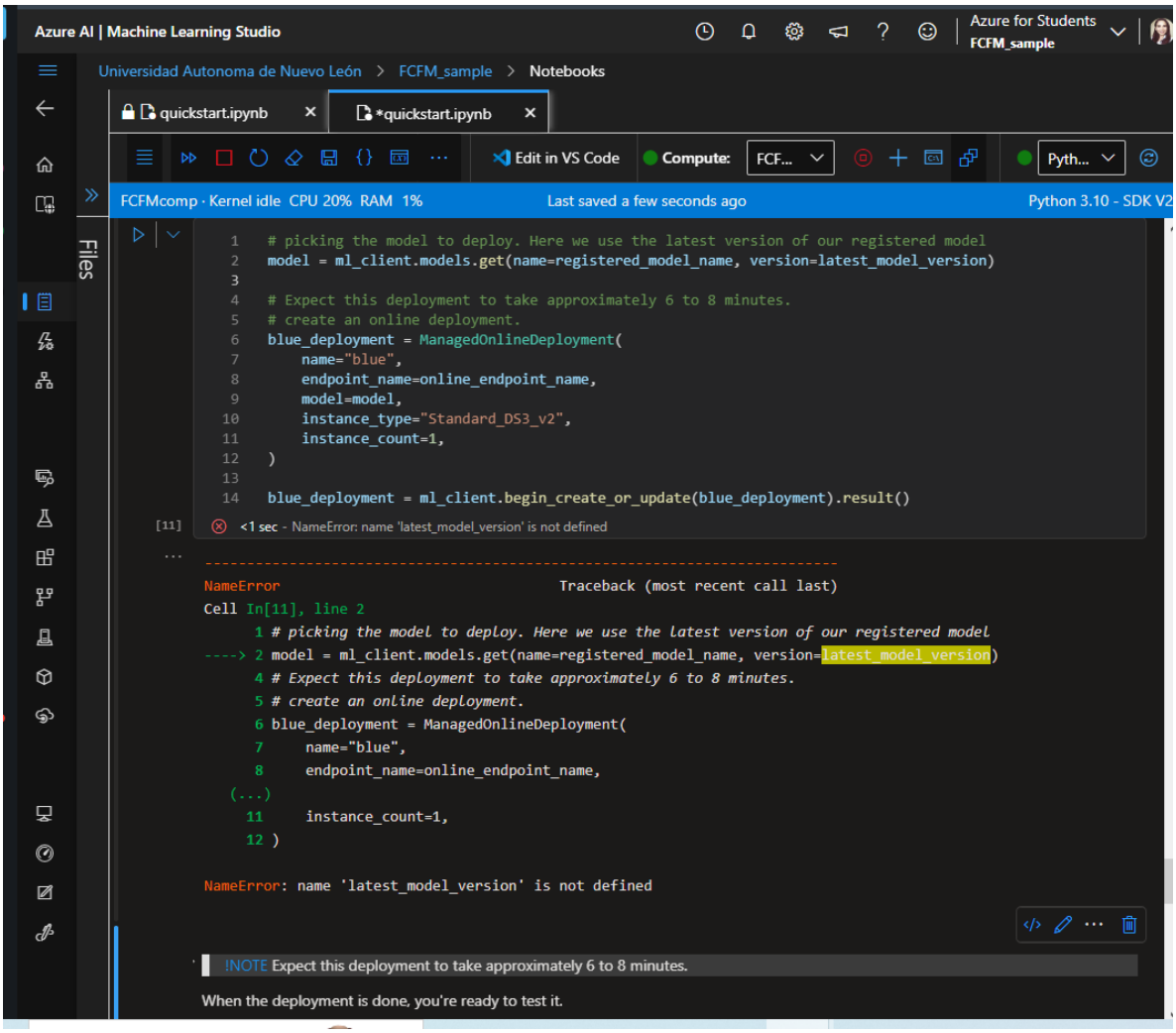

Azure AI | Machine Learning Studio
Azure for Students
FCFM_sample
Universidad Autonoma de Nuevo León > FCFM_sample > Notebooks
quickstart.ipynb
quickstart.ipynb
Edit in VS Code
Compute:
FCF...
Pyth...
FCFMcomp · Kernel idle CPU 0% RAM 1%
Last saved a few seconds ago
Python 3.10 - SDK V2
Files
!IMPORTANT Wait until the status of the job is complete before returning to this notebook to continue. The job will take 2 to 3 minutes to run. It could take longer (up to 10 minutes) if the compute cluster has been scaled down to zero nodes and custom environment is still building.
Deploy the model as an online endpoint
Now deploy your machine learning model as a web service in the Azure cloud, an online endpoint.
To deploy a machine learning service, you'll use the model you registered.
Create a new online endpoint
Now that you have a registered model, it's time to create your online endpoint. The endpoint name needs to be unique in the entire Azure region. For this tutorial, you'll create a unique name using UUID.
[7]
import uuid
# Creating a unique name for the endpoint
online_endpoint_name = "credit-endpoint-" + str(uuid.uuid4())[:8]
Create the endpoint:
# Expect the endpoint creation to take a few minutes
from azure.ai.ml.entities import (
ManagedOnlineEndpoint,

