

Azure MLops: Azure Machine Learning + Azure Devops.

Azure Machine Learning Studio:

- A collaborative tool for the team members to work with.
- **pip freeze** → This shows the list of installed packages.
- Activities are automatically logged under Azure Machine Learning Platform.
- **Auto-ML:** Automated machine learning is the process of automating the tasks of applying machine learning to real-world problems. Auto-ML potentially includes every stage from beginning with a raw dataset to building a machine learning model ready for deployment.

Services of importance: Azure Devops.


Go to Azure Devops Service →

- Create new Project.
- The Repos folder is like a GitHub containing relevant codes for the project, user can work directly over it and do updates accordingly.
- Why is a workspace needed?
→ Workspace helps to collaborate when working over a project.

Project:

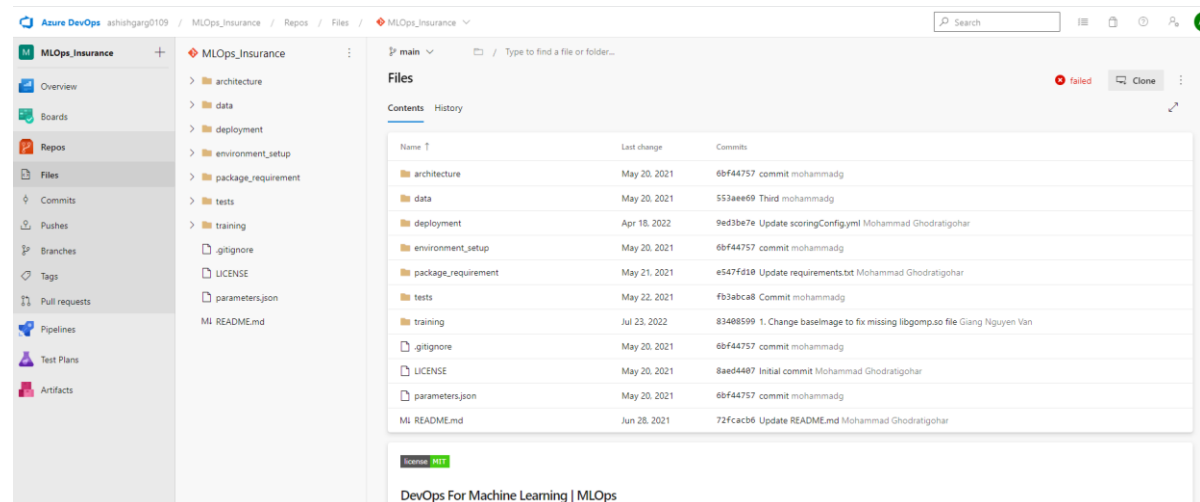
- Make new Project.
- **Make a service connection with your Azure Subscription**, by going under project settings.
- We will write the code to setup the infrastructure for Azure Machine Learning Resources. Go to project → Pipelines → Library → Specify variable group name.
Define variables under group like:
BASE_NAME: mlopsash1(This will be the name of azure resource this is creating Ex:
WORKSPACE_NAME: mlops-ash-aml1
AZURE_RM_SVC_CONNECTION: azure-resource-connection (This is the name of service connection created earlier).

Variables

Name ↑		Value
AZURE_RM_SVC_CONNECTION		azure-resource-connection
BASE_NAME		mlopswsh1
LOCATION		centralus
RESOURCE_GROUP		mlops-wsh-rg1
WORKSPACE_NAME		mlops-wsh-aml1
WORKSPACE_SVC_CONNECTION		aml-workspace-connection

Under Azure DevOps:






All the relevant code is in one place under Repos Folder.



Set the yaml pipeline configuration as shown below to create infrastructure under azure portal (Infrastructure as a code):

```
1 pr: none
2 trigger: none
3
4 variables:
5   - group: mlops-ash-vg
6
7 stages:
8   - stage: 'Dev'
9     displayName: 'Dev'
10    jobs:
11      - job: "Provision_Dev"
12        displayName: "Provision Dev resources"
13        pool:
14          vmImage: 'ubuntu-latest'
15          timeoutInMinutes: 0
16        steps:
17          - task: AzureResourceGroupDeployment@2
18            inputs:
19              azureSubscription: '$(AZURE_RM_SVC_CONNECTION)'
20              action: 'Create Or Update Resource Group'
21              resourceGroupName: '$(RESOURCE_GROUP)'
22              location: $(LOCATION)
23              templateLocation: 'Linked artifact'
24              csmFile: '$(Build.SourcesDirectory)/environment_setup/cloud-environment.json'
25              overrideParameters: '-baseName $(BASE_NAME) -location $(LOCATION) -workspace $(WORKSPACE_NAME)'
26              deploymentMode: 'Incremental'
27              displayName: 'Deploy OH resources to Azure'
28
```

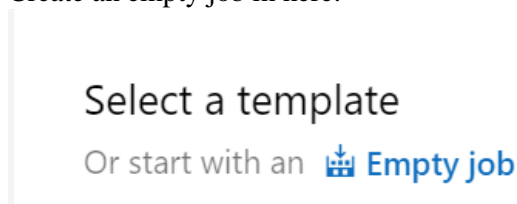
Run the pipeline and azure resources will be created as below:

<input type="checkbox"/>	 mlops-ash-aml1	Azure Machine Learning workspace	Central US
<input type="checkbox"/>	 mlopsash1-AML-AI	Application Insights	Central US
<input type="checkbox"/>	 mlopsash1-AML-KV	Key vault	Central US
<input type="checkbox"/>	 mlopsash1amlcr	Container registry	Central US
<input type="checkbox"/>	 mlopsash1amlsa	Storage account	Central US

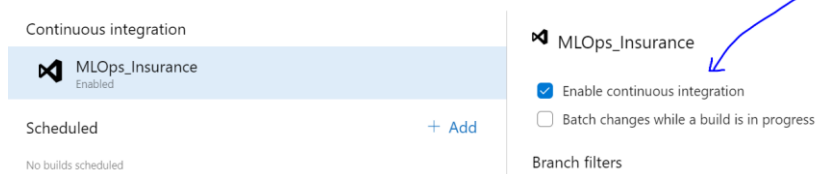
- Data Store contains the different endpoints towards data points.

Suppose if a data-scientist did some manual change, then that change needs to trigger the pipeline and create the new model.

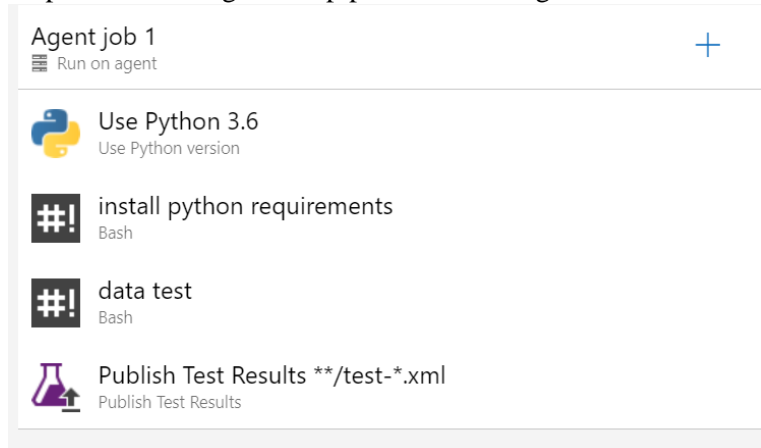
- Create a separate CI pipe for it. Use classic editor this time.
- Create an empty job in here:



- Enable Continuous integration:



- Steps can be configured in pipeline something like this:



For tests: Use command: `pytest training/train_test.py --doctest-modules --junitxml=junit/test-results.xml --cov=data_test --cov-report=xml --cov-report=html`.
Under data test inline option in order to write script command.

- Azure also has CLI to interact with. Install it in your local computer and excess it through terminal. Command for azure cli starts with az.
Specific extensions can be installed like this through terminal.

```
PS C:\Users\asgarg> az extension add -n azure-cli-ml
```

- User can also add azure CLI in the pipeline like this for various different tasks:

install python require...

Bash

data test

Bash

Publish Test Results **/...

Publish Test Results

install Azure ML CLI

Azure CLI

Script type

Shell

Script Location *

Inline script

Inline Script *

az extension add -n azure-cli-ml

install Azure ML CLI

Azure CLI

Create Azure ML Work...

Azure CLI

Azure ML Compute Cl...

Azure CLI

Inline script

Inline Script *

az ml workspace create -g \$(azureml.resourceGroup) -w \$(azureml.workspaceName) -l \$(azureml.location) --exist-ok --yes

install Azure ML CLI

Azure CLI

Create Azure ML Work...

Azure CLI

Azure ML Compute Cl...

Azure CLI

Inline script

Inline Script *

az ml computetarget create amlcompute -g \$(azureml.resourceGroup) -w \$(azureml.workspaceName) -n \$(amlcompute.clusterName) -s \$(amlcompute.vmSize) --min-nodes \$(amlcompute.minNodes) --max-nodes \$(amlcompute.maxNodes) --idle-seconds-before-scaledown \$(amlcompute.idleSecondsBeforeScaledown)

Publish Test Results

install Azure ML CLI

Azure CLI

Create Azure ML Work...

Azure CLI

Azure ML Compute Cl...

Azure CLI

upload data to Datast...

Azure CLI

Script Location

Inline script

Inline Script *

az ml datastore upload -w \$(azureml.workspaceName) -g \$(azureml.resourceGroup) -n \$(az ml datastore show-default -w \$(azureml.workspaceName) -g \$(azureml.resourceGroup) --query name -o tsv) -p data -u insurance --overwrite true