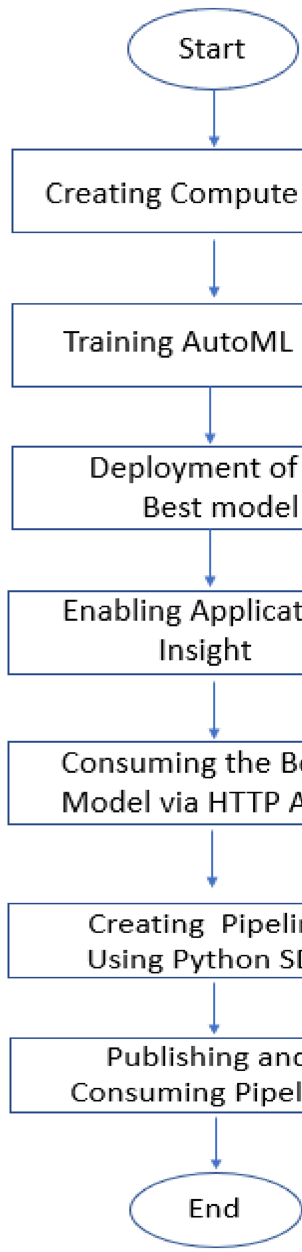
Operationalizing Machine Learning:

In this project, we will use Azure ML studio to configure an Automated ML experiment, deploy the best model, enable logging to monitor and collect data from the deployed model, and interact with this model. Next, we will use Python SDK to create, publish, and consume a pipeline.

Architectural Diagram:

In the following diagram, we will define all the steps of our project from start to end, and give an overview of each step:



- Creating compute cluster: configure a compute cluster to run experiments through project steps.
- Training AutoML model: create an experiment using Automated ML in Azure ML Studio.
- Deployment of the best model: deploy the best model from the generated Automated ML models.
- Enabling logging: to enable Application Insights and retrieve logs of the deployed model.
- Consuming the Best Model via HTTP API: consume the deployed model using Swagger.
- Creating Pipeline Using Python SDK: create and schedule ML pipeline run.
- Publishing and Consuming Pipeline: publish and interact with a pipeline via an HTTP API endpoint.

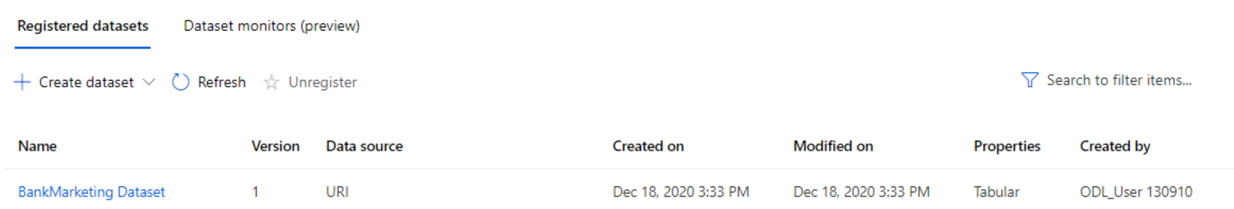
Future Work:

We can add more data or use the feature engineering technique to add more columns which may give us better result in our model training. Also, this project use ACI service to deploy the model which is known by its fast and simplicity and as a future improvement we can try to use AKS service that can expand but it will take more effort.

Key Steps:

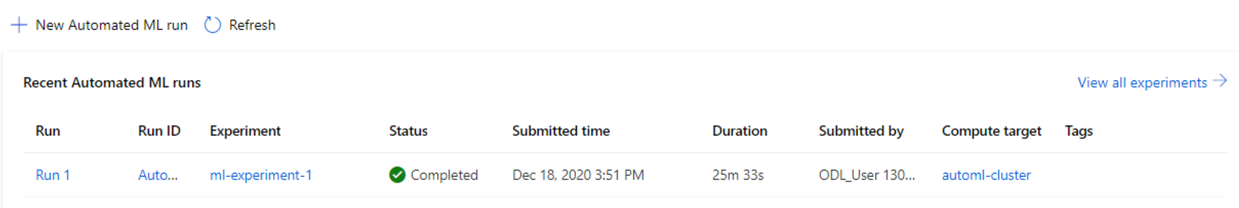
Part 1: Deploy model in Azure ML Studio

1- Create dataset called "BankMarketing Dataset" in ML Studio, figure 1 shows the registered datasets in our project workspace



Name	Version	Data source	Created on	Modified on	Properties	Created by
BankMarketing Dataset	1	URI	Dec 18, 2020 3:33 PM	Dec 18, 2020 3:33 PM	Tabular	ODL_User 130910

2- Create an Automated ML run called “ml-experiment-1”, within this step we will upload the dataset and configure the compute cluster, figure 2 shows the experiment as completed.



Run	Run ID	Experiment	Status	Submitted time	Duration	Submitted by	Compute target	Tags
Run 1	Auto...	ml-experiment-1	Completed	Dec 18, 2020 3:51 PM	25m 33s	ODL_User 130...	automl-cluster	

3- After the experiment run completed, we will deploy the best model using Azure Container Instance (ACI). The deployed model is called “bankmarketing-model”

4- Enable application insights of the deployed model by editing the provided logs.py script using the terminal, figure 3 shows the output of the script run and figure 4 shows the application insights enabled in the details tab of the endpoint.

```
demouser@labvm MINGW64 ~/desktop/nd00333_AZMLND_C2-master/starter_files
$ python logs.py
WARNING - Warning: Falling back to use azure cli login credentials.
If you run your code in unattended mode, i.e., where you can't give a user input, then we recommend to use ServicePrincipalAuthentication or MsiAuthentication.
Please refer to aka.ms/aml-notebook-auth for different authentication mechanisms in azureml-sdk.
2020-12-18T16:35:43,310960344+00:00 - iot-server/run
2020-12-18T16:35:43,310357014+00:00 - gunicorn/run
2020-12-18T16:35:43,310276810+00:00 - rsyslog/run
/usr/sbin/nginx: /azureml-envs/azureml_e1251fbb90896de9de2cde1255eba7bb/lib/libcrypto.so.1.0.0: no version information available (required by /usr/sbin/nginx)
/usr/sbin/nginx: /azureml-envs/azureml_e1251fbb90896de9de2cde1255eba7bb/lib/libcrypto.so.1.0.0: no version information available (required by /usr/sbin/nginx)
/usr/sbin/nginx: /azureml-envs/azureml_e1251fbb90896de9de2cde1255eba7bb/lib/libssl.so.1.0.0: no version information available (required by /usr/sbin/nginx)
/usr/sbin/nginx: /azureml-envs/azureml_e1251fbb90896de9de2cde1255eba7bb/lib/libssl.so.1.0.0: no version information available (required by /usr/sbin/nginx)
/usr/sbin/nginx: /azureml-envs/azureml_e1251fbb90896de9de2cde1255eba7bb/lib/libssl.so.1.0.0: no version information available (required by /usr/sbin/nginx)
2020-12-18T16:35:43,350541120+00:00 - nginx/run
rsyslogd: /azureml-envs/azureml_e1251fbb90896de9de2cde1255eba7bb/lib/libuuid.so.1: no version information available (required by rsyslogd)
EdgeHubConnectionString and IOTEDGE_IOTHUBHOSTNAME are not set. Exiting...
2020-12-18T16:35:43,519904476+00:00 - iot-server/finish 1 0
2020-12-18T16:35:43,521386750+00:00 - Exit code 1 is normal. Not restarting iot-server.
Starting gunicorn 19.9.0
Listening at: http://127.0.0.1:31311 (11)
Using worker: sync
worker timeout is set to 300
Booting worker with pid: 39
SPARK_HOME not set. Skipping PySpark Initialization.
Generating new fontManager, this may take some time...
Initializing logger
2020-12-18 16:35:45,968 | root | INFO | Starting up app insights client
Starting up app insights client
2020-12-18 16:35:45,968 | root | INFO | Starting up request id generator
Starting up request id generator
2020-12-18 16:35:45,968 | root | INFO | Starting up app insight hooks
Starting up app insight hooks
2020-12-18 16:35:45,968 | root | INFO | Invoking user's init function
```

bankmarketing-model

CPU
1.8

Memory
4 GB

Application Insights enabled
true

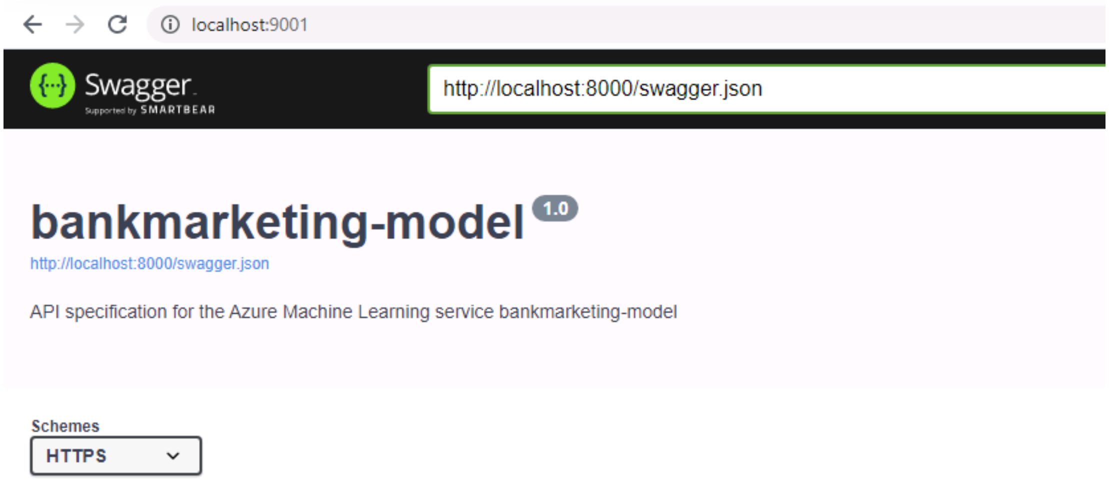
Application Insights url
https://portal.azure.com#resource/subscriptions/da775cb9-9ca6-4943-ad21-26dfa99526fc/resourcegroups/aml-quickstarts-130910/providers/microsoft.insights/components/mlappinsight130910

5- Azure provides a Swagger JSON file for deployed model in details tab, we will download the file locally and move it to the same directory of the provided `serve.py` and `swagger.sh` files.

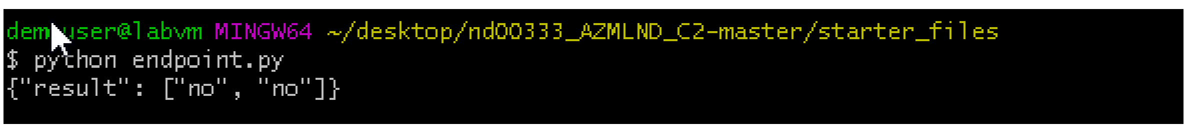
6- Run two scripts:

- `swagger.sh` which will download the latest Swagger container, and it will run it on port 9001.
- `serve.py` will start a Python server on port 8000.

as figure 5, the Swagger runs on localhost showing the HTTP API methods and responses for our model



7- Azure provides URL and key of the deployed model in consume tab, we will copy them and modify the `scoring_uri` and the `key` in the provided endpoint.py file to match the copied values.
8- Execute `endpoint.py` script, figure 6 shows the output of this execution.



9- update the provided `benchmark.sh` file with the same URL and Key and execute the file, figure 7 shows a part of this execution.

```
Concurrency Level:      1
Time taken for tests:   0.856 seconds
Complete requests:      10
Failed requests:        0
Total transferred:      2590 bytes
Total body sent:        10640
HTML transferred:       320 bytes
Requests per second:    11.68 [#/sec] (mean)
Time per request:       85.591 [ms] (mean)
Time per request:       85.591 [ms] (mean, across all concurrent requests)
Transfer rate:          2.96 [Kbytes/sec] received
                        12.14 kb/s sent
                        15.09 kb/s total

Connection Times (ms)
              min  mean[+/-sd] median   max
Connect:        2    2   0.5      2       3
Processing:    74   83   8.6     83     101
Waiting:       74   83   8.7     83     101
Total:         76   85   8.9     86     104

Percentage of the requests served within a certain time (ms)
  50%     86
  66%     88
  75%     90
  80%     95
  90%    104
  95%    104
  98%    104
  99%    104
 100%    104 (longest request)
```

Part 2: Publish an ML Pipeline using Python SDK

1- Upload the provided notebook aml-piplines-with-automated-machine-learning-step.ipynb to Azure ML studio and update experiment, dataset, and compute cluster to match the existing Automated ML run.
2- Run through the cells to create, publish, and consume the pipeline
3- The following screenshots show the pipeline has been created as figure 8, the pipeline endpoint in Azure ML studio as figure 9, the dataset with AutoML as figure 10, the published pipeline overview where the status is active as figure 11, and the output of "Use RunDetails Widget" for the pipeline run as figure 12

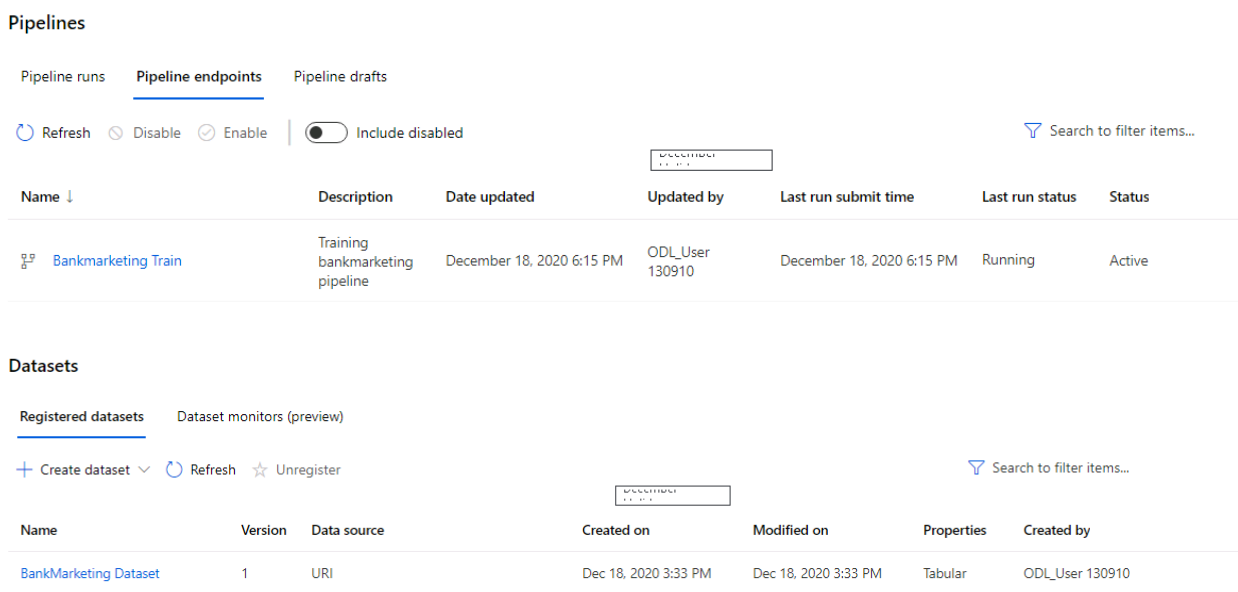
Pipelines

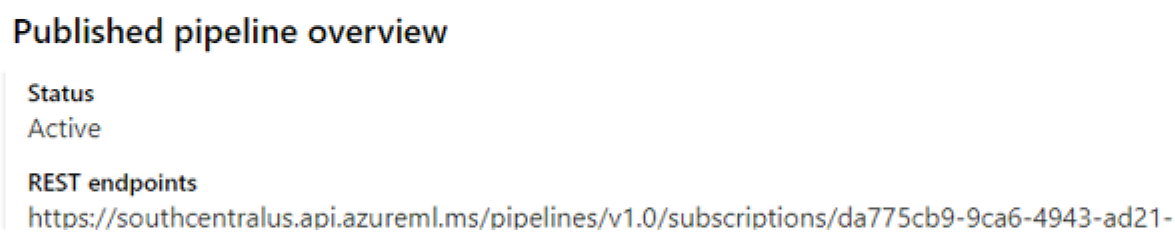
Pipeline runs | Pipeline endpoints | Pipeline drafts

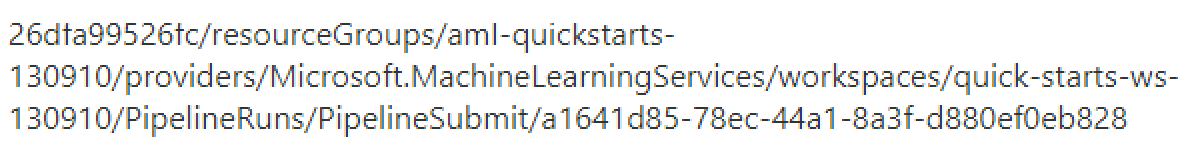
+ New pipeline | Refresh

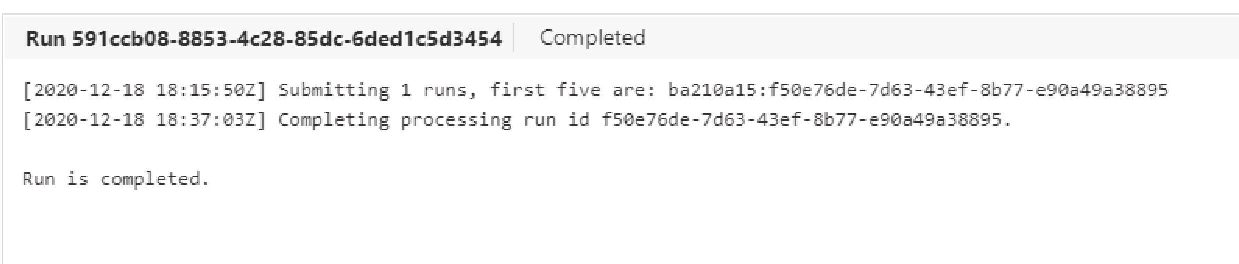
Add filter

Run	Run ID	Experiment	Status	Description	Submitted time	Duration	Submitted by	Tags
Run 1	591ccb...	pipeline-rest-endpoint	Running		Dec 18, 2020 6:15 PM	5m 37s	ODL_User 130...	azureml.pipelinei
Run 64	36d0bb...	ml-experiment-1	Completed	pipeline_with_automlstep	Dec 18, 2020 5:39 PM	31m 35s	ODL_User 130...	azureml.pipeline









Screen Recording:

https://www.youtube.com/watch?v=SKPvYNLECiQ