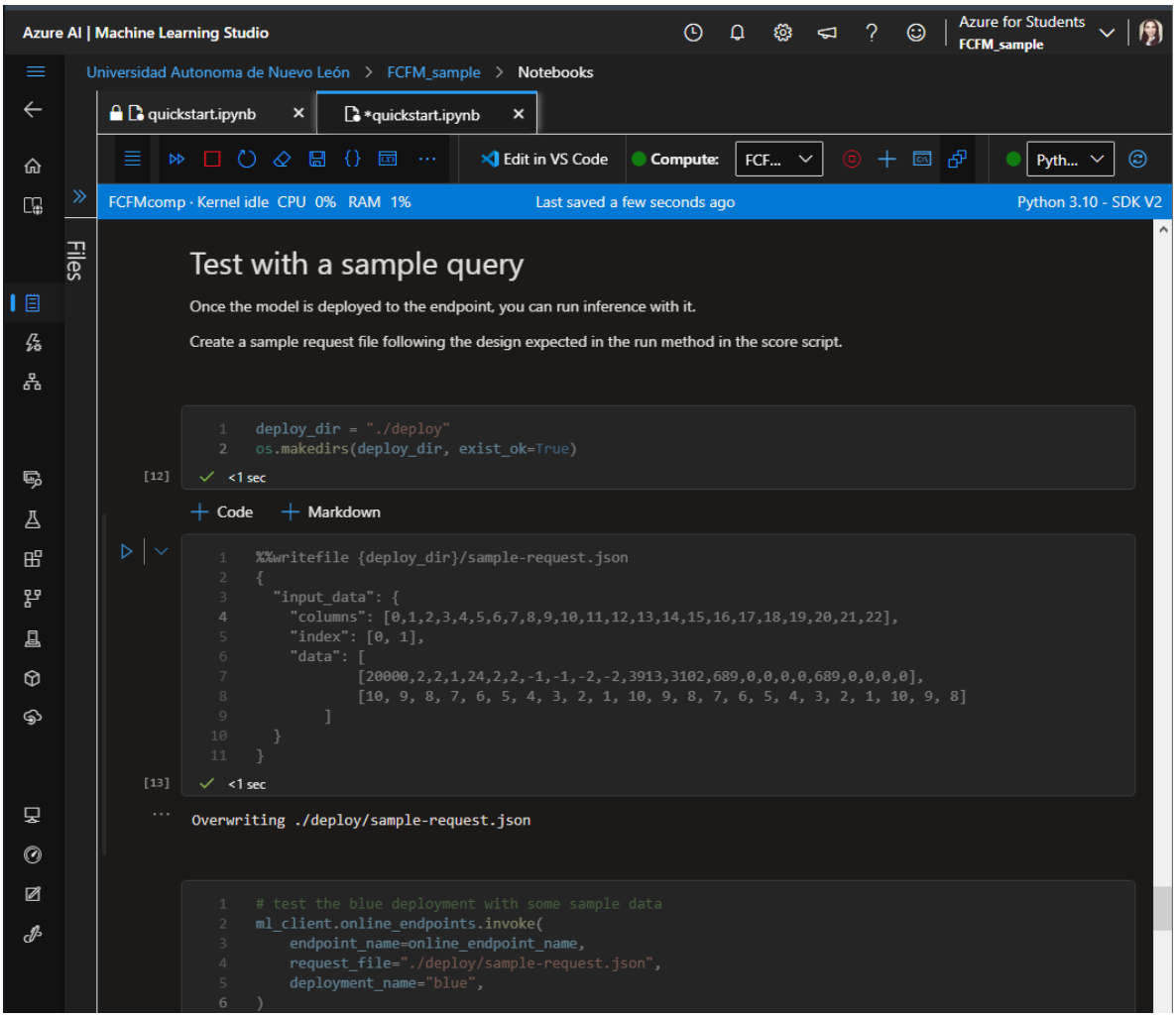

Azure AI | Machine Learning Studio
Azure for Students
FCFM_sample
Universidad Autonoma de Nuevo León > FCFM_sample > Notebooks
quickstart.ipynb
quickstart.ipynb
Edit in VS Code
Compute:
FCF...
Pyth...
FCFMcomp · Kernel idle CPU 0% RAM 1%
Last saved a few seconds ago
Python 3.10 - SDK V2
Files
# Expect the endpoint creation to take a few minutes
from azure.ai.ml.entities import (
ManagedOnlineEndpoint,
ManagedOnlineDeployment,
Model,
Environment,
)
# create an online endpoint
endpoint = ManagedOnlineEndpoint(
name=online_endpoint_name,
description="this is an online endpoint",
auth_mode="key",
tags={
"training_dataset": "credit_defaults",
"model_type": "sklearn.GradientBoostingClassifier",
},
)
endpoint = ml_client.online_endpoints.begin_create_or_update(endpoint).result()
print(f"Endpoint {endpoint.name} provisioning state: {endpoint.provisioning_state}")
[8]
Endpoint credit-endpoint-4ccdbc3e provisioning state: Succeeded
!NOTE Expect the endpoint creation to take a few minutes.
Once the endpoint has been created, you can retrieve it as below:
endpoint = ml_client.online_endpoints.get(name=online_endpoint_name)
print(

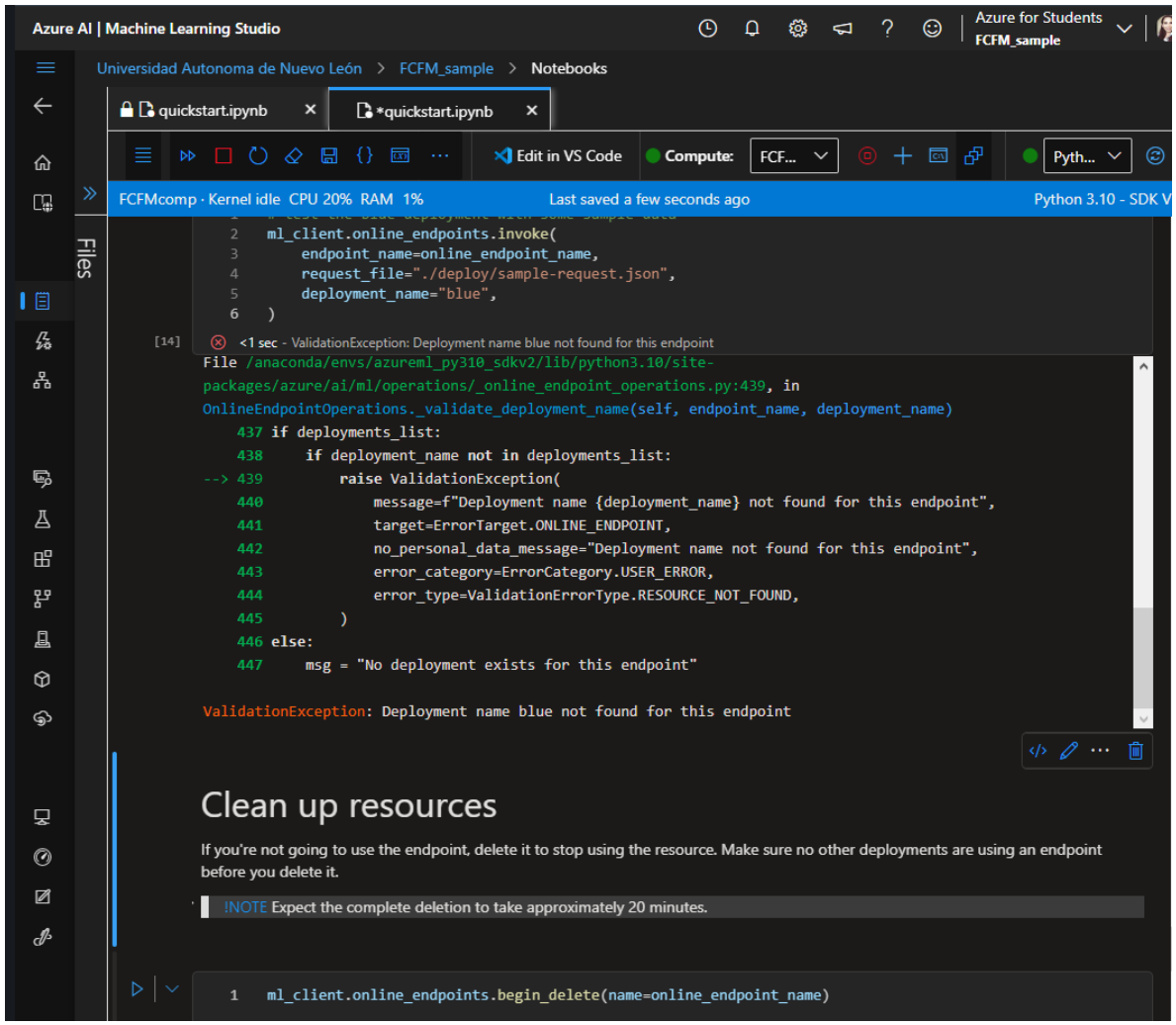

Azure AI | Machine Learning Studio
Azure for Students
FCFM_sample
Universidad Autonoma de Nuevo León > FCFM_sample > Notebooks
quickstart.ipynb
quickstart.ipynb
Edit in VS Code
Compute:
FCF...
Pyth...
FCFMcomp · Kernel idle CPU 0% RAM 1%
Last saved a few seconds ago
Python 3.10 - SDK V2
Files
endpoint = ml_client.online_endpoints.get(name=online_endpoint_name)
print(
f'Endpoint "{endpoint.name}" with provisioning state "{endpoint.provisioning_state}" is retrieved'
)
[9]
Endpoint "credit-endpoint-4ccdbc3e" with provisioning state "Succeeded" is retrieved
Deploy the model to the endpoint
Once the endpoint is created, deploy the model with the entry script. Each endpoint can have multiple deployments. Direct traffic to these deployments can be specified using rules. Here you'll create a single deployment that handles 100% of the incoming traffic. We have chosen a color name for the deployment, for example, blue, green, red deployments, which is arbitrary.
You can check the Models page on Azure Machine Learning studio, to identify the latest version of your registered model. Alternatively, the code below will retrieve the latest version number for you to use.
# Let's pick the latest version of the model
latest_model_version = max(
[int(m.version) for m in ml_client.models.list(name=registered_model_name)]
)
print(f'Latest model is version "{latest_model_version}" ')
[10]
ResourceNotFoundError: (UserError) The specified resource was not found. Code: UserError Message: The specified resource was not found. Exception Det...
ResourceNotFoundError Traceback (most recent call last)
Cell In[10], line 3
1 # Let's pick the latest version of the model
2 latest_model_version = max(
----> 3 [int(m.version) for m in ml_client.models.list(name=registered_model_name)]

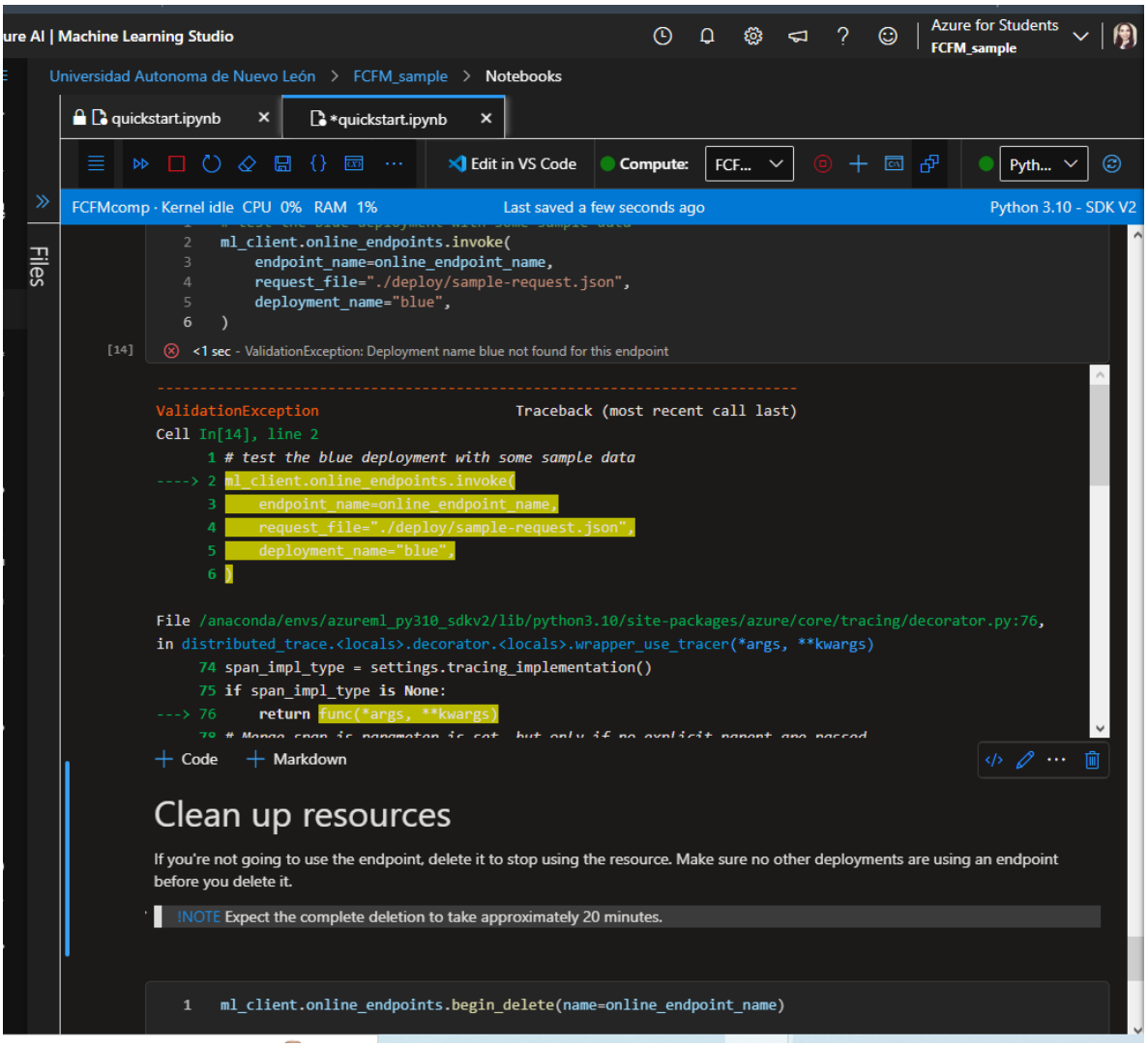

Azure AI | Machine Learning Studio
Azure for Students
FCFM_sample
Universidad Autonoma de Nuevo León > FCFM_sample > Notebooks
quickstart.ipynb
quickstart.ipynb
Edit in VS Code
Compute:
FCF...
Pyth...
FCFMcomp · Kernel idle CPU 0% RAM 1%
Last saved a few seconds ago
Python 3.10 - SDK V2
Files
You can check the Models page on Azure Machine Learning studio, to identify the latest version of your registered model. Alternatively, the code below will retrieve the latest version number for you to use.
# Let's pick the latest version of the model
latest_model_version = max(
[int(m.version) for m in ml_client.models.list(name=registered_model_name)]
)
print(f'Latest model is version "{latest_model_version}" ')
[10]
ResourceNotFoundError: (UserError) The specified resource was not found. Code: UserError Message: The specified resource was not found. Exception Det...
--> 373 map_error(status_code=response.status_code, response=response, error_map=error_map)
374 error = self._deserialize.failsafe_deserialize(_models.ErrorResponse, pipeline_response)
375 raise HttpResponseError(response=response, model=error, error_format=ARMErrorFormat)
File /anaconda/envs/azureml_py310_sdkv2/lib/python3.10/site-packages/azure/core/exceptions.py:109, in map_error(status_code, response, error_map)
107 return
108 error = error_type(response=response)
--> 109 raise error
ResourceNotFoundError: (UserError) The specified resource was not found.
Code: UserError
Message: The specified resource was not found.
Exception Details: (ModelNotFound) Model container with name: credit_defaults_model not found.
Code: ModelNotFound
Message: Model container with name: credit_defaults_model not found.
Deploy the latest version of the model.
# picking the model to deploy. Here we use the latest version of our registered model
model = ml_client.models.get(name=registered_model_name, version=latest_model_version)

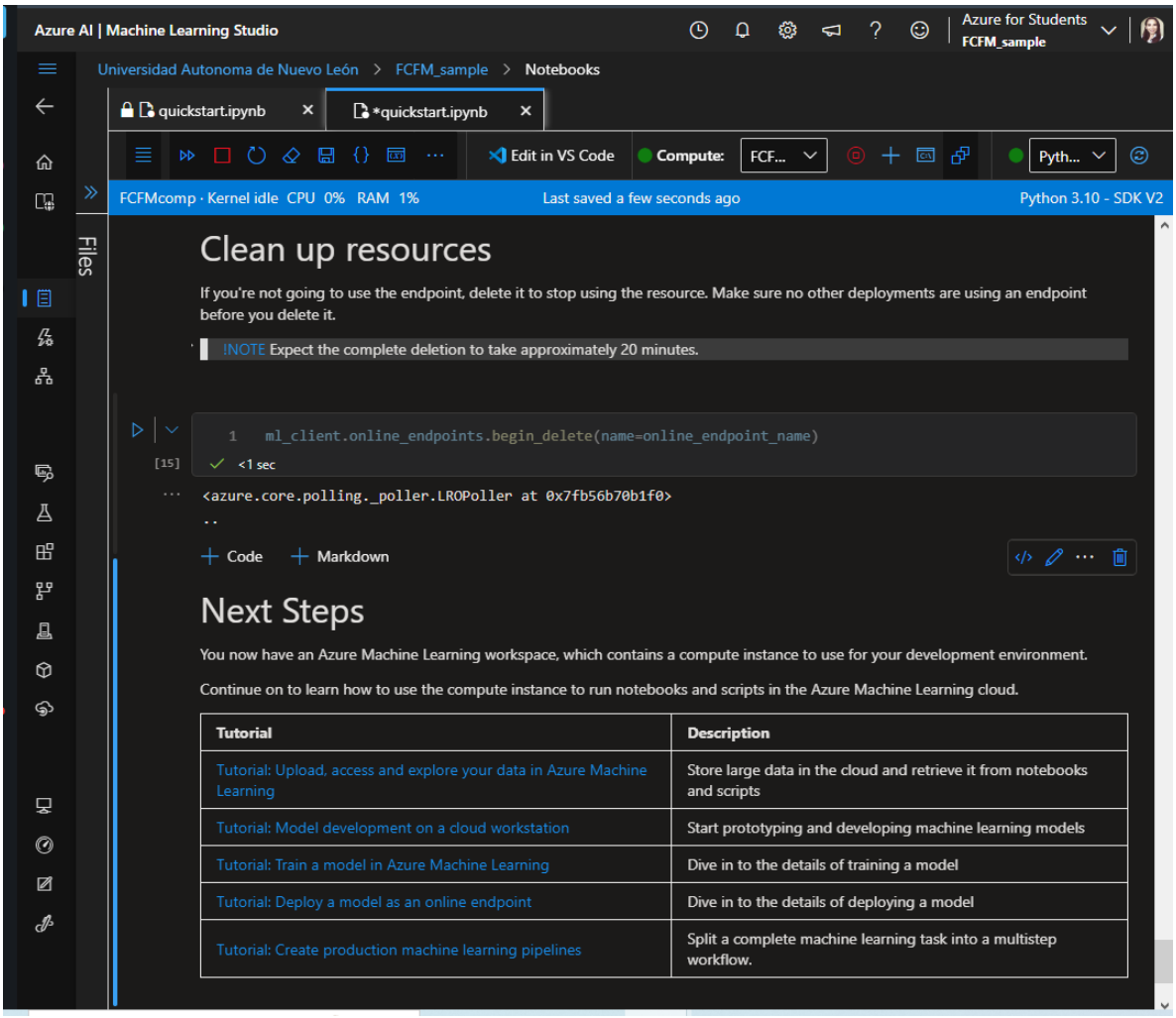

Azure AI | Machine Learning Studio
Azure for Students
FCFM_sample
Universidad Autonoma de Nuevo León > FCFM_sample > Notebooks
quickstart.ipynb
quickstart.ipynb
Edit in VS Code
Compute:
FCF...
Pyth...
FCFMcomp · Kernel idle CPU 0% RAM 1%
Last saved a few seconds ago
Python 3.10 - SDK V2
Files
Code: ModelNotFound
Message: Model container with name: credit_defaults_model not found.
Deploy the latest version of the model.
# picking the model to deploy. Here we use the latest version of our registered model
model = ml_client.models.get(name=registered_model_name, version=latest_model_version)
# Expect this deployment to take approximately 6 to 8 minutes.
# create an online deployment.
blue_deployment = ManagedOnlineDeployment(
name="blue",
endpoint_name=online_endpoint_name,
model=model,
instance_type="Standard_DS3_v2",
instance_count=1,
)
blue_deployment = ml_client.begin_create_or_update(blue_deployment).result()
Code
Markdown
NOTE Expect this deployment to take approximately 6 to 8 minutes.
When the deployment is done, you're ready to test it.
Test with a sample query
Once the model is deployed to the endpoint, you can run inference with it.
Create a sample request file following the design expected in the run method in the score script.

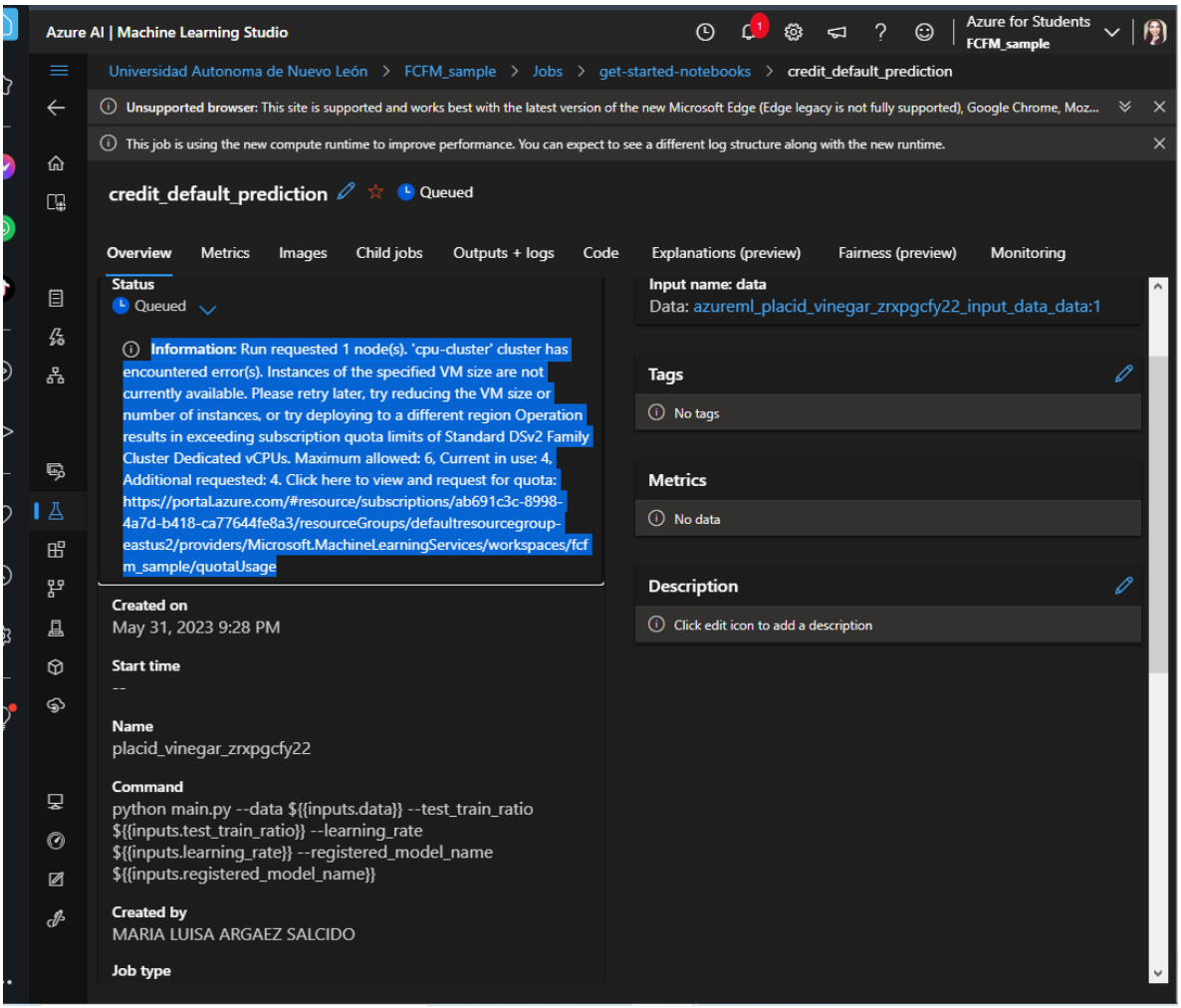

Azure AI | Machine Learning Studio
Azure for Students
FCFM_sample
Universidad Autonoma de Nuevo León > FCFM_sample > Notebooks
quickstart.ipynb
*quickstart.ipynb
Edit in VS Code
Compute:
FCF...
Pyth...
FCFMcomp · Kernel idle CPU 20% RAM 1%
Last saved a few seconds ago
Python 3.10 - SDK V2
Files
# picking the model to deploy. Here we use the latest version of our registered model
model = ml_client.models.get(name=registered_model_name, version=latest_model_version)
# Expect this deployment to take approximately 6 to 8 minutes.
# create an online deployment.
blue_deployment = ManagedOnlineDeployment(
name="blue",
endpoint_name=online_endpoint_name,
model=model,
instance_type="Standard_DS3_v2",
instance_count=1,
)
blue_deployment = ml_client.begin_create_or_update(blue_deployment).result()
[11]
<1 sec - NameError: name 'latest_model_version' is not defined
NameError Traceback (most recent call last)
Cell In[11], line 2
1 # picking the model to deploy. Here we use the latest version of our registered model
----> 2 model = ml_client.models.get(name=registered_model_name, version=latest_model_version)
4 # Expect this deployment to take approximately 6 to 8 minutes.
5 # create an online deployment.
6 blue_deployment = ManagedOnlineDeployment(
7 name="blue",
8 endpoint_name=online_endpoint_name,
(...)
11 instance_count=1,
12)
NameError: name 'latest_model_version' is not defined
!NOTE Expect this deployment to take approximately 6 to 8 minutes.
When the deployment is done, you're ready to test it.


Azure AI | Machine Learning Studio
Azure for Students
FCFM_sample
Universidad Autonoma de Nuevo León > FCFM_sample > Notebooks
quickstart.ipynb
*quickstart.ipynb
Edit in VS Code
Compute:
FCF...
Pyth...
FCFMcomp · Kernel idle CPU 0% RAM 1%
Last saved a few seconds ago
Python 3.10 - SDK V2
Files
Test with a sample query
Once the model is deployed to the endpoint, you can run inference with it.
Create a sample request file following the design expected in the run method in the score script.
deploy_dir = "./deploy"
os.makedirs(deploy_dir, exist_ok=True)
[12]
<1 sec
Code
Markdown
%%writefile {deploy_dir}/sample-request.json
{
"input_data": {
"columns": [0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22],
"index": [0, 1],
"data": [
[20000,2,2,1,24,2,2,-1,-1,-2,-2,3913,3102,689,0,0,0,0,689,0,0,0,0],
[10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 10, 9, 8]
]
}
}
[13]
<1 sec
Overwriting ./deploy/sample-request.json
test the blue deployment with some sample data
ml_client.online_endpoints.invoke(
endpoint_name=online_endpoint_name,
request_file="./deploy/sample-request.json",
deployment_name="blue",
)


Azure AI | Machine Learning Studio
Azure for Students
FCFM_sample
Universidad Autonoma de Nuevo León > FCFM_sample > Notebooks
quickstart.ipynb
*quickstart.ipynb
Edit in VS Code
Compute:
FCF...
Pyth...
FCFMcomp · Kernel idle CPU 20% RAM 1%
Last saved a few seconds ago
Python 3.10 - SDK V
Files
ml_client.online_endpoints.invoke(
endpoint_name=online_endpoint_name,
request_file="./deploy/sample-request.json",
deployment_name="blue",
)
[14]
<1 sec - ValidationException: Deployment name blue not found for this endpoint
File /anaconda/envs/azureml_py310_sdkv2/lib/python3.10/site-packages/azure/ai/ml/operations/_online_endpoint_operations.py:439, in OnlineEndpointOperations._validate_deployment_name(self, endpoint_name, deployment_name)
437 if deployments_list:
438 if deployment_name not in deployments_list:
--> 439 raise ValidationException(
440 message=f"Deployment name {deployment_name} not found for this endpoint",
441 target=ErrorTarget.ONLINE_ENDPOINT,
442 no_personal_data_message="Deployment name not found for this endpoint",
443 error_category=ErrorCategory.USER_ERROR,
444 error_type=ValidationErrorType.RESOURCE_NOT_FOUND,
445)
446 else:
447 msg = "No deployment exists for this endpoint"
ValidationException: Deployment name blue not found for this endpoint
Clean up resources
If you're not going to use the endpoint, delete it to stop using the resource. Make sure no other deployments are using an endpoint before you delete it.
!NOTE Expect the complete deletion to take approximately 20 minutes.
1 ml_client.online_endpoints.begin_delete(name=online_endpoint_name)


ure AI | Machine Learning Studio
Azure for Students
FCFM_sample
Universidad Autonoma de Nuevo León > FCFM_sample > Notebooks
quickstart.ipynb
*quickstart.ipynb
Edit in VS Code
Compute:
FCF...
Pyth...
FCFMcomp · Kernel idle CPU 0% RAM 1%
Last saved a few seconds ago
Python 3.10 - SDK V2
Files
ml_client.online_endpoints.invoke(
endpoint_name=online_endpoint_name,
request_file="./deploy/sample-request.json",
deployment_name="blue",
)
[14]
<1 sec - ValidationException: Deployment name blue not found for this endpoint
ValidationException Traceback (most recent call last)
Cell In[14], line 2
1 # test the blue deployment with some sample data
----> 2 ml_client.online_endpoints.invoke(
3 endpoint_name=online_endpoint_name,
4 request_file="./deploy/sample-request.json",
5 deployment_name="blue",
6)
File /anaconda/envs/azureml_py310_sdkv2/lib/python3.10/site-packages/azure/core/tracing/decorator.py:76,
in distributed_trace.<locals>.decorator.<locals>.wrapper_use_tracer(*args, **kwargs)
74 span_impl_type = settings.tracing_implementation()
75 if span_impl_type is None:
---> 76 return func(*args, **kwargs)
Code
Markdown
Clean up resources
If you're not going to use the endpoint, delete it to stop using the resource. Make sure no other deployments are using an endpoint before you delete it.
NOTE Expect the complete deletion to take approximately 20 minutes.
1 ml_client.online_endpoints.begin_delete(name=online_endpoint_name)


Azure AI | Machine Learning Studio
Azure for Students
FCFM_sample
Universidad Autonoma de Nuevo León > FCFM_sample > Notebooks
quickstart.ipynb
*quickstart.ipynb
Edit in VS Code
Compute:
FCF...
Pyth...
FCFMcomp · Kernel idle CPU 0% RAM 1%
Last saved a few seconds ago
Python 3.10 - SDK V2
Files
Clean up resources
If you're not going to use the endpoint, delete it to stop using the resource. Make sure no other deployments are using an endpoint before you delete it.
NOTE Expect the complete deletion to take approximately 20 minutes.
1 ml_client.online_endpoints.begin_delete(name=online_endpoint_name)
[15]
<1 sec
<azure.core.polling._poller.LROPoller at 0x7fb56b70b1f0>
..
Code
Markdown
Next Steps
You now have an Azure Machine Learning workspace, which contains a compute instance to use for your development environment.
Continue on to learn how to use the compute instance to run notebooks and scripts in the Azure Machine Learning cloud.
Tutorial
Description
Tutorial: Upload, access and explore your data in Azure Machine Learning
Store large data in the cloud and retrieve it from notebooks and scripts
Tutorial: Model development on a cloud workstation
Start prototyping and developing machine learning models
Tutorial: Train a model in Azure Machine Learning
Dive in to the details of training a model
Tutorial: Deploy a model as an online endpoint
Dive in to the details of deploying a model
Tutorial: Create production machine learning pipelines
Split a complete machine learning task into a multistep workflow.



La instancia de estudiante es de 4 cores, 28 GB y 54 GB disk, por eso no fue posible seguir con el ejercicio